D4.5 Final version of the self-assessment nexus engine with the corresponding validation

Lead: Nuria Nievas (EUT)

Date: 28/02/2025

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Project Deliverable

Project Number	Project Acronym	Project Title
101003881	NEXOGENESIS	Facilitating the next generation of effective and intelligent water-related policies, utilizing artificial intelligence and reinforcement learning to assess the water-energy-food-ecosystem (WEFE) nexus

Instrument:	Thematic Priority
H2020 RIA	LC-CLA-14-2020

Title
D4.5 Final version of the self-assessment nexus engine with the corresponding validation

Contractual Delivery Date	Actual Delivery Date
M42: February 2025	M42

Start Date of the project	Duration
01 September 2021	48 months

Organisation name of lead contractor for this deliverable	Document version
EUT	1.0

Dissemination level	Deliverable Type
Public	**Demonstrator**

Authors (organisations)
Nuria Nievas (EUT), Chaymaa Dkouk (EUT), and Lluís Echeverria (EUT)
Reviewers (organisations)
Antonio Trabucco (CMCC)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Abstract

The objective of NEXOGENESIS (NXG) in Work Package 4 (WP4) is to enhance decision-making on WEFE interlinkages through the NExus Policy Assessment Tool (NEPAT) platform, enabling intelligent assessments in policy-making. This document accompanies Deliverable D4.5 *Final version of the self-assessment nexus engine*, classified as a "Demonstrator", and provides a detailed explanation of the NEPAT, also referred to as the Self-Learning Nexus Assessment Engine (SLNAE).

NEPAT is a platform designed to analyze the interconnections between Water, Energy, Food, and Ecosystems (WEFE) under diverse climate and socioeconomic scenarios. It assesses the effects of policies on WEFE sectors, delivers tailored policy recommendations to efficiently achieve nexus-related goals, and promotes informed dialogue and collaborative decision-making on WEFE challenges.

Deliverable D4.5 complements the digital solutions developed in Task T4.5 *Self-Learning Nexus Assessment Engine*, as the final output of WP4. This engine is the result of a co-creation process with stakeholders across all case studies. It builds upon the design established in Task T4.1 *Design of the Decision-Making Framework*, which was presented in Deliverable D4.1 *Self-Learning Nexus Engine Specifications and Technical Design*.

In D4.1, the six foundational WP4 pillars were introduced, which define the core components of the SLNAE and link them to outputs from other NXG work packages. These pillars include the WEFE Policy Framework, WEFE Goals, Targets, and Indicators, Nexus Complexity Modeling, Decision Support System Functionalities, Data Sharing Tools, and the Graphical User Interface.

The final version of the NEPAT is embedded into the public release accessible at the following URLs: https://slnae.nexogenesis.eu or https://nepat.nexogenesis.eu.

Related Deliverables:
D3.4 Complexity science models implemented for all the Case Studies including explanatory manuals
D3.6 Sensitivity/Uncertainty Analysis Report
D4.1 Self-learning nexus engine specifications and technical design
D4.3. Simulation Policy Framework
D4.4. Core module of the self-learning nexus engine

Keywords
SLNAE, NEPAT, co-creation, testing, UI, design, policy impact evaluation, AI, DSS

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Abbreviation/Acronyms

AI	Artificial Intelligence
API	Application Protocol Interface
CS	Case Study
DMP	Data Management Plan
DSS	Decision Support System
GUI/UI	Graphical User Interface/User Interface
HTTPS	Hypertext Transfer Protocol Secure
ICT	Information and Communication Technologies
JSON	JavaScript Object Notation
JWT	Json Web Tokens
ML	Machine Learning
MS	Milestone
NEPAT	NExus Policy Assessment Tool
NXG	Nexogenesis project
NXGT	Nexogenesis tool
ORM	Object Relational Mapper
PP	Policy Package
RCP	Representative Concentration Pathways
REST	REpresentational State Transfer
RL	Reinforcement Learning
SDGs	Sustainable Development Goals
SDM	System Dynamic Model
SLNAE	Self-Learning Nexus Assessment Engine
SH	Stakeholder
SPF	Simulation Policy Framework
SSP	Shared Socioeconomic Pathways
WEFE	Water-Energy-Food-Ecosystem
WP	Work Package
WS	Workshop

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Contents

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Figures

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Tables

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

1. Introduction

The Self-Learning Nexus Assessment Engine (SLNAE), also known as the Nexogenesis Tool (NXGT) or NExus Policy Assessment Tool (NEPAT), is a sophisticated interactive platform designed to analyze the interconnections between Water, Energy, Food, and Ecosystems (WEFE) across various climate and socioeconomic scenarios. Developed as part of the Nexogenesis (NXG) project, NEPAT applies a holistic approach to nexus governance, which is utilized in project case studies to evaluate policy impacts. The tool assesses the effects of policies on WEFE sectors under future scenarios that integrate both Representative Concentration Pathways (RCPs) and Shared Socioeconomic Pathways (SSPs).

Central to NEPAT's functionality, Artificial Intelligence (AI) and Machine Learning (ML) algorithms enable the effective management of complex WEFE interlinkages. By providing tailored policy recommendations, NEPAT supports the efficient achievement of nexus-related goals. The overarching aim of NEPAT is to promote stakeholder (SH) collaboration, fostering informed dialogue and cooperative decision-making to address critical WEFE challenges.

Throughout the NXG project, NEPAT has been a key component in the co-creation framework for nexus governance. It has played a crucial role in validating the WEFE policy framework, WEFE goals, targets, and indicators, as well as nexus complexity modelling and Decision Support System (DSS) functionalities. This validation has been carried out through close collaboration with various Work Packages (WPs), Case Studies (CSs), and SHs.

The ultimate objective of NEPAT is to enable end-users to autonomously engage with the NXG research and developments related to the NXG case studies. In this context, the fully integrated NEPAT online platform has been finalized, consolidating all NXG outcomes and providing the broader community with a deeper understanding of WEFE nexus interlinkages, as well as the policy impacts and implications. This is facilitated through a validated, user-friendly Graphical User Interface (GUI) and the Representational State Transfer Application Protocol Interface (REST API).

Additionally, NEPAT has contributed to the NXG project by providing tools for data-sharing. A centralized data repository has been established to store and manage all data generated throughout the project, as outlined in Deliverable D4.2, *Data Lake for Data Sharing*. All data is managed in accordance with the Data Management Plan (DMP) and made publicly accessible via the NXG open repositories, unless restrictions apply. The DMP is documented in Deliverable D4.7, *Data Management Plan – Final Version*.

Deliverable D4.5, titled *Final Version of the Self-Assessment Nexus Engine with Corresponding Validation*, is classified as a "Demonstrator." This document serves as a complement to the digital solutions developed under Task T4.5 *Self-Learning Nexus Assessment Engine*, by providing comprehensive details on the NEPAT interactive platform. It covers the platform's modules, the latest functionality implementations, system deployment, testing, validation, and the application of the co-creation process. Additionally, the document outlines how stakeholder

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

feedback has been incorporated into the final version of the tool. To further enhance user guidance, a tutorial has been developed and shared with the CSs.

The final version of NEPAT is publicly accessible at the following URLs to ensure consistency with both names of the tool: https://slnae.nexogenesis.eu or https://nepat.nexogenesis.eu.

1.1. Disclaimer

The NEPAT has been fully developed and deployed, with all functionalities, modules, and components successfully integrated. However, the tool remains subject to refinements based on inputs from CSs and feedback from SHs until the project's completion.

While the core framework and features are finalized, updates may be implemented to enhance data accuracy, improve model performance, and ensure optimal usability. Any necessary modifications will be systematically incorporated to maintain NEPAT's robustness and alignment with evolving project requirements.

1.2. Links to the ICT4WATER Cluster

WP4 is considered the 'digital' WP in the NXG project. Thus, it is the natural link between the project and the ICT4WATER Cluster[1].

The outcomes of the task T4.5 are specially linked to the ICT4WATER Updated Digital Water Action Plan[2] and its 'Intelligent and smart systems', 'Actor engagement and co-creation', 'Policies', and 'Enabling Data Sharing' Action Groups[3].

With regards to the actions and activities outlined in the Digital Water Action Plan, this deliverable and the SLNAE/NEPAT tool (developed under WP4 in NXG) contributes to the following (all action numbers refer to the Digital Water Action Plan):

- The 'Intelligent and smart systems' action 2 activities 1 & 2: A multi-optimization decision-making framework (the Self-Learning Nexus Engine), based on Deep Reinforcement Learning, is implemented for decision support.
- The 'Intelligent and smart systems' action 5 activities 1 & 2: Uncertainty is taken into account in the integrated complexity science models.
- The 'Actor engagement and co-creation' action 1 activity 1: A public online tool (the SLNAE) is developed.

[1] https://ict4water.eu/

[2] https://ict4water.eu/wp-content/uploads/2023/06/Update-Digital-Water-Action-Plan-V7.pdf

[3] https://ict4water.eu/action-groups/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

- The ‘Actor engagement and co-creation’ action 2 activities 1 & 2: Stakeholders from all nexus sectors are taken into account during the NXG co-creation process for the SLNAE design.
- The ‘Policies’ action 5 activity 1: The SLNAE tool enables policy co-creation.
- The ‘Policies’ action 6 activity 2: The SLNAE tool enables improved management of governance complexity including uncertainty and other factors.
- The ‘Data Sharing’ action activities 1, 2, & 3, by implementing, deploying, offering and maintaining the NXG Semantic Repository service.

1.3. NEPAT: a new name for the SLNAE

Explaining the SLNAE to non-technical audiences proved challenging due to its technical terminology and difficult pronunciation. The term *Self-Learning*—linked to AI and ML—often caused confusion, and the acronym itself was not user-friendly.

To address this, the NXG consortium decided to adopt a clearer, more intuitive name. WP4 led a co-creation process, resulting in the selection of "Nexogenesis - NExus Policy Assessment Tool (NEPAT)" as the preferred alternative. NEPAT is simpler, easier to pronounce, and better reflects the tool's purpose.

For consistency, both SLNAE and NEPAT are valid references to the same tool and are used interchangeably in this document.

1.4. Structure

This document is organized as follows: Section 2 introduces the final version of the NEPAT interactive platform, providing a detailed overview of each module. Section 3 summarizes the NXG data-sharing tool. Section 4 outlines the deployment requirements for the system. Section 5 presents the testing process, with details on interoperability, security, resilience, and scalability. Section 6 discusses the user feedback and the co-creation process, which have guided the finalization of NEPAT. Finally, Section 7 concludes the document by summarizing the main contributions.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

2. SLNAE Interactive Platform

The NEPAT platform is designed as an online tool, enabling users to access it remotely without the need for additional hardware. A standard web browser is the only requirement, ensuring accessibility and ease of use.

The development of the NEPAT Interactive Platform has been guided by the specifications outlined in D4.1 *Self-Learning Nexus Engine Specifications and Technical Design*, as well as feedback provided by CSs and SHs.

The beta version of the tool, which included the WEFE policy framework, WEFE goals, targets, and indicators, nexus complexity modelling, and DSS functionalities, has undergone validation through a co-creation process involving CSs and SHs. This process also helped define future refinement steps, which are currently included in the final version of the platform. Based on stakeholder feedback, the tool was updated and relevant modifications were implemented, including the integration from WP3 of refined System Dynamics Models (SDMs), policies, and goals.

To enhance the platform's usability, the user interface has been improved to provide a more user-friendly experience. Advanced functionalities and detailed analysis options are now accessible in a separate, more complex view designed for expert users, while the basic version has been simplified for general use. Additional refinements include renaming sections with clearer and more descriptive labels, incorporating help features to better explain the various views, graphics, and tool functionalities, and enhancing the design and visualization capabilities. Furthermore, uncertainty and comparison views have been included to support more comprehensive analysis.

Furthermore, efforts have focused on identifying optimal policy packages and training Reinforcement Learning (RL) agents alongside optimization algorithms for the DSS. These optimal policy packages were discussed during fifth workshop with CSs to validate them, refine SDMs, policies, goals, and DSS functionalities, and to further consolidate the overall NEPAT platform. This workshop also contributed to the development of a roadmap for the co-creation framework, providing guidelines for nexus governance design and implementation based on lessons learned from the CSs.

This section provides a detailed overview of the final version of the NEPAT platform, which reflects the feedback gathered through the co-creation process with WPs, CSs, and SHs. The specific inputs from this process that informed the platform’s finalization are elaborated in Section 6.

The platform’s architecture is divided into several key modules, each fulfilling a specific purpose and responsibility (see Figure 1) and these modules include: **Simulation Policy Framework, SLNAE/NEPAT Core Service, Web Service API, SLNAE/NEPAT Database, SLNAE/NEPAT Graphical User Interface, Self-Learning Nexus Engine and AI Algorithms, Decision Support System, and Analytic Engine.**

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

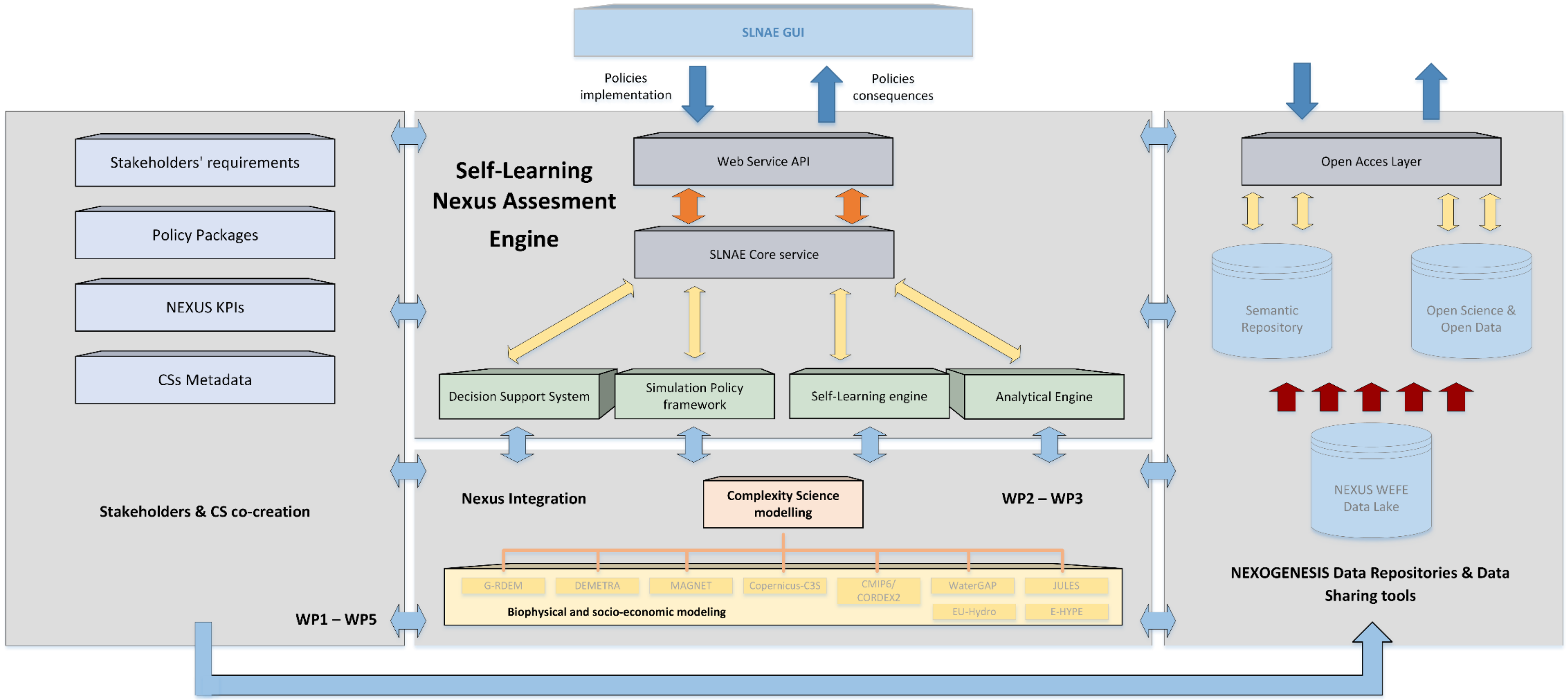


Figure 1. NEPAT platform schema

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Python[4] programming language has been used to develop these modules, with special attention given to component testing. Continuous delivery methodologies have enabled automated and continuous validation, integration, and deployment of new developments. Details on these modules are provided in the corresponding subsections.

The NEPAT modules were initially designed in D4.1. Building upon the original design, this section presents the final versions of these components, highlighting the refinements made to align with the platform's objectives and user needs.

2.1. Simulation Policy Framework

The Simulation Policy Framework (SPF) is a central component of the NXG project, designed to simulate the impacts policy packages on the WEFE nexus. It supports multi-scale analysis across spatial and temporal dimensions and integrates advanced functionalities, including the NXG Decision Support System. Detailed in Deliverable D4.3 *Simulation policy framework*, the SPF represents the culmination of collaborative efforts among all technical WPs within the NXG co-creation framework.

Key contributions from various WPs include policy and goal identification by WP1, WP5, CSs, and SHs, as well as the collection and development of climate and socio-economic data by WP2. WP3 utilized this data to create System Dynamic Models, which incorporate policies, goals, and the WEFE footprint. WP4 integrated these SDMs into the SLNAE (or NEPAT) system, enabling accessibility through the user interface and its REST API.

The SPF allows stakeholders to evaluate the impact of policy packages on the five NXG case studies by running simulations through the NEPAT platform. Users can analyze configurations of policy packages and assess their performance against case study goals or the Nexus footprint index. As the NEPAT system undergoes continuous updates based on inputs from CSs, SHs, and data providers, all SPF modules—including SDMs, policies, goals, and the UI—are subject to ongoing validation and refinement. Recent co-creation and validations workshops have led to updates in data and SDMs, which have been automated to facilitate eventual future modifications.

The process from an SDM to running a simulation in the tool involves several interconnected components, including the Data Manager, SDM Translator, SDM Manager, SPF Core Service, SPF Web Service API, and the user interface (see Figure 2). While these components work together, the SDM Translator and SDM Manager are unique to the SPF, as they handle the translation and management of SDMs within the framework. The SDM Translator, converts SDMs from STELLA format into Python, integrating them into the NEPAT system. These Python-based SDMs are then managed by the SDM Manager, which executes simulations based on user-defined parameters, such as the case study, reference scenario, and policy

[4] https://www.python.org/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

package. The SPF generates results that reflect the nexus impacts of the applied policy package, providing valuable insights to inform decision-making.

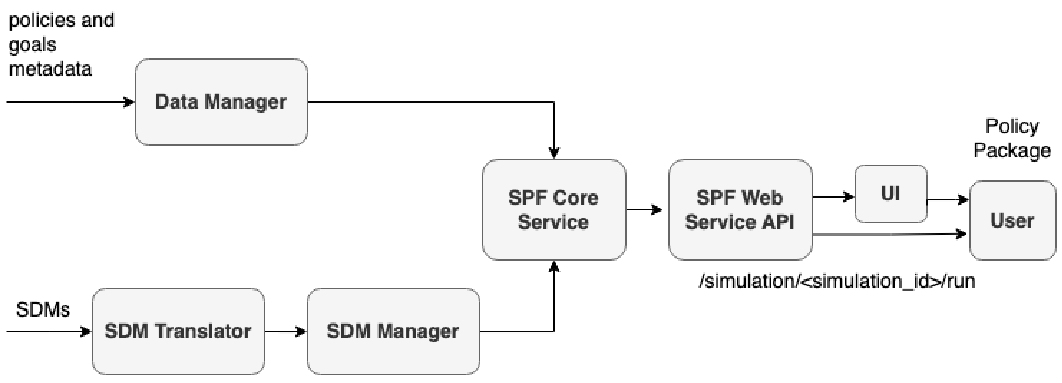


Figure 2. Simulation policy framework pipeline

For further details, refer to Deliverable D4.3 *Simulation policy framework*.

2.2. NEPAT Core Service

The SLNAE (or NEPAT) Core Service serves as the central hub of the SLNAE (or NEPAT), orchestrating communication across various modules, monitoring the status of services, and providing critical functionalities that ensure smooth operation. Its architecture is purposefully designed to meet the platform's current operational demands while maintaining flexibility for future scalability and enhancements.

At the heart of the Core Service is its seamless integration with the Web Service API module, which enables the validation of all incoming communications to ensure secure and reliable interactions. To assist the API module, the Core Service includes the following components: the Data Manager, the Text Translator, the Authentication and Authorization Module, the Simulation Utilities Module and the DSS Manager.

Data Manager

The Data Manger plays a crucial role in ensuring the persistence and retrieval of data from the SLNAE Data Repositories. It automates processes to integrate outputs from other WPs into the platform's logic. This integration is achieved through its connection to the Data Sharing Tool, where it ingests and processes key information such as policies, goals, targets, indicators, case study data, and SDMs. By working in close collaboration with the Text Translator and the database module, the Data Manager ensures that all data is accurately loaded and structured, enabling seamless operation of the platform’s functionalities.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Text Translator

The Text Translator is an integral module of the Core Service, specifically designed to process and translate textual data stored in the database. It functions in close collaboration with the Data Manager, processing essential information such as policy-related data, goals, and other critical inputs. After completing the translation, the processed data is seamlessly stored in the database, ensuring accessibility across the platform. The diagram presented in Figure 3 provides a detailed illustration of the interaction pipeline and the connections between the Text Translator, the Data Manager, and the database.

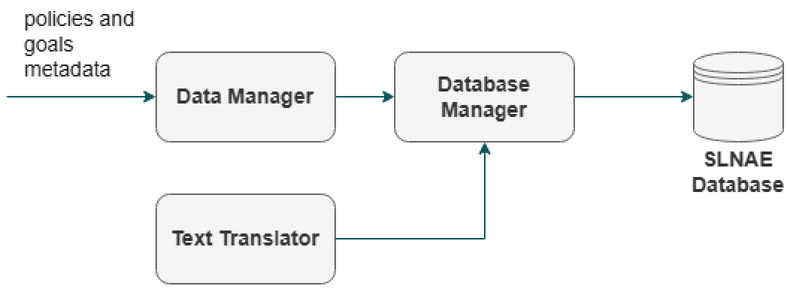


Figure 3. Interaction pipeline between Text Translator, Data Manager, and database

Authentication and Authorization Module

The Authentication and Authorization Module ensures secure communication across the platform and effectively manages user access. Given that the Web API, detailed in following sections, has been developed using the framework FastAPI, the authentication module was implemented following its recommended best practices. Specifically, the python-jose[5] library was used for JSON Web Token (JWT) management, while passlib[6] was utilized for password handling.

This module has three main responsibilities:

- **JWT token generation:** After validating user credentials, the module generates a JWT token with a 48-hour expiration, granting access to all NEPAT functionalities.
- **JWT token validation:** For user authentication, the module decodes and verifies a given JWT token, ensuring its validity, expiration, and integrity before granting access.
- **Password encryption and verification:** For users registering and logging in with an email and password, the module securely encrypts passwords and verifies credentials.

The JWT token generation process relies on prior validation of user credentials by the Web Service API. Once authenticated, the primary task is to generate and return an encrypted token based on the provided username. This process involves the current timestamp, a private key, and the username to create a secure access token, as illustrated in Figure 4.

[5] https://pypi.org/project/python-jose/
[6] https://pypi.org/project/passlib/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

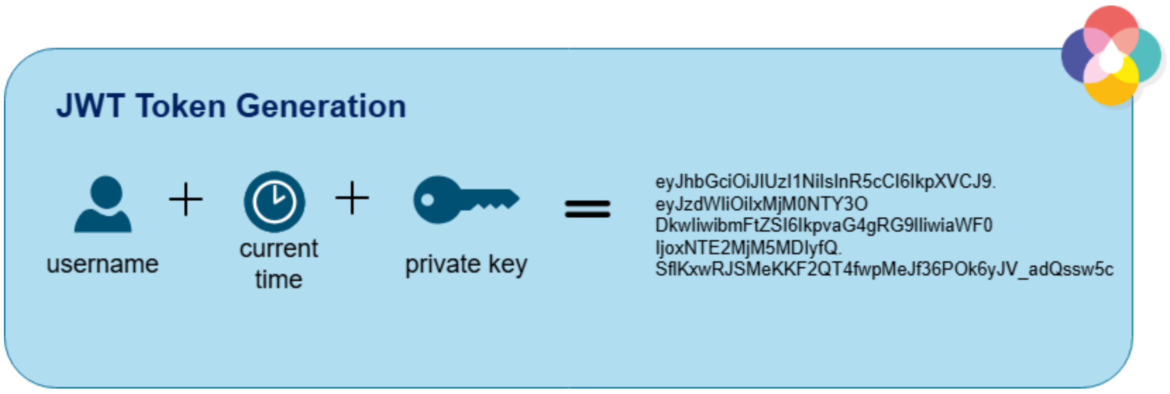

JWT Token Generation
username
current time
private key
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJzdWIiOiIxMjM0NTY3O
DkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0
IjoxNTE2MjM5MDIyfQ.
SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c

Figure 4.JWT token generation process

The implemented JWT token validation is a method that uses the same private key to decrypt the token and raises an HTTP exception in case of any errors. This method serves as a crucial dependency for the Web Service API, ensuring it is the first step executed upon receiving a request. The flow diagram (Figure 5) outlines the entire validation process, detailing the potential errors that may arise during user authentication.

The password encryption and validation functionalities are implemented in a lightweight component that leverages Passlib to securely encrypt and verify passwords using a private key.

Simulation Utilities Module

The Simulation Utilities Module encompasses all the necessary methods for the API to execute requests related to simulation management. Specifically, it is responsible for the following tasks:

- **SDM variable list retrieval:** when required by the Web Service API, this method fetches the list of variable names for a specific case study and scenario.
- **SDM execution:** this functionality runs the specific SDM translation given a run configuration and policies by the API.
- **Footprint calculations:** after the SDM execution, footprint values are computed based on the run results.

In addition to the translation of SDMs detailed in deliverable D4.3 *Simulation policy framework*, the SLNAE Platform also includes auxiliary methods for generating supporting structures, such as retrieving the list of SDM variables, associated units, and other related metadata. Specifically, for the list of variables, the module, given a case study ID and scenario ID, is responsible for fetching the corresponding JSON file containing the variable list and related information.

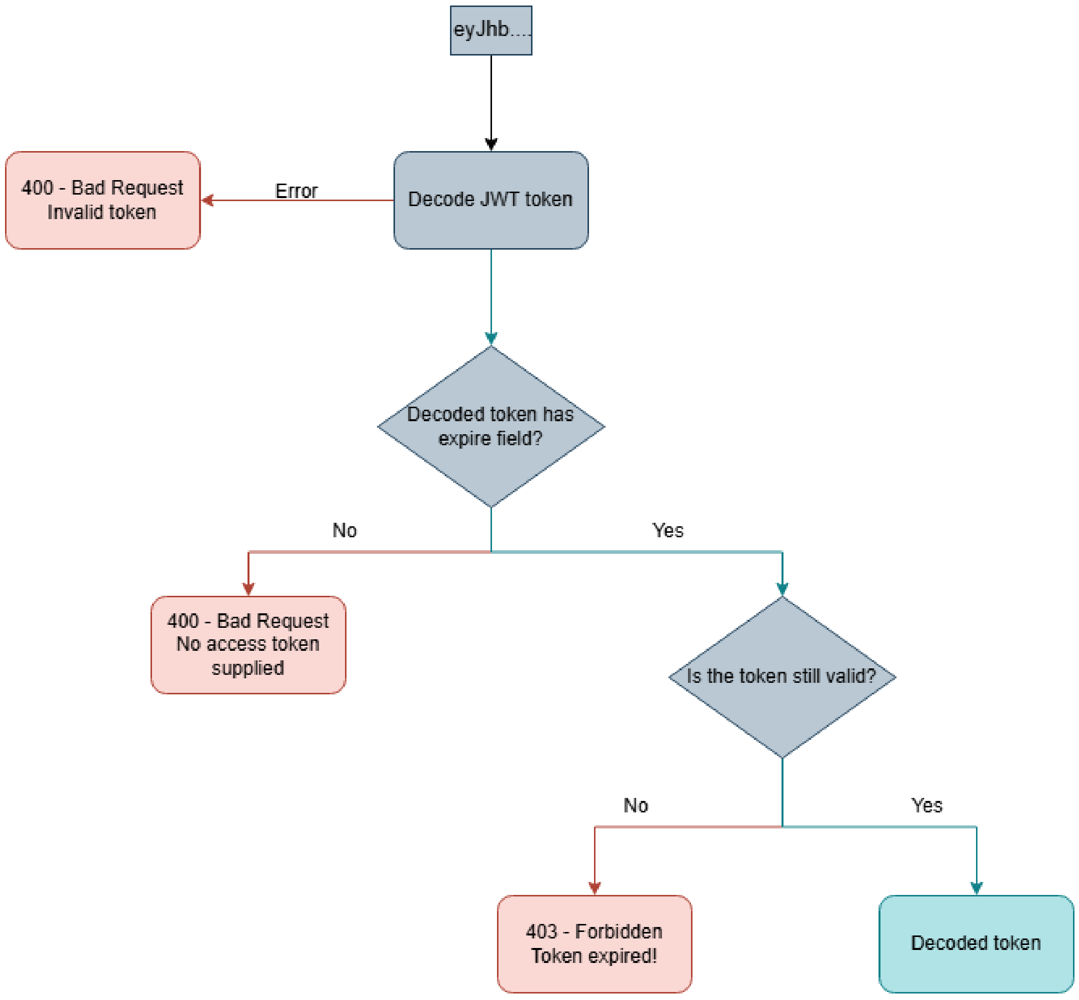


Figure 5. JWT token validation flux

The policy simulation process, as outlined in D4.3, follows the pipeline shown in Figure 2. Within this pipeline, this module is responsible for the portion allocated to the SPF Core service. The pipeline describes the process of running a simulation, which has the following configurable parameters:

- **Policies:** List of policies to activate during the simulation.
- **Number of runs:** Defines the number of times the selected SDM is executed. If the simulation runs more than once, a stochastic simulation is performed, meaning that the SDM Manager will run the original translation file. Otherwise, an average translation file is used, attaining always deterministic results. These translations are detailed in D4.3.
- **Baseline execution:** If enabled, an additional SDM run is performed using an empty policy package. This provides reference scenario results for comparison alongside the policy package run.

These parameters are set through the Web Service API and passed along the pipeline components to calculate the simulation results. As outlined in D4.3, the SDM Manager executes the SDM when provided with the policy, the specific SDM to run, and the random mode. The SPF Core Service, which is responsibility of the Core Service, manages the execution of the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

SDM Manager with the correct parameters and number of runs. For example, when baseline execution is required, the SPF Core service runs the SDM Manager twice: once with the policy package and once without. Similarly, when multiple runs are needed, the module runs the SDM Manager the times required and calculates the three quartiles for each SDM variable's random executions.

With regards to the footprint calculations, what this module does is integrating the normalization function provided by WP3 and detailed in D3.7 *Final report on the WEFE Nexus Index methodology and visualisation*. These calculations are run after the execution of the SDMs using the results computed.

DSS Manager

The DSS Manager is a critical component that facilitates seamless interaction between the trained AI models and the Web Service API. As outlined in Figure 1, the system architecture places the Core Service as an intermediary layer between the DSS Manager and the Web Service API. The Core Service plays a pivotal role in orchestrating multiple recommendation engines, which are further detailed in Section 2.7. Its primary function is to aggregate, process, and harmonize the outputs from these diverse recommendation models, ensuring consistency and interpretability. Given the inputs received by the API, the Core Service applies filtering mechanisms to refine and select the most relevant recommendations before unifying them into a single structured response. This response is then formatted for visualization and interaction within the GUI.

2.3. Web Service API

The Web Service API serves as a bridge between the SLNAE (or NEPAT) GUI and the SLNAE (or NEPAT) Core Service. Its primary function is to handle requests from the GUI and provide responses by leveraging both the SLNAE Database and the Core Service.

Developed using the FastAPI[7] web framework, the SLNAE API is designed with REST principles, offering scalability through its stateless architecture. FastAPI's built-in support for asynchronous programming, automatic documentation generation, and robust data validation ensures exceptional performance and user-friendliness. Furthermore, the API aligns with industry standards by being fully compatible with OpenAPI (formerly Swagger) and JSON Schema, facilitating seamless integration and maintaining a consistent, standards-based approach to interaction.

The decision for using FastAPI over the initial idea of Flask was due to FastAPI's advanced design and better performance considering different metrics. Specifically, a comparative analysis of Python frameworks was made to determine the most suitable option for meeting the

[7] https://fastapi.tiangolo.com/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

project's specific requirements. This analysis, summarized in Table 1, highlights the decision-making process, with green indicating high quality, yellow representing moderate quality, and orange signifying lower quality.

Table 1. Comparative analysis of Python Frameworks for the Web Service API

Concept	Fast API	Flask	Django
Performance	🟢	🟡	🟠
API Documentation	🟢	🟡	🟡
Concurrency and Async suport	🟢	🟠	🟠
Security & privacy	🟠	🟡	🟢
Community & Documentation	🟡	🟢	🟢
Scalability	🟡	🟢	🟢
Extensions	🟡	🟢	🟢
Flexibility	🟡	🟢	🟡
Domains of use	Speed + monitoring tools	AI + Prototypes	CRM+ long apps
Development Curve	🟢	🟢	🟡
Monitoring Tools	🟢	🟠	🟠
Resource	🟢	🟢	🟢
Testing	🟢	🟢	🟢

To guarantee optimal performance, the API is deployed using Gunicorn[8], a robust WSGI HTTP server. It is configured with six workers, enabling it to efficiently handle multiple requests concurrently.

The API provides all the necessary endpoints to support the functionalities available in the SLNAE GUI. These endpoints are grouped into two primary categories: **Authentication** and **Simulation Management**. Below is a detailed explanation of their usage and references to the annexes with the API description. However, for a more visual interaction with the API, there's also the documentation online in https://nepat.nexogenesis.eu/api/docs.

2.3.1. Authentication

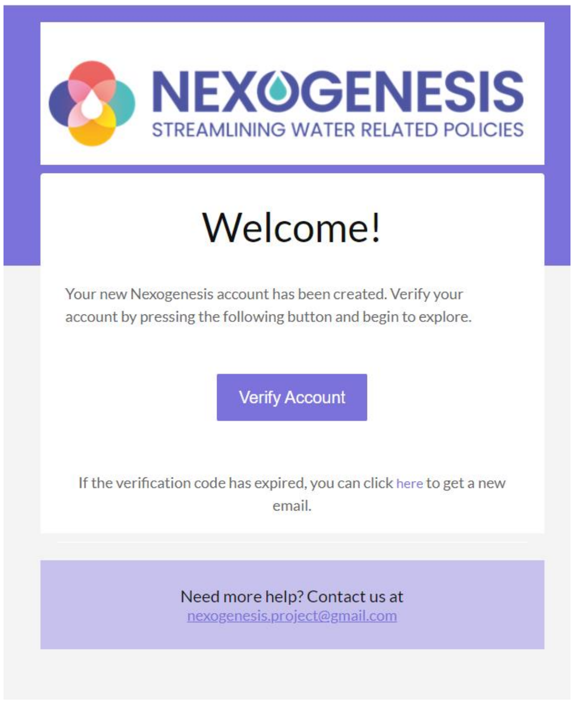
Authentication is required to use the tool functionalities. The API offers three forms of user authentication: common user-password authentication, guest login and login with Google. Each method is detailed in this section. For more details on the endpoint's specifications, refer to Annex I: Authentication endpoints.

[8] https://gunicorn.org/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Common user-password authentication

To use this option, the user must first register using a new username, email and password via the /auth/signup endpoint. After registering, the user will receive a verification email containing a "Verify Account" button as seen in Figure 6, which will make a POST request to /auth/very-email.


NEXOGENESIS
STREAMLINING WATER RELATED POLICIES

Welcome!

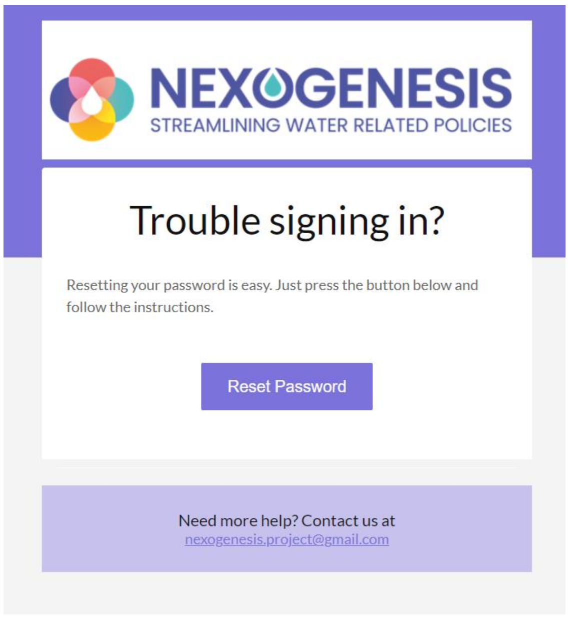
Your new Nexogenesis account has been created. Verify your account by pressing the following button and begin to explore.

If the verification code has expired, you can click here to get a new email.

Need more help? Contact us at nexogenesis.project@gmail.com

Figure 6. Account verification email

Clicking this button is necessary to activate the account. Once verified, the user can log in using their registered credentials in the /auth/login endpoint, which will provide a JWT token granting access to all functionalities.


NEXOGENESIS
STREAMLINING WATER RELATED POLICIES

Trouble signing in?

Resetting your password is easy. Just press the button below and follow the instructions.

Need more help? Contact us at nexogenesis.project@gmail.com

Figure 7. Password recovery email

In case the user forgets their credentials, a password recovery mechanism is available. This can be used through a POST request to /auth/password-recovery, including the user's email address and the desired new password. Like with the account registration, a confirmation email (Figure 7) is sent to verify the request by clicking on the button "Reset Password". By using this verification email, only the legitimate account owner can complete the recovery process.

Guest Login

This method allows for a quick and straightforward way to access the tool without providing any personal credentials. The user simply sends a POST request to the /auth/login endpoint and a JWT token will be issued, valid for 48 hours. However, progress made during the session will be lost once the token expires.

Login with Google

This method leverages Google account credentials for authentication. The user sends a request to the /auth/google-login endpoint and, upon successful login, receives a JWT token that grants access to the tool's features.

2.3.2. Simulation Management

Once authenticated, users can access the primary functionalities of the API: creating and managing simulations. The API distinguishes between two key components of the simulation process:

- **Simulation Wrappers:** These represent what is referred to as simulations or simulation comparisons in the GUI. They act as containers for simulations and are always tied to the same case study.
- **Simulations:** They represent a specific case study and scenario selection combined with a defined set of policies. In the GUI, these are referred to as policy packages.

To support the GUI functionalities, different methods are implemented for simulation wrappers and simulations in the API.

To manage simulation wrappers, users can interact with the /simulation-wrappers endpoint. A GET request to this endpoint retrieves the user's existing simulation wrappers, which populate the list of simulations in the NEPAT tool's management view. Creating a new simulation wrapper is done via a POST request to the same endpoint, allowing users to define one or more simulations within the wrapper. This process corresponds to the "New" and "Import" buttons in the GUI. Users can also open or delete individual wrappers by sending GET or DELETE requests to /simulation-wrapper/{id}. Additionally, the simulations contained within a wrapper

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

can be updated using a PUT request to /simulation-wrappers, aligning with the GUI's "Save" and “Edit” functionalities.

Simulations are managed through the /simulations/ endpoint. A POST request to this endpoint enables the creation of a new simulation by specifying the desired case study and scenario. This mirrors the "Add Policy Package" feature in the GUI. Once created, simulations can be executed by sending a POST request to /simulations/{id}/run, simulating the policy package effects for the selected case study and scenario. For decision support, users can access recommendations via the /simulations/{id}/advice endpoint. This provides insights generated by the DSS based on the defined case study, scenario, and policy package.

For more details on the endpoint’s specifications, refer to Annex I: Simulation endpoints.

2.3.3. Other functionalities

Beyond authentication, simulation wrappers, and simulations, the API includes other functionalities to support critical features of the NEPAT. These functionalities primarily involve accessing case study data, customizing scenarios, and providing feedback.

The case study endpoints provide essential data for selecting and customizing case studies and scenarios. A GET request to /case-studies/ retrieves information about all available case studies, which corresponds to the wizard that appears in the GUI when the user clicks "New." This data allows users to explore the details of each case study before making a selection.

For users who want to define a custom goal, the API provides access to scenario-specific indicators. By sending a GET request to /case-studies/{id}/scenarios/{scenario_id}/indicators, users can retrieve the list of SDM variables associated with a particular case study and scenario combination. These indicators are displayed in the GUI when users opt to consider a new custom goal, facilitating tailored scenario configurations.

The API also includes functionality for reporting feedback. Users can send a POST request to /bug-reports to submit issues, bugs, or general feedback about the tool. This feature ensures that user concerns are documented and addressed, contributing to ongoing improvements of the SLNAE platform.

Further details on the reviewed endpoints are available in Annex I: Case study endpoints and Feedback reports endpoint.

NEXOGENESIS STREAMLINING WATER RELATED POLICIES

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

2.4. SLNAE database

As outlined in deliverable D4.1, all the information of the SLNAE entities is persisted by using a SQL database. Specifically, PostgreSQL, a widely recognized and robust open-source object-relational database system, is used for its implementation. The management and interaction with the SLNAE backend is handled by the ORM, which implements CRUD methods for the database entities.

The database architecture consists of a single instance, as the anticipated traffic levels do not necessitate a distributed setup. This configuration efficiently handles several hundreds of queries per second under typical conditions, ensuring reliability and responsiveness.

To maintain security, user passwords are encrypted before storage, and database access is restricted to authenticated users with valid credentials. Access is further limited to users within the organization who connect via a secure VPN, adding an additional layer of protection. Additionally, database backups are performed weekly to safeguard against data loss.

The database structure is shown in Figure 8 using UML (Unified Model Language), offering a comprehensive overview of the database schema and the relationships between entities. This schema includes all the information on case study scenarios, policies, goals, and related data required to provide all the information displayed in the GUI. For performance considerations, simulation results are excluded from the database; instead, they are computed dynamically, with commonly used values cached where possible, reducing storage and query load.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

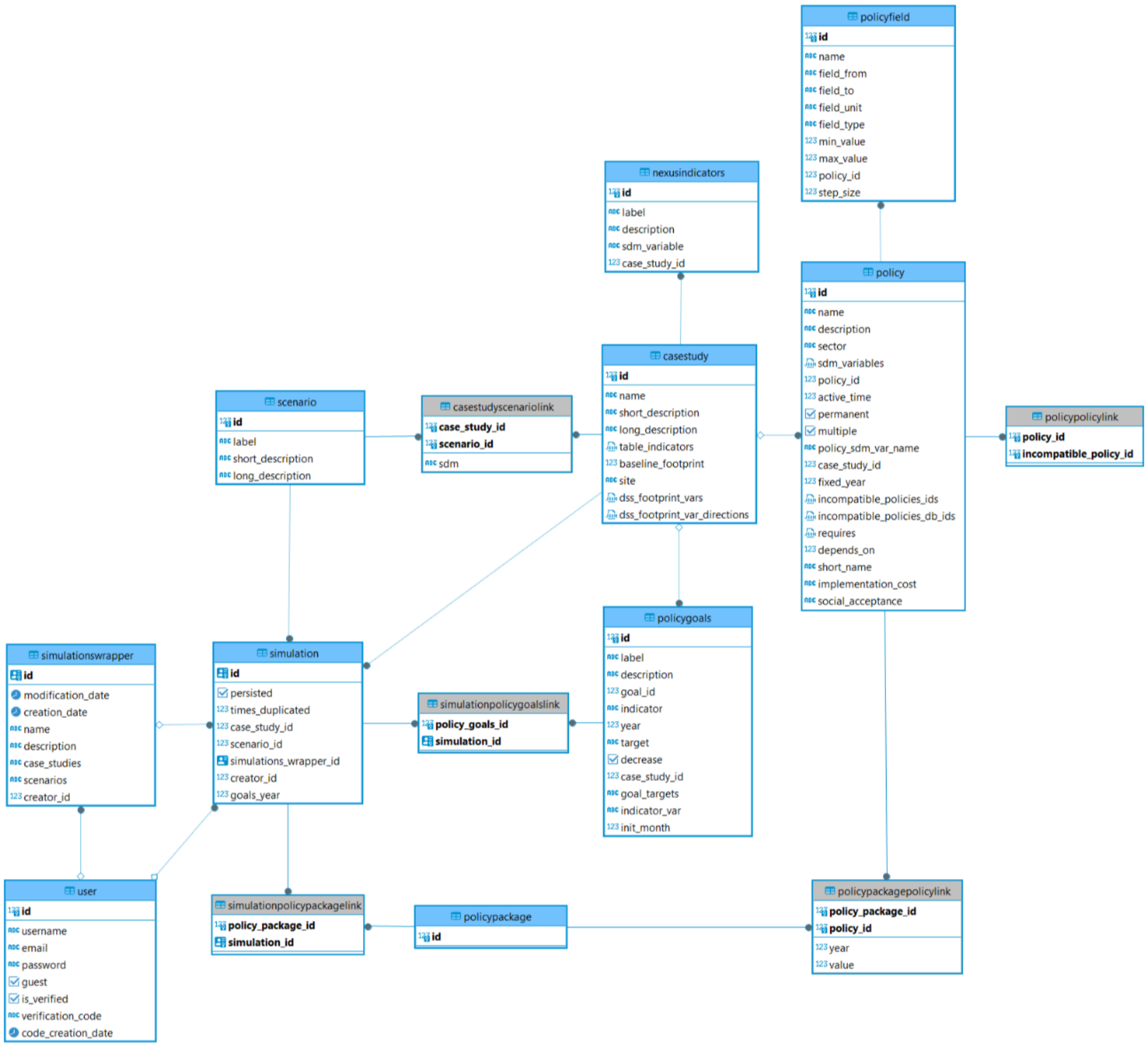


Figure 8. SLNAE Database UML

2.5. SLNAE GUI

The SLNAE Graphical User Interface (GUI) functions as the primary interface of the platform, providing users with the ability to interact with the research and outcomes of the NXG project. It equips users with the essential tools to conduct WEFE simulations, applying Nexus policy instruments across the five NXG CSs. Moreover, the interface facilitates the analysis of the impacts on the interconnected sectors of the nexus, enabling users to evaluate policy goals and assess the achievement of WEFE footprint indicators. This ultimately assists users in identifying the most effective policy packages for each scenario.

The final version of NEPAT consists of three main views:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

1. **Login View**: Allows users to log into their personal accounts, enabling them to save and access previous work.
2. **Management View**: Enables users to create new simulations and retrieve saved simulations from both current and past sessions.
3. **Simulations View**: Supports the simulation of various policy scenarios, facilitates the analysis of policy impacts, and allows users to compare different scenarios.

To enhance the user experience and ensure seamless navigation, NEPAT includes a help functionality embedded within each view. This feature provides predefined explanations of how each view and its associated functionalities work, offering users clear guidance on using the tool, understanding the graphics, and interpreting the results.

These main NEPAT views, along with the help functionality, are illustrated in Figure 9.

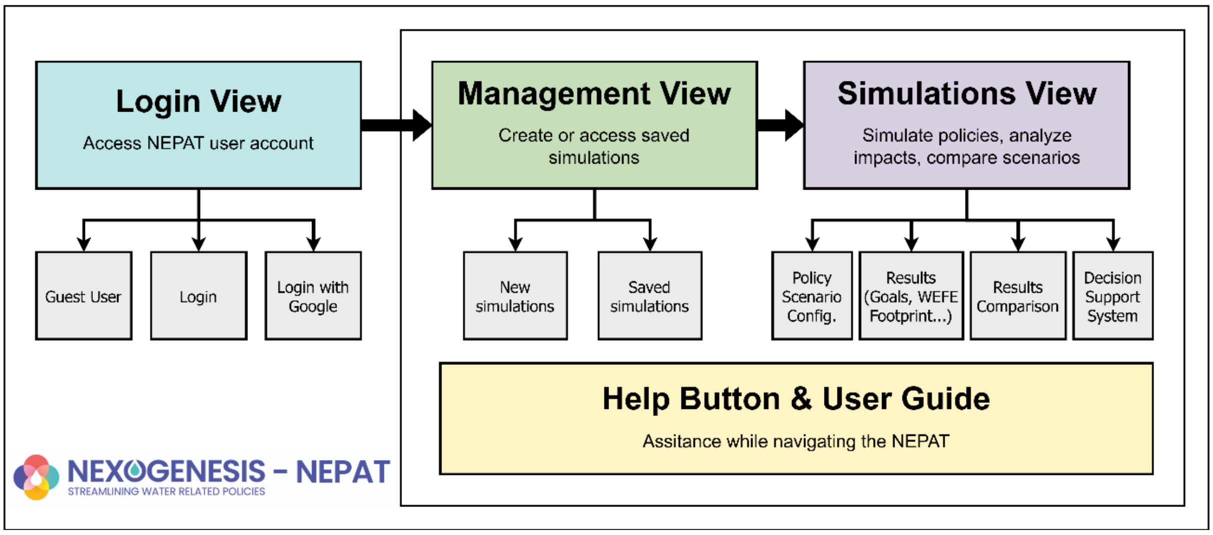

Login View
Access NEPAT user account
Guest User
Login
Login with Google
Management View
Create or access saved simulations
New simulations
Saved simulations
Simulations View
Simulate policies, analyze impacts, compare scenarios
Policy Scenario Config.
Results (Goals, WEFE Footprint...)
Results Comparison
Decision Support System
Help Button & User Guide
Assitance while navigating the NEPAT
NEXOGENESIS – NEPAT
STREAMLINING WATER RELATED POLICIES

Figure 9. NEPAT platform views and embedded guidance system

As detailed in D4.1, two primary user groups were identified: **strategic users** and **technical users**. These groups were defined based on their distinct requirements for information presentation and system interaction.

The platform's GUI is designed to address the needs of both user types through the integration of two interconnected views. This dual-view approach ensures accessibility and usability for all users, allowing seamless transitions between views. Advanced functionalities can be activated or deactivated as needed, tailoring the interface to the specific needs and expertise of the user.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

2.5.1. Advanced functionalities

The NEPAT offers a range of functionalities tailored to meet the diverse needs of its users. These capabilities are divided into two primary views, designed to accommodate both strategic and technical user requirements.

Strategic View

The Strategic View is designed for users who prioritize high-level, easily interpretable insights to guide decision-making. This view targets policymakers, business leaders, investors, NGOs, and other stakeholders who rely on summarized information for strategic decisions. Key features include:

- **Simplified information**: Concise, synthesized data presented through colourful graphics, diagrams, and minimal text.
- **Clear visualization**: Straightforward presentation of simulation results, indicator evaluations, and comparisons to support effective decision-making.

Technical View

By enabling advanced features in the user settings, users can access tools designed for in-depth analysis and scientific reasoning. To activate these features, the user should navigate to *User > Settings > Show advanced functionalities*, as illustrated in Figure 10.

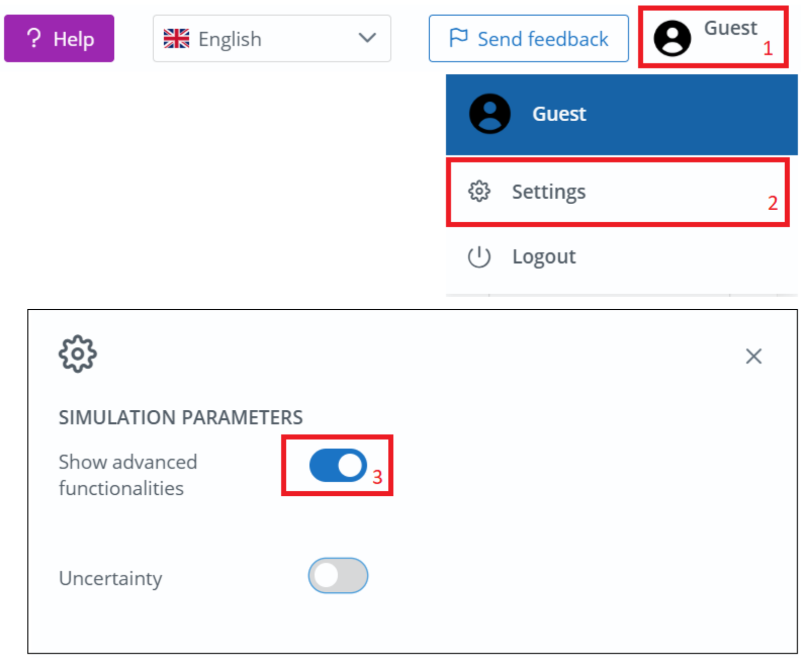


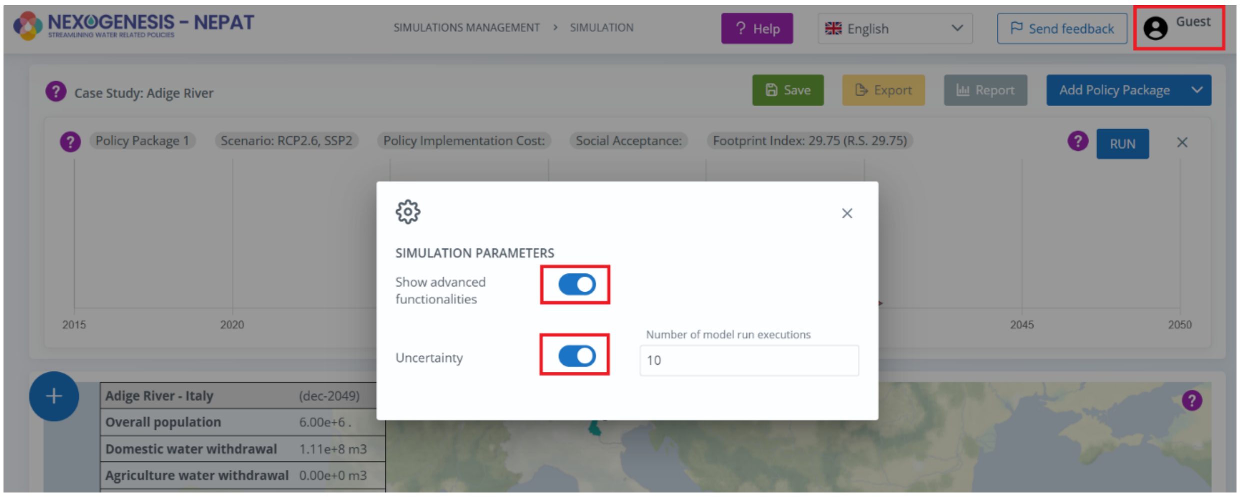
Figure 10. Enabling advanced functionalities in user settings

The Technical View is aimed at individuals and organizations with technical expertise, such as scientists, academics, government agencies, consultancy firms, NGOs, and civil society organizations. Its core functionalities include detailed policy impact analysis and customizable features.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Detailed policy impact analysis

- **Sunburst graph:** A hierarchical visualization tool that illustrates policy impacts and the relationships between variables, providing a comprehensive understanding of interconnected effects across the WEFE sectors.
- **Stochastic simulations**: Introduces uncertainty into policy execution by incorporating stochastic variables. Users can define the number of model executions per run. Results for stochastic variables are presented using quartiles (Q1, Q2, Q3) to illustrate variability and outcomes.
  Uncertainty can be enabled after activating advanced functionalities by clicking the *Uncertainty* button and specifying the *Number of model run executions* parameter (see Figure 11). Note that a higher number of runs will increase execution time.


Case Study: Adige River

Policy Package 1 | Scenario: RCP2.6, SSP2 | Policy Implementation Cost: | Social Acceptance: | Footprint Index: 29.75 (R.S. 29.75)

SIMULATION PARAMETERS

Show advanced functionalities: on

Uncertainty: on

Number of model run executions: 10

Adige River - Italy	(dec-2049)
Overall population	6.00e+6
Domestic water withdrawal	1.11e+8 m3
Agriculture water withdrawal	0.00e+0 m3

Figure 11. Activating advanced functionalities with uncertainty in the NEPAT

Customizable features

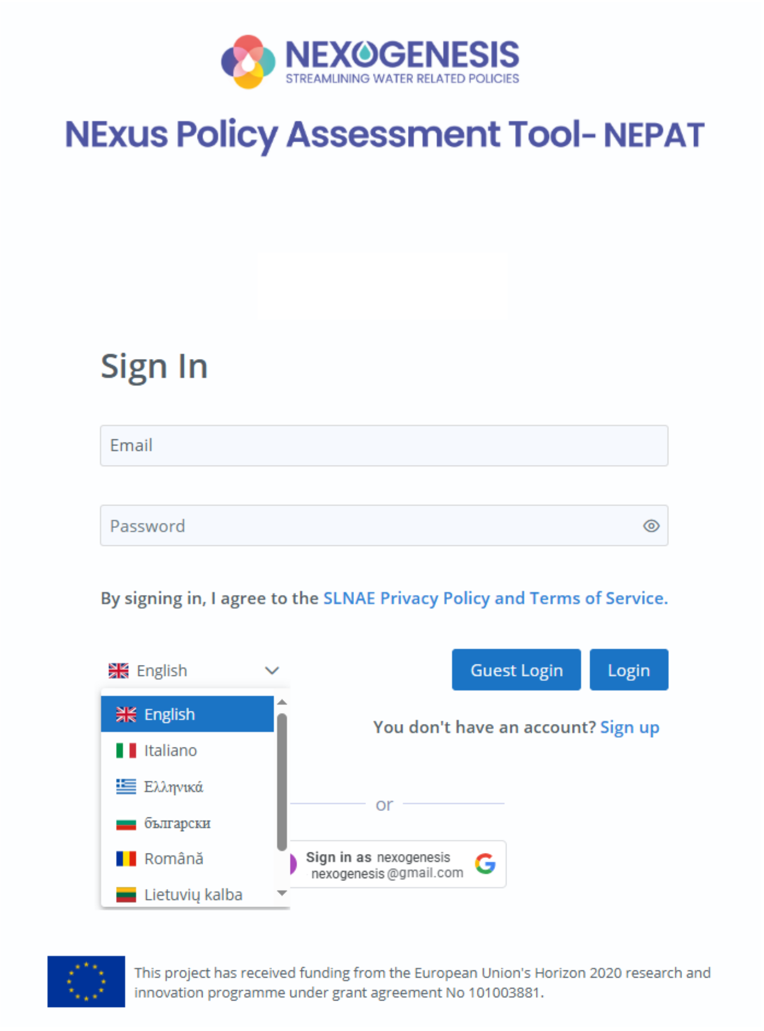
- **Custom goals:** Enables users to define specific goals during simulation configuration.
- **Custom policy implementation years:** Allows users to determine the year of application for any policy, providing greater flexibility in customizing policy instruments. This feature has been developed exclusively for the Inkomati CS.
- **Qualitative indicators:** Enables customization of implementation costs and social acceptance for policy instruments. Default values are based on intuitive estimates rather than rigorous scientific analyses.
- **AI-based recommendations:** Provides tailored recommendations with options to:
  - Build on existing policy packages.
  - Limit the number of recommended policy instruments.
  - Restrict recommendations to specific sectors, ensuring relevance and precision.

The advanced functionalities available in each view are described in more detail in the corresponding sections.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

2.5.2. Login View

To access the NEPAT platform, visit https://nepat.nexogenesis.eu/#/login. NEPAT offers a flexible and user-friendly login system with three primary access options: enter as a guest, create an account, and log in with Google (see Figure 12). The login page also allows users to customize the platform interface by selecting their preferred language.


NEXOGENESIS
STREAMLINING WATER RELATED POLICIES
NExus Policy Assessment Tool- NEPAT

Sign In

Email

Password

By signing in, I agree to the SLNAE Privacy Policy and Terms of Service.

English
English
Italiano
Ελληνικά
български
Română
Lietuvių kalba

Guest Login
Login

You don't have an account? Sign up

or

Sign in as nexogenesis
nexogenesis@gmail.com

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881.

Figure 12. Login view of the NEPAT platform

By logging in, users automatically accept the SLNAE Privacy Policy and Terms of Service, which are binding upon access. A direct link to the full *Terms and Conditions of Use* is available on the login page, enabling users to review them in detail before proceeding.

Common user-password authentication

By selecting the Sign up option, users can register for an account, enabling the ability to save simulations. This feature is particularly valuable for users who need to preserve their work for ongoing or long-term projects.

Guest Login

Users who do not wish to create an account can explore NEPAT as guests. This option provides full access to the platform's functionalities, including the ability to save simulations during the session. However, all data will be lost upon logging out unless it has been exported. Guest access is ideal for users conducting an initial evaluation or working on temporary projects.

Login with Google

For added convenience, users can log in using their Google accounts by clicking the *Login with Google* button. This option eliminates the need for manual registration, providing quick and secure access.

2.5.3. Management View

The Management View in NEPAT serves as the main interface for handling simulations. Upon logging in, users are directed to this view, where they can create new simulations or manage existing ones.

The table displayed in the Management View shows all simulations saved from previous sessions, as well as any new simulations created and saved during the current session. If the user logs in as a guest, this table will initially be empty. Figure 13 illustrates this view, showcasing the table populated with some saved simulations.

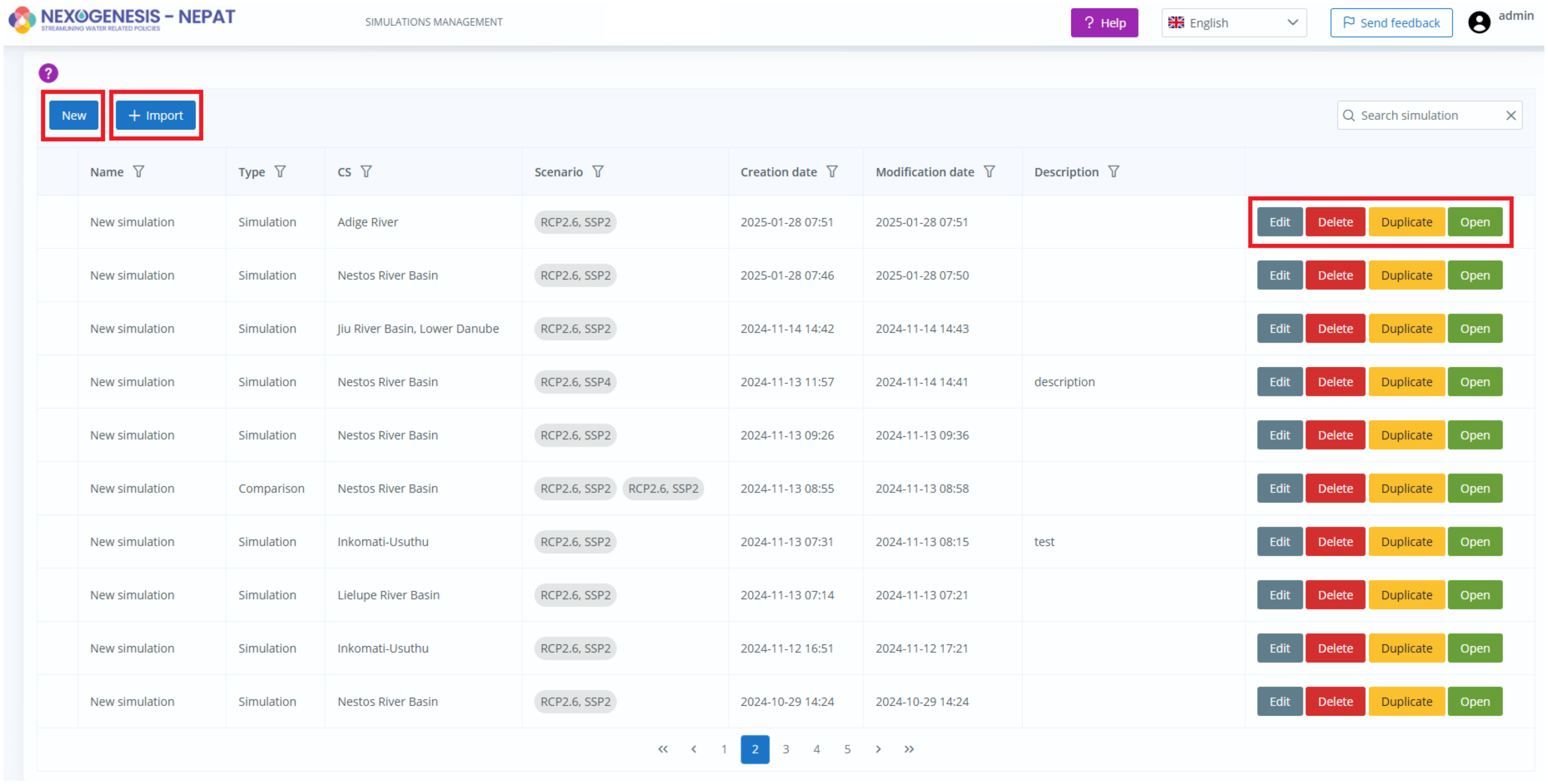
For each simulation, the following details are displayed:

- **Name**: User-defined simulation name.
- **Type**: Indicates whether it is an individual simulation or a comparison of multiple simulations.
- **CS**: The associated case study.
- **Scenario**: The selected scenario or multiple reference scenarios for comparisons.
- **Timestamps**: Creation and modification dates.
- **Description**: User-defined description.

Users can *Delete* simulations or *Open* them to continue working on them. To facilitate navigation when dealing with a large number of saved simulations, the table includes a filtering feature that allows sorting by any column. Additionally, a search bar is provided to quickly locate specific simulations.

Users can *Duplicate* an existing simulation to preserve the original while making modifications. To rename or update the description of a saved simulation, users can click the *Edit* button, which provides an option to change its name or description. To create a new simulation, users can click the *New* button to launch the simulation wizard or *Import* an existing simulation. If users require assistance, the interface includes help buttons marked with a ?, which provide detailed information when clicked. All these buttons are highlighted in Figure 13.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881



Name	Type	CS	Scenario	Creation date	Modification date	Description	
New simulation	Simulation	Adige River	RCP2.6, SSP2	2025-01-28 07:51	2025-01-28 07:51		Edit Delete Duplicate Open
New simulation	Simulation	Nestos River Basin	RCP2.6, SSP2	2025-01-28 07:46	2025-01-28 07:50		Edit Delete Duplicate Open
New simulation	Simulation	Jiu River Basin, Lower Danube	RCP2.6, SSP2	2024-11-14 14:42	2024-11-14 14:43		Edit Delete Duplicate Open
New simulation	Simulation	Nestos River Basin	RCP2.6, SSP4	2024-11-13 11:57	2024-11-14 14:41	description	Edit Delete Duplicate Open
New simulation	Simulation	Nestos River Basin	RCP2.6, SSP2	2024-11-13 09:26	2024-11-13 09:36		Edit Delete Duplicate Open
New simulation	Comparison	Nestos River Basin	RCP2.6, SSP2 RCP2.6, SSP2	2024-11-13 08:55	2024-11-13 08:58		Edit Delete Duplicate Open
New simulation	Simulation	Inkomati-Usuthu	RCP2.6, SSP2	2024-11-13 07:31	2024-11-13 08:15	test	Edit Delete Duplicate Open
New simulation	Simulation	Lielupe River Basin	RCP2.6, SSP2	2024-11-13 07:14	2024-11-13 07:21		Edit Delete Duplicate Open
New simulation	Simulation	Inkomati-Usuthu	RCP2.6, SSP2	2024-11-12 16:51	2024-11-12 17:21		Edit Delete Duplicate Open
New simulation	Simulation	Nestos River Basin	RCP2.6, SSP2	2024-10-29 14:24	2024-10-29 14:24		Edit Delete Duplicate Open

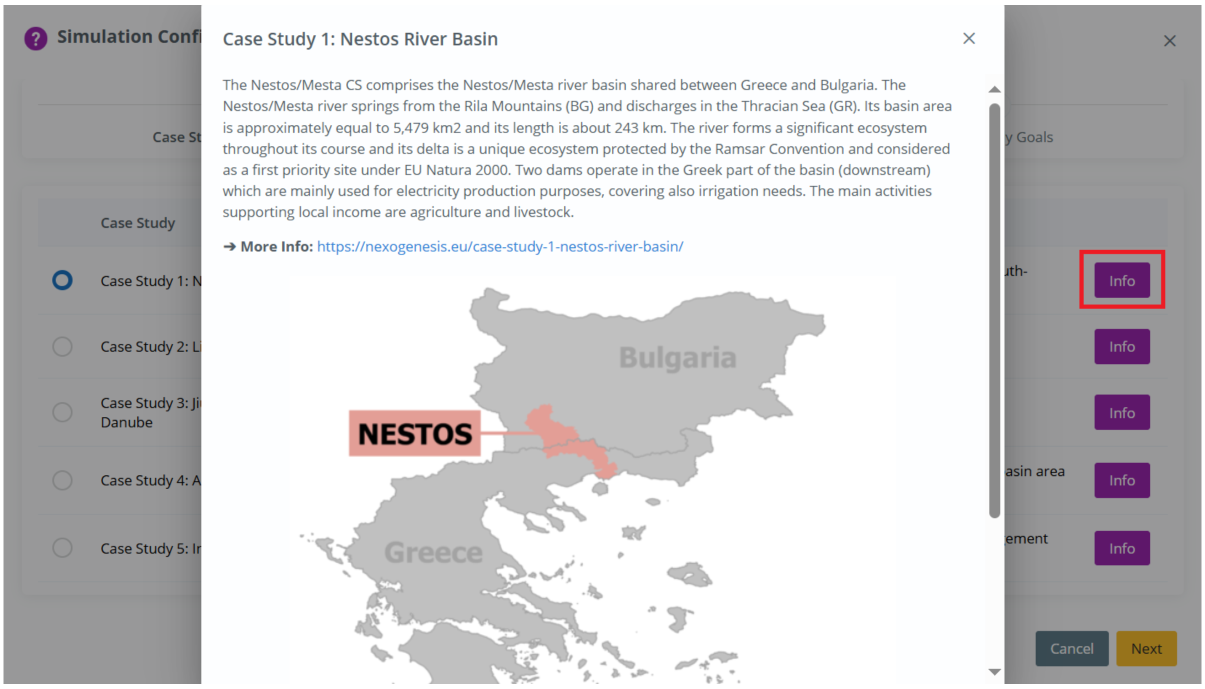
Figure 13. Management view of the NEPAT platform

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Configuring a Simulation

Clicking *New* opens the simulation wizard, guiding users through a three-step setup:

1. **Select the Case Study:** Users can choose from a range of case studies, each accompanied by additional information accessible via an *info* button (see Figure 14).



Case Study 1: Nestos River Basin

The Nestos/Mesta CS comprises the Nestos/Mesta river basin shared between Greece and Bulgaria. The Nestos/Mesta river springs from the Rila Mountains (BG) and discharges in the Thracian Sea (GR). Its basin area is approximately equal to 5,479 km2 and its length is about 243 km. The river forms a significant ecosystem throughout its course and its delta is a unique ecosystem protected by the Ramsar Convention and considered as a first priority site under EU Natura 2000. Two dams operate in the Greek part of the basin (downstream) which are mainly used for electricity production purposes, covering also irrigation needs. The main activities supporting local income are agriculture and livestock.

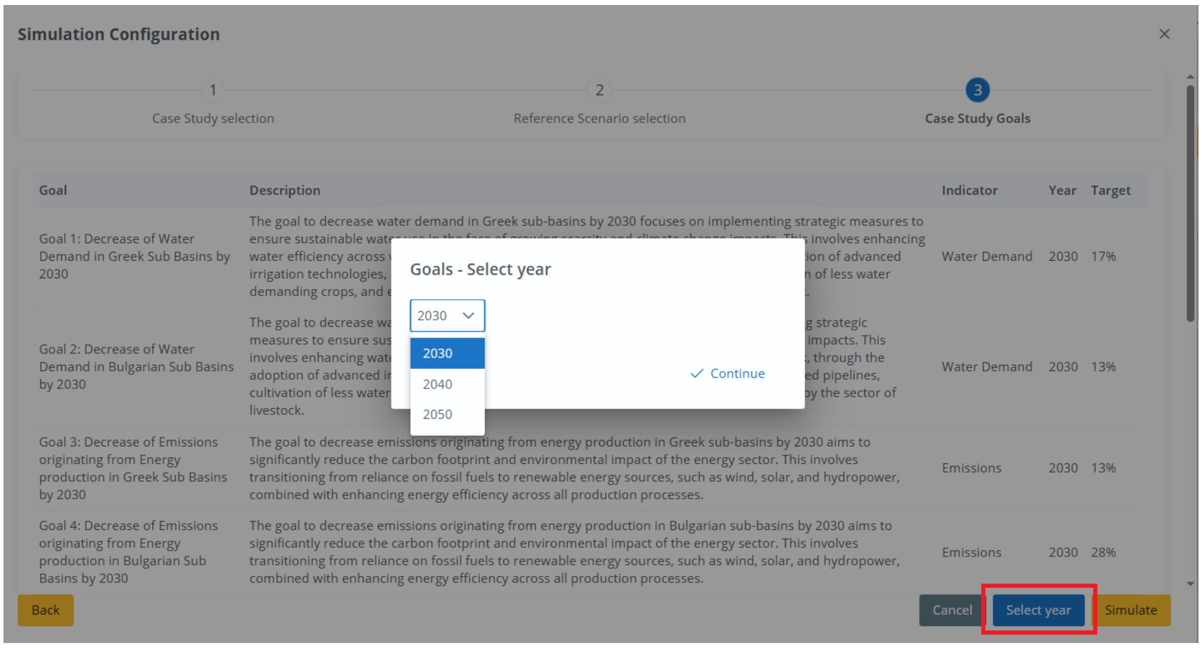
➔ **More Info:** https://nexogenesis.eu/case-study-1-nestos-river-basin/

Figure 14. Case study information in the Simulation Configuration

2. **Select the Reference Scenario:** Users can select from various combinations of RCPs and SSPs. Each combination defines distinct climate change and socio-economic scenarios. Further details about these scenarios are available through an info button.

3. **Policy Goals:** This step presents predefined policy goals for the selected case study, each with a description, an associated indicator, a target value, and the year by which the target is intended to be achieved. In the Nestos case study, users also have the option to customize the target year for the goals available in this section (see Figure 15).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881



Simulation Configuration

1 Case Study selection — 2 Reference Scenario selection — 3 Case Study Goals

Goal	Description	Indicator	Year	Target
Goal 1: Decrease of Water Demand in Greek Sub Basins by 2030	The goal to decrease water demand in Greek sub-basins by 2030 focuses on implementing strategic measures to ensure sustainable wat… involves enhancing water efficiency across …ion of advanced irrigation technologies, …n of less water demanding crops, and …	Water Demand	2030	17%
Goal 2: Decrease of Water Demand in Bulgarian Sub Basins by 2030	The goal to decrease wa… g strategic measures to ensure sus… impacts. This involves enhancing wat… , through the adoption of advanced i… ed pipelines, cultivation of less water… by the sector of livestock.	Water Demand	2030	13%
Goal 3: Decrease of Emissions originating from Energy production in Greek Sub Basins by 2030	The goal to decrease emissions originating from energy production in Greek sub-basins by 2030 aims to significantly reduce the carbon footprint and environmental impact of the energy sector. This involves transitioning from reliance on fossil fuels to renewable energy sources, such as wind, solar, and hydropower, combined with enhancing energy efficiency across all production processes.	Emissions	2030	13%
Goal 4: Decrease of Emissions originating from Energy production in Bulgarian Sub Basins by 2030	The goal to decrease emissions originating from energy production in Bulgarian sub-basins by 2030 aims to significantly reduce the carbon footprint and environmental impact of the energy sector. This involves transitioning from reliance on fossil fuels to renewable energy sources, such as wind, solar, and hydropower, combined with enhancing energy efficiency across all production processes.	Emissions	2030	28%

Goals - Select year

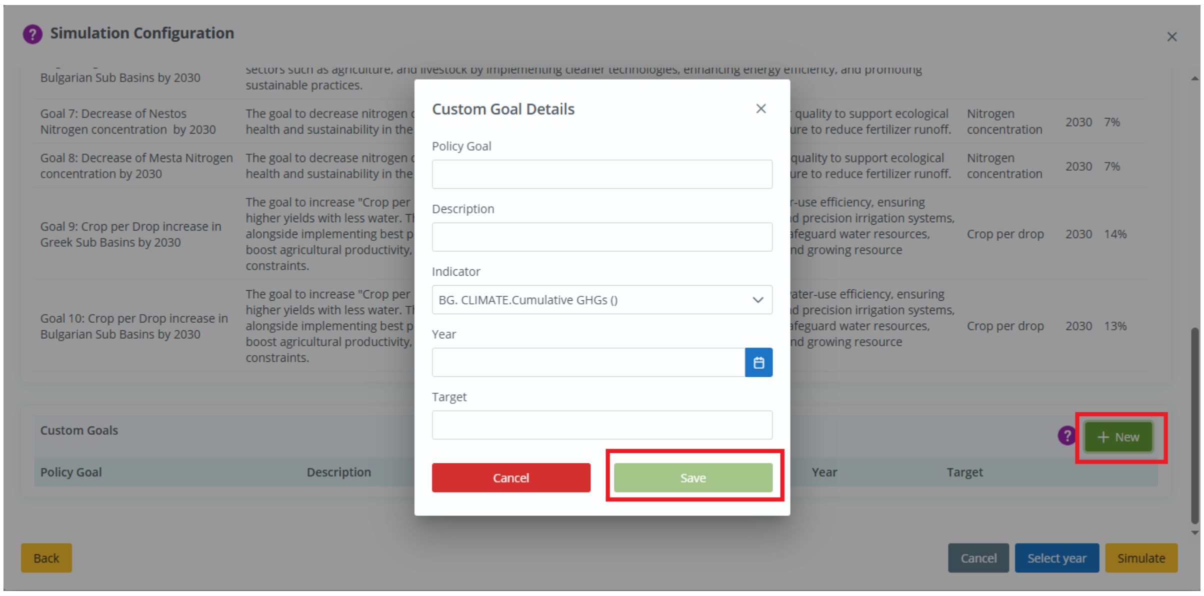
2030

Figure 15. Policy goals customization year in the Simulation Configuration

After completing the configuration, users can click *Simulate*, which transitions them directly to the **Simulations View**.

Advanced functionalities

When advanced functionalities are enabled, users can define custom goals during simulation configuration. Each goal consists of an indicator, a target value—specified as an increase or decrease from the reference scenario—and a target year for achievement. Additionally, users can assign a name and description to provide further context. This process takes place in the third step of the configuration wizard, known as the *Policy Goals* step (Figure 16). Once all goals are defined, users can proceed to the *Simulate* option, which transitions them to the simulation view.



Simulation Configuration

Bulgarian Sub Basins by 2030	sectors such as agriculture, and livestock by implementing cleaner technologies, enhancing energy efficiency, and promoting sustainable practices.			
Goal 7: Decrease of Nestos Nitrogen concentration by 2030	The goal to decrease nitrogen c… quality to support ecological health and sustainability in the… ure to reduce fertilizer runoff.	Nitrogen concentration	2030	7%
Goal 8: Decrease of Mesta Nitrogen concentration by 2030	The goal to decrease nitrogen c… quality to support ecological health and sustainability in the… ure to reduce fertilizer runoff.	Nitrogen concentration	2030	7%
Goal 9: Crop per Drop increase in Greek Sub Basins by 2030	The goal to increase "Crop per… r-use efficiency, ensuring higher yields with less water. Th… d precision irrigation systems, alongside implementing best p… afeguard water resources, boost agricultural productivity, … nd growing resource constraints.	Crop per drop	2030	14%
Goal 10: Crop per Drop increase in Bulgarian Sub Basins by 2030	The goal to increase "Crop per… ater-use efficiency, ensuring higher yields with less water. Th… d precision irrigation systems, alongside implementing best p… afeguard water resources, boost agricultural productivity, … nd growing resource constraints.	Crop per drop	2030	13%

Custom Goals

Policy Goal	Description	Year	Target

Custom Goal Details

Policy Goal:

Description:

Indicator: BG. CLIMATE.Cumulative GHGs ()

Year:

Target:

Figure 16. Defining custom goals in the simulation configuration

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

2.5.4. Simulations View

At the end of the simulation configuration process, the user is directed to the Simulation View. The initial interface is shown in Figure 17. This view is divided into two main sections, highlighted in blue in the figure.

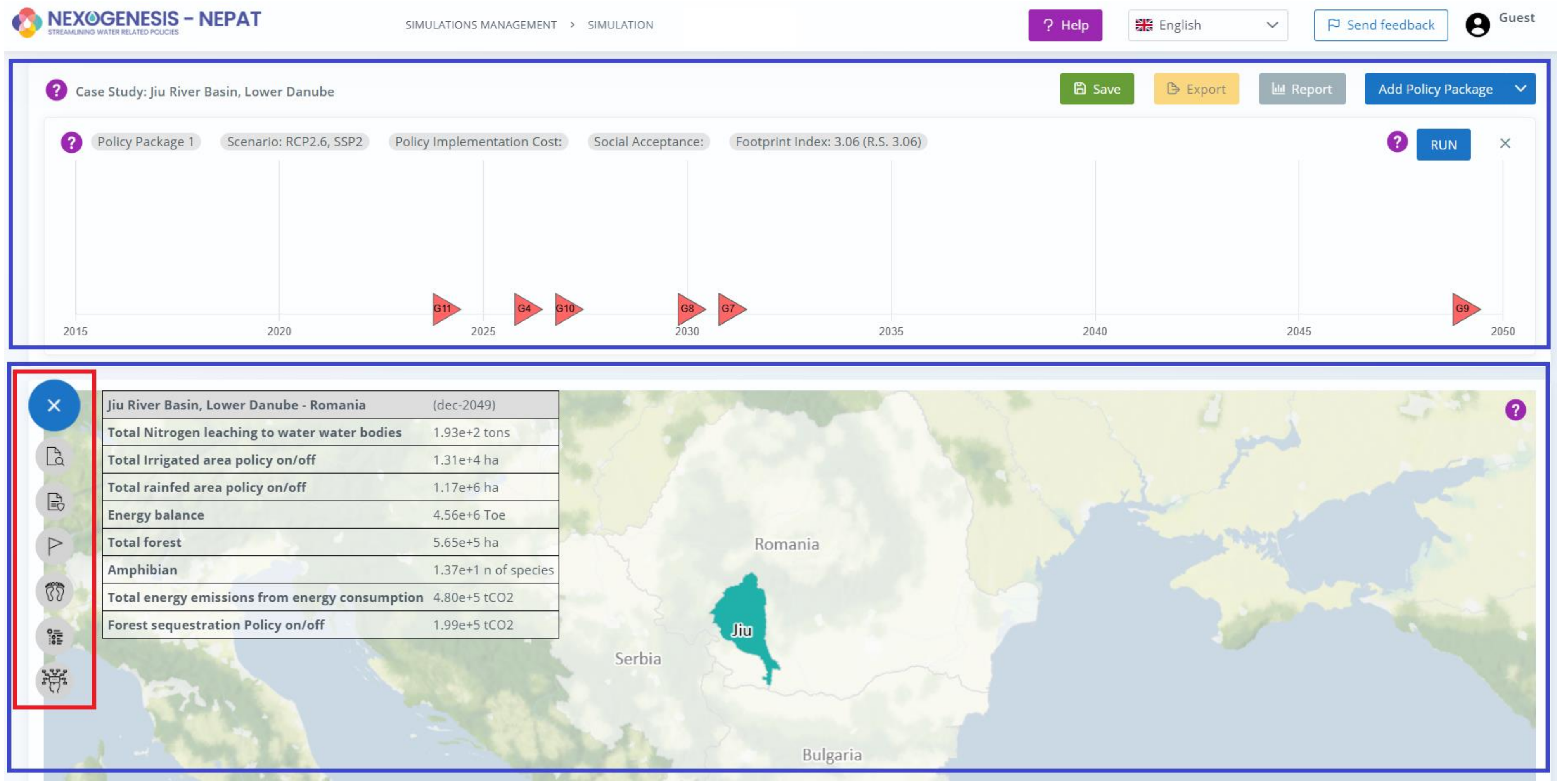
The upper section displays the **Policy Package (PP) Summary**, where the policies configured by the user are graphically represented. This section allows users to visualize the duration of each policy, from the moment of implementation to its conclusion. Additionally, coloured flags along the x-axis indicate the degree of achievement of the simulation's policy goals.

The lower section provides access to the policy configuration for policy scenarios as well as visualization tools for the simulation results. The dropdown menu, highlighted in red, enables navigation through the various features available in the Simulation View:

- **Global View**: Displays relevant information of the case study.
- **Policy Instruments**: Allows users to add and configure the policies instruments within a policy package.
- **Policy Goals**: Provides an overview of goal achievement levels for each objective in the case study.
- **Nexus Footprint**: Evaluates the impacts of policy packages on specific sectors and the overall nexus health.
- **Detailed View**: Offers an in-depth analysis of variables used to simulate the future scenario.
- **Decision Support System**: Leverages AI to provide optimized policy recommendations tailored to user-defined preferences.

The main sections and functionalities are described in detail below.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881



Jiu River Basin, Lower Danube - Romania	(dec-2049)
Total Nitrogen leaching to water water bodies	1.93e+2 tons
Total Irrigated area policy on/off	1.31e+4 ha
Total rainfed area policy on/off	1.17e+6 ha
Energy balance	4.56e+6 Toe
Total forest	5.65e+5 ha
Amphibian	1.37e+1 n of species
Total energy emissions from energy consumption	4.80e+5 tCO2
Forest sequestration Policy on/off	1.99e+5 tCO2

Figure 17. Simulations view of the NEPAT platform

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Policy Package Summary

The Policy Package Summary section provides an overview of the simulated policy packages, enabling users to effectively manage and analyze policy scenarios. Key functionalities available in the top menu, as indicated in Figure 18, include: the ability to add new policy packages, save progress, export simulations, and generate detailed reports.

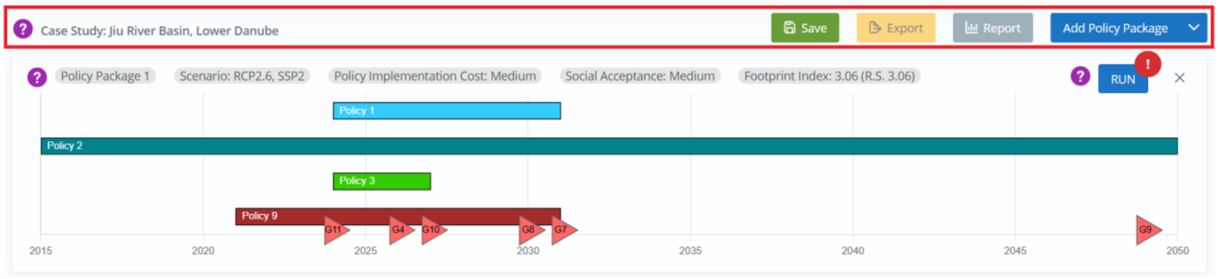


Figure 18. Top menu of the Policy Package Summary

By selecting the *Add Policy Package* option, users can compare the effects of different policies or scenarios, facilitating scenario analysis. Progress of individual simulations and multiple simulations comparisons can be preserved in the Management View by clicking the *Save* button, while the *Export* option allows users to share simulations with others. Additionally, users can generate PDF reports that summarize key insights by selecting the *Report* option.

Each policy package simulates scenarios spanning from 2015 to 2050. The information displayed in the Policy Package Summary regarding the policy package simulation is highlighted in Figure 19. The provided information includes the following:

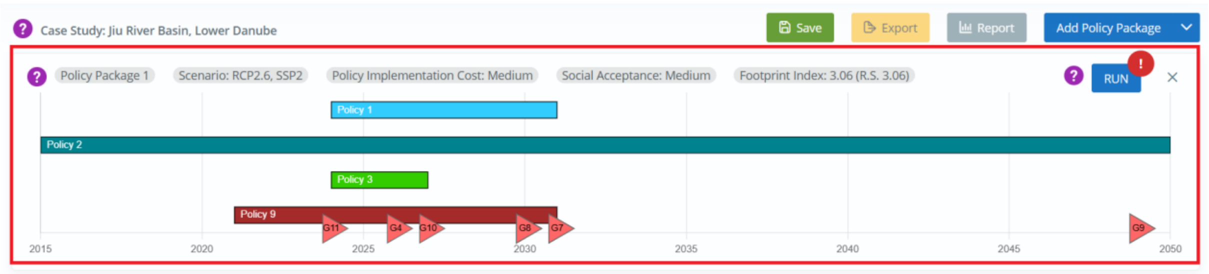


Figure 19. Information displayed in the Policy Package Summary

- **Scenario**: A combination of RCPs and SSPs that represent the biophysical and socioeconomic future.
- **Policy Implementation Cost**: A qualitative estimate of the economic costs associated with applying the selected policies.
- **Social Acceptance**: A qualitative measure reflecting societal acceptance of the policies implemented.
- **WEFE Footprint Index**: A numeric value indicating the environmental footprint at the end of the simulation. Additional details on this metric can be accessed through the *Nexus Footprint* section.
- **Policy Instruments**: Each horizontal bar represents a policy instrument. The start of the bar indicates the year of implementation, its length represents the duration of

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

application, and the endpoint marks the year of completion. Users can modify policy packages by adding or removing individual policies. To add a new policy, navigate to the *Policy Instruments* menu, while individual policies can be removed by hovering over them and selecting the red *Delete* button.

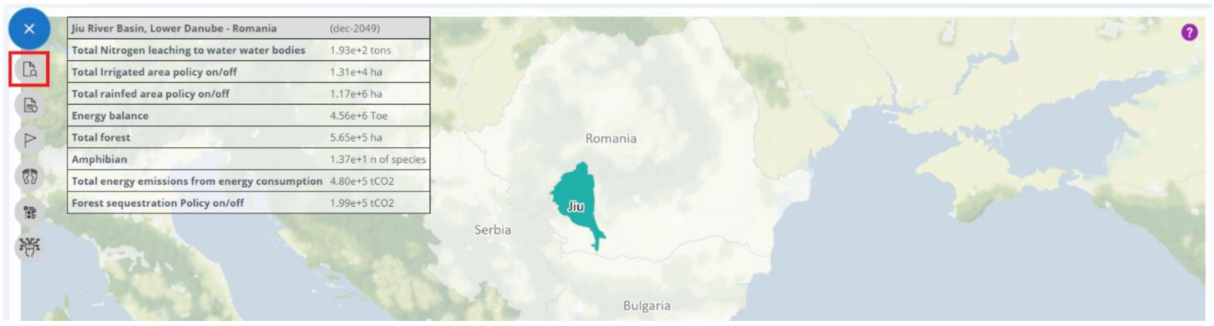
- **Goal Flags**: Color-coded indicators that display the level of achievement for each goal in the selected case study. More information about these indicators is available in the *Goals* section.

Once a policy package has been configured, users can simulate its effects by clicking the *RUN* button. After the simulation is complete, results can be explored through the various views accessible via the main menu. Each view includes a help icon, offering detailed guidance and tips for maximizing the use of the tool's features.

It is important to note that any modifications to a policy package require re-simulation. The *RUN* button will appear with an exclamation mark, indicating the need to rerun the simulation to update all result views.

Global View

This section presents key indicators for the selected Case Study, along with its geographic location displayed on a map. It provides a concise overview of the basin's main characteristics and context, setting the stage for simulation analysis. For certain cases, such as Nestos or Lielupe, users can interact with the map to view indicators for each country or region (see Figure 20).



Jiu River Basin, Lower Danube - Romania	(dec-2049)
Total Nitrogen leaching to water water bodies	1.93e+2 tons
Total Irrigated area policy on/off	1.31e+4 ha
Total rainfed area policy on/off	1.17e+6 ha
Energy balance	4.56e+6 Toe
Total forest	5.65e+5 ha
Amphibian	1.37e+1 n of species
Total energy emissions from energy consumption	4.80e+5 tCO2
Forest sequestration Policy on/off	1.99e+5 tCO2

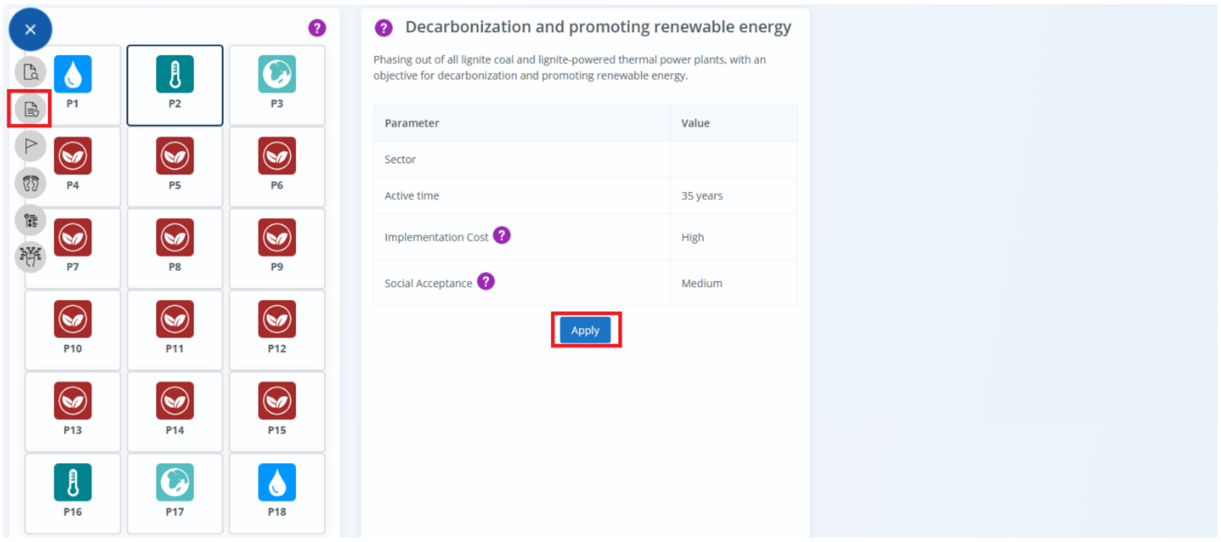
Figure 20. Global View of the NEPAT platform

Policy Instruments View

The Policy Instruments View offers users detailed insights into all the policy instruments defined for the case study. Policy instruments are organized by sector, with each sector represented by distinct colors and icons for easy identification. These instruments are integrated into SDMs, enabling detailed analysis of their impacts—either individually or as part of a combined policy package—on the WEFE nexus.
Selecting a policy provides a clear view of its definition and key parameters, displayed at the center of the screen. To include a policy instrument in the simulation, users can click the *Apply* button. Once added, the policy will appear in the *Policy Package Summary* section at the top of the page. Users can then simulate the policy scenario for the 2015–2050 period by clicking the *RUN* button. This generates results across the WEFE nexus and updates all views. The

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

simulated scenario based on the selected policy package is referred to as the policy future scenario. This view is illustrated in Figure 21.



Decarbonization and promoting renewable energy

Phasing out of all lignite coal and lignite-powered thermal power plants, with an objective for decarbonization and promoting renewable energy.

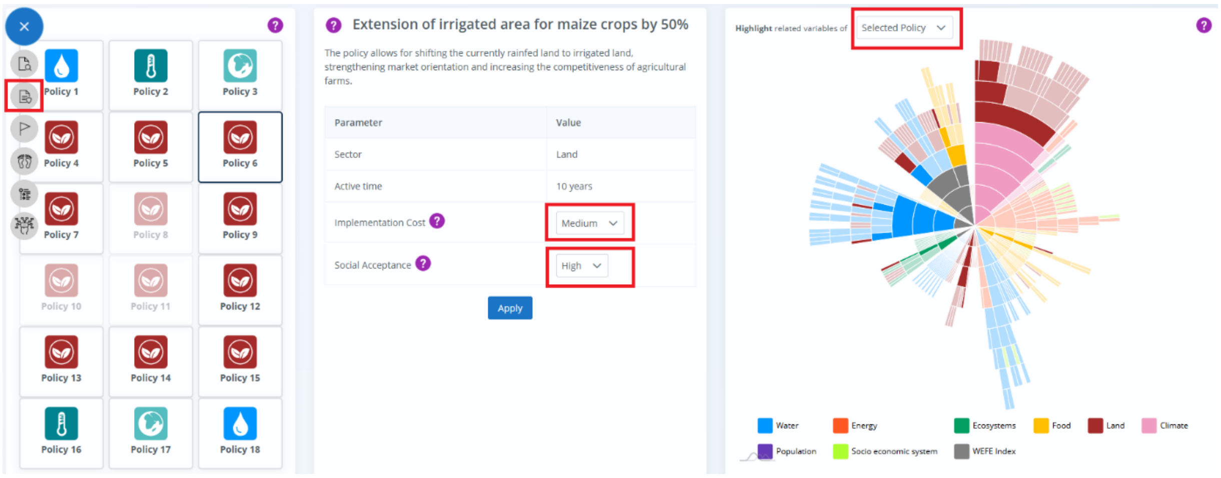
Parameter	Value
Sector	
Active time	35 years
Implementation Cost	High
Social Acceptance	Medium

Apply

Figure 21. Policy Instruments View of the NEPAT platform

Advanced functionalities

When advanced functionalities are activated, a **Sunburst Graph** appears on the right-hand side. This graph offers a hierarchical visualization of all variables in the system dynamics model, showing their interdependencies and relationships (see Figure 22).



Extension of irrigated area for maize crops by 50%

The policy allows for shifting the currently rainfed land to irrigated land, strengthening market orientation and increasing the competitiveness of agricultural farms.

Parameter	Value
Sector	Land
Active time	10 years
Implementation Cost	Medium
Social Acceptance	High

Apply

Highlight related variables of: Selected Policy

Figure 22. Policy Instruments View of the NEPAT platform with advanced functionalities

The **Sunburst Graph** provides a visual representation of the relationships and dependencies within the sectors and system dynamics model. High-level variables are located at the center, with outer layers representing the components contributing to their calculation. As users move outward from the center, they can explore how each variable is broken down into its constituent elements. Variables are color-coded according to typologies and sectors, as indicated in the legend.
In the graph, a dropdown menu allows users to focus the graph on specific aspects, such as:

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

- Variables affected by a selected policy instrument to analyze the impacts of each individual policy.
- Variables influenced by a specific policy package to evaluate the impacts of policy packages as a whole.
- Comparisons between the reference scenario and a policy scenario. In this comparison, the impacts are presented to highlight the differences between the expected future outcomes of the reference scenario—determined by the selected combination of RCP and SSP—and the anticipated outcomes of the selected policy scenario within the same climate and socio-economic context.

Moreover, with advanced functionalities activated, users can customize specific parameters before applying policies. These parameters are highlighted in the center of Figure 22. While default values are intuitive estimates, they are not derived from rigorous scientific analysis.

1. **Implementation Cost**: A qualitative indicator representing the estimated economic burden of a policy can be customized.
2. **Social Acceptance**: A qualitative indicator reflecting the estimated societal reception of a policy can be customized.

Finally, in the Inkomati CS, users can customize the **policy application year** as a configuration parameter. This option is exclusive to the Inkomati CS. Each policy instrument has a default application year, which varies across policies (see Figure 19). For all other CS, the application year is fixed and cannot be customized.

Policy Goals View

The Policy Goals View provides information on the degree of achievement of the various objectives defined for the case study (see Figure 23).

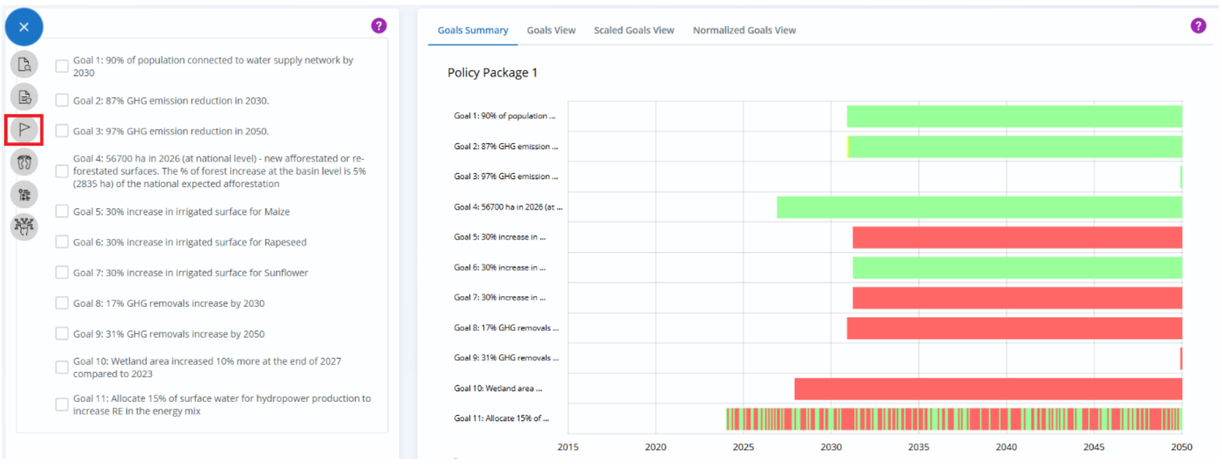

Goal 1: 90% of population connected to water supply network by 2030
Goal 2: 87% GHG emission reduction in 2030.
Goal 3: 97% GHG emission reduction in 2050.
Goal 4: 56700 ha in 2026 (at national level) - new afforestated or re-forestated surfaces. The % of forest increase at the basin level is 5% (2835 ha) of the national expected afforestation
Goal 5: 30% increase in irrigated surface for Maize
Goal 6: 30% increase in irrigated surface for Rapeseed
Goal 7: 30% increase in irrigated surface for Sunflower
Goal 8: 17% GHG removals increase by 2030
Goal 9: 31% GHG removals increase by 2050
Goal 10: Wetland area increased 10% more at the end of 2027 compared to 2023
Goal 11: Allocate 15% of surface water for hydropower production to increase RE in the energy mix
Goals Summary
Goals View
Scaled Goals View
Normalized Goals View
Policy Package 1
Goal 1: 90% of population ...
Goal 2: 87% GHG emission ...
Goal 3: 97% GHG emission ...
Goal 4: 56700 ha in 2026 (at ...
Goal 5: 30% increase in ...
Goal 6: 30% increase in ...
Goal 7: 30% increase in ...
Goal 8: 17% GHG removals ...
Goal 9: 31% GHG removals ...
Goal 10: Wetland area ...
Goal 11: Allocate 15% of ...
2015
2020
2025
2030
2035
2040
2045
2050

Figure 23. Policy Goals View of the NEPAT platform

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

The **Goals Summary** provides an overview of progress toward achieving the defined targets, enabling users to assess the overall performance of the selected policy package. The degree of achievement is calculated relative to the reference scenario and is represented using a color-coded system for easy interpretation, indicating whether the goals are met, improved upon, or worsened in terms of progress toward their achievement.

- **Green** indicates that the goal has been achieved.
- **Yellow** signifies that progress is closer to the goal than to the reference scenario.
- **Red** shows that progress is closer to the reference scenario than to the goal.
- **Black** indicates that progress is moving further away from the goal.

Additional detailed views enable in-depth exploration of individual goals. Users can select a specific goal from the list and visualize it using one of three specialized views: the Goals View, the Scaled Goals View, or the Normalized Goals View. All charts are interactive, allowing users to customize them by toggling variables on or off via the legend.

The **Goals View** displays the evolution of selected indicators over time, from 2015 to 2050, using a monthly timestep. It enables users to compare the reference scenario, policy scenario, and target values.

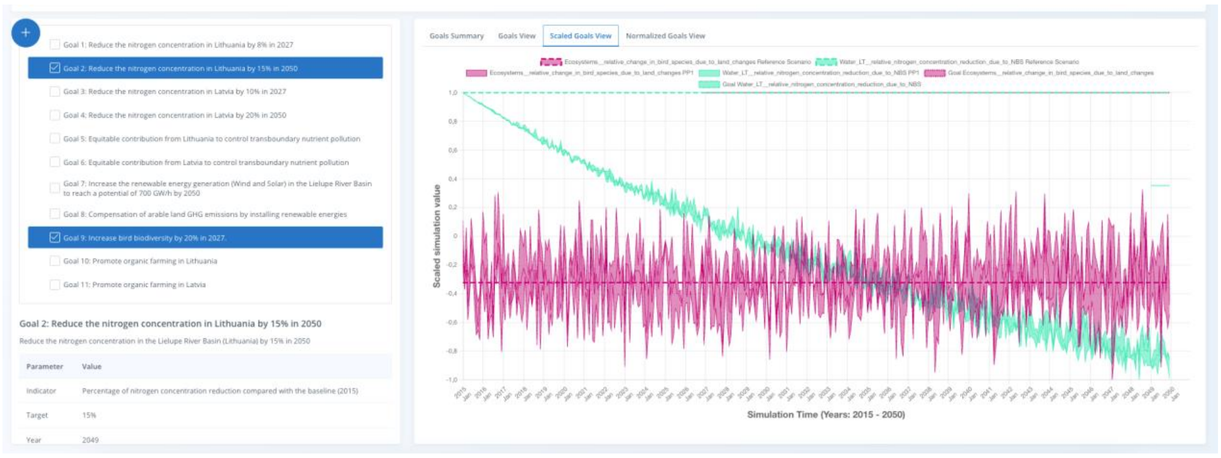
The **Scaled Goals View** enhances insights by normalizing data using min-max scaling, which adjusts all values to a range between -1 and 1. This approach allows for clearer comparisons of selected indicators over time, highlighting trends and changes more effectively.

The **Normalized Goals View** further refines analysis by normalizing data relative to the reference scenario, which is fixed at a value of 1. This enables users to observe how the policy scenario diverges or aligns with the reference scenario over time.

Advanced functionalities

When advanced functionalities are activated and custom goals are defined in the simulation configuration, these goals become visible in this view. This allows users to track progress not only toward the default goals but also toward their user-defined goals.

Additionally, when uncertainty is activated and set to a value greater than 1, the system executes multiple runs of the stochastic SDM within each simulation. Consequently, variations in outcomes may be observed across several variables. The goal detail views aggregate these results by displaying their quartiles (Q1, Q2 or median, and Q3), enabling users to analyze the distribution of Monte Carlo simulation results. An example of these results is shown in Figure 24.



- [ ] Goal 1: Reduce the nitrogen concentration in Lithuania by 8% in 2027
- [✓] Goal 2: Reduce the nitrogen concentration in Lithuania by 15% in 2050
- [ ] Goal 3: Reduce the nitrogen concentration in Latvia by 10% in 2027
- [ ] Goal 4: Reduce the nitrogen concentration in Latvia by 20% in 2050
- [ ] Goal 5: Equitable contribution from Lithuania to control transboundary nutrient pollution
- [ ] Goal 6: Equitable contribution from Latvia to control transboundary nutrient pollution
- [ ] Goal 7: Increase the renewable energy generation (Wind and Solar) in the Lielupe River Basin to reach a potential of 700 GW/h by 2050
- [ ] Goal 8: Compensation of arable land GHG emissions by installing renewable energies
- [✓] Goal 9: Increase bird biodiversity by 20% in 2027.
- [ ] Goal 10: Promote organic farming in Lithuania
- [ ] Goal 11: Promote organic farming in Latvia

Goal 2: Reduce the nitrogen concentration in Lithuania by 15% in 2050

Reduce the nitrogen concentration in the Lielupe River Basin (Lithuania) by 15% in 2050

Parameter	Value
Indicator	Percentage of nitrogen concentration reduction compared with the baseline (2015)
Target	15%
Year	2049

Goals Summary
Goals View
Scaled Goals View
Normalized Goals View
Ecosystems__relative_change_in_bird_species_due_to_land_changes Reference Scenario
Water_LT__relative_nitrogen_concentration_reduction_due_to_NBS Reference Scenario
Ecosystems__relative_change_in_bird_species_due_to_land_changes PP1
Water_LT__relative_nitrogen_concentration_reduction_due_to_NBS PP1
Goal Ecosystems__relative_change_in_bird_species_due_to_land_changes
Goal Water_LT__relative_nitrogen_concentration_reduction_due_to_NBS
Scaled simulation value
1,0
0,8
0,6
0,4
0,2
0
-0,2
-0,4
-0,6
-0,8
-1,0
Simulation Time (Years: 2015 - 2050)

Figure 24. Aggregated goal results with quartile distribution in stochastic simulations

Nexus Footprint View

The **WEFE Footprint Index** offers a comprehensive annual evaluation of the WEFE Nexus, assessing performance and interrelations across its four main pillars: Water, Energy, Food, and Ecosystems. The index assigns values on a scale from -100 to 100, reflecting progress, stability, or regression in these areas. This framework enables users to analyze both individual and collective impacts across the Nexus in a structured manner.

The index is composed of a hierarchical structure, which includes:

- **Four main pillars**: Water, Energy, Food, and Ecosystems.
- **Nine sub-pillars:** Each main pillar is further divided into sub-pillars that provide finer detail.
- **Sixteen indicators:** These measurable variables contribute to the sub-pillars and pillars, offering specific insights into the Nexus dynamics.

This layered structure allows for analysis at multiple levels, ranging from a high-level overview to detailed insights into individual indicators.

The **Nexus Footprint Summary**, located on the right side of the interface, provides an aggregated overview of how a selected future scenario impacts the WEFE Footprint Index compared to the baseline year of 2015 (see Figure 25). The Nexus Footprint Summary combines all variables and presents their effects at the index, pillar, sub-pillar, and indicator levels, enabling users to assess whether, for example, the Water pillar performs significantly better, worse, or similarly to the baseline.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

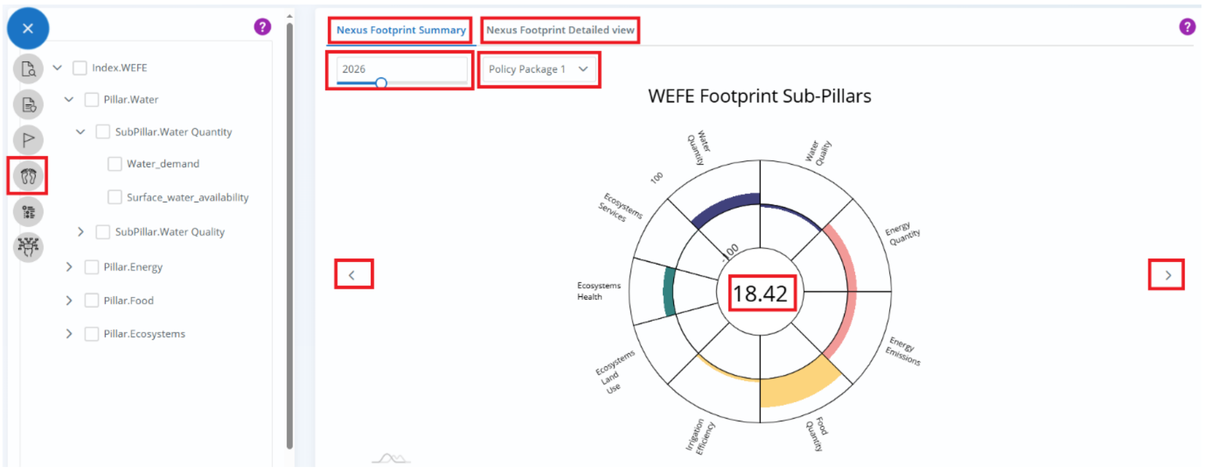


Figure 25. WEFE Nexus Footprint View of the NEPAT platform

The summary is visualized through an interactive footprint diagram, which includes several key features:

- **Year selection:** The footprint updates dynamically based on the year selected for analysis.
- **Policy Package selection:** When multiple policy packages are available, users can select one to observe its specific impact on the footprint.
- **Central Index:** The aggregated WEFE Footprint Index is displayed at the center of the diagram, summarizing the overall performance.
- **Navigation through levels**: On-screen arrows allow users to navigate between different levels—pillars, sub-pillars, and indicators—offering a granular analysis.
- **Color coding**: Each pillar is represented by a specific color for easy identification: Blue for Water, Red for Energy, Yellow for Food, and Green for Ecosystems.
- **Hover functionality**: Users can hover over colored sections of the diagram to view specific values of individual indicators, providing detailed insights.

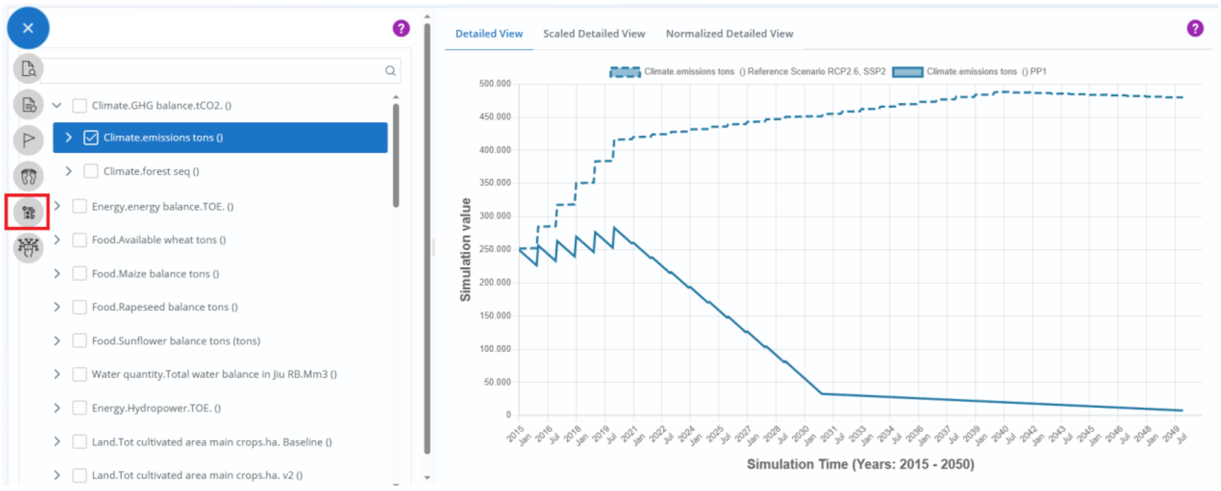
At all levels (indicators, sub-pillars, and pillars), values range between -100 and 100, with positive values indicating beneficial impacts compared to 2015, and negative values representing detrimental effects.

The **Nexus Footprint Detailed View** provides a deeper analysis of the selected indicators over time, covering the period from 2015 to 2050 with an annual timestep. This view enables users to compare the evolution of indicators between the reference scenario (without policies) and the policy future scenario (with selected policy packages), offering insights into the respective impacts of these scenarios. The chart is interactive and customizable. Users can tailor their analysis by selecting specific variables from the legend, toggling their visibility on or off to focus on the data most relevant to their interests.

Detailed View

The Detailed View visualizes all system dynamics model variables on a monthly timestep over the 35-year simulation (2015–2050). Variables are hierarchically organized, allowing users to expand them for a step-by-step examination of dependencies. A search bar enables efficient navigation.

Users can compare each variable between the reference scenario (no policies) and the policy scenario (with selected policies). Selecting variables from the list displays detailed information through three viewing options: Detailed View, Scaled Detailed View, and Normalized Detailed View (see Figure 26). The interactive chart allows variable customization via the legend.



Climate.GHG balance.tCO2. ()
Climate.emissions tons ()
Climate.forest seq ()
Energy.energy balance.TOE. ()
Food.Available wheat tons ()
Food.Maize balance tons ()
Food.Rapeseed balance tons ()
Food.Sunflower balance tons (tons)
Water quantity.Total water balance in Jiu RB.Mm3 ()
Energy.Hydropower.TOE. ()
Land.Tot cultivated area main crops.ha. Baseline ()
Land.Tot cultivated area main crops.ha. v2 ()
Detailed View
Scaled Detailed View
Normalized Detailed View
Climate.emissions tons () Reference Scenario RCP2.6, SSP2
Climate.emissions tons () PP1
Simulation value
Simulation Time (Years: 2015 - 2050)

Figure 26. Detailed View of the NEPAT platform

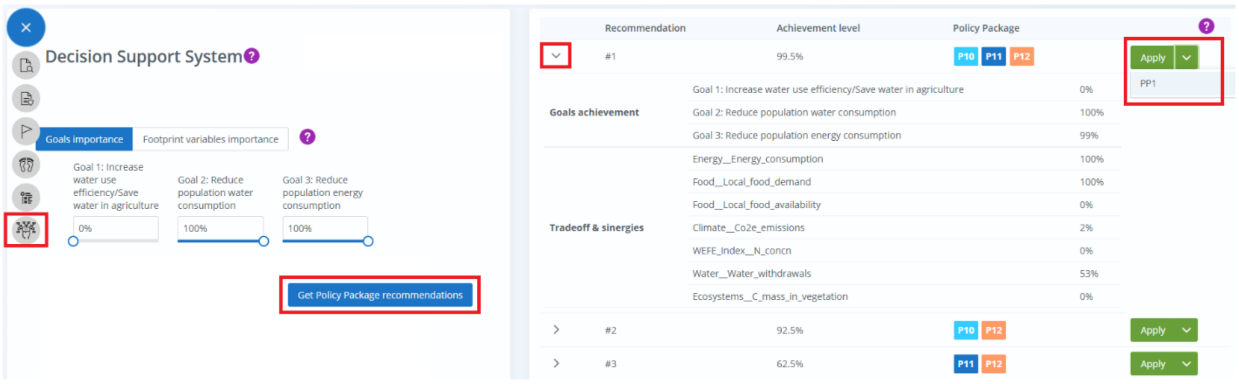
Advanced functionalities

When advanced functionalities are activated and uncertainty is set to a value greater than 1, multiple runs of the stochastic SDM are executed in each simulation. As a result, different outcomes may be observed for several variables. The detailed view aggregates the results of multiple runs by displaying their quartiles—Q1, Q2, and Q3—allowing users to analyze the distribution of Monte Carlo results in the stochastic simulations.

Decision Support System

The NEPAT DSS is designed to assist in identifying policy packages that align with specific user-defined goals. The tool provides flexible functionality to generate recommendations based on various criteria, enabling users to explore tailored policy solutions and evaluate their potential impacts. To generate policy recommendations, users must first define their criteria and then click on the **Get Policy Package Recommendations** button (see Figure 27).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881


Decision Support System
Goals importance
Footprint variables importance
Get Policy Package recommendations
Apply
PP1

Goal 1: Increase water use efficiency/Save water in agriculture: 0%
Goal 2: Reduce population water consumption: 100%
Goal 3: Reduce population energy consumption: 100%

	Recommendation	Achievement level	Policy Package
	#1	99.5%	P10 P11 P12
Goals achievement	Goal 1: Increase water use efficiency/Save water in agriculture	0%	
	Goal 2: Reduce population water consumption	100%	
	Goal 3: Reduce population energy consumption	99%	
Tradeoff & sinergies	Energy__Energy_consumption	100%	
	Food__Local_food_demand	100%	
	Food__Local_food_availability	0%	
	Climate__Co2e_emissions	2%	
	WEFE_Index__N_concn	0%	
	Water__Water_withdrawals	53%	
	Ecosystems__C_mass_in_vegetation	0%	
	#2	92.5%	P10 P12
	#3	62.5%	P11 P12

Figure 27. Decision Support System View of the NEPAT platform

The DSS allows users to specify their priorities by focusing on either goals or footprint variables, but not both simultaneously.

- **For goals**: Users can click on the *Goals Importance* button, select the desired goals, and assign weights (in percentages) to indicate their relative importance. Goals can either be weighted equally or prioritized by assigning higher percentages to certain goals.
- **For footprint variables**: Users can follow a similar process by clicking on the *Footprint Variables Importance* button, selecting variables, and assigning weights. Adjusting these weights allows users to emphasize certain footprint variables in the recommendations.

Recommendations are displayed on the right, ranked by overall goal achievement, with up to 10 options provided. Each policy package includes:

- **Overall goal achievement score**: The average achievement of selected goals.
- **Detailed policy list**: Recommended policies.
- **Expandable goal details**: A dropdown reveals individual goal achievement levels, helping users assess trade-offs and impacts.
- **Apply button**: Allows users to directly apply the recommended policy package.

Advanced functionalities

When advanced functionalities are activated, users can customize DSS criteria based on:

- **Predefined policy package**: Users can base recommendations on an existing package by selecting it from the dropdown after checking the box. To start fresh, leave the box unchecked.
- **Limiting policy instruments or sectors**:
  - *Policy Instruments*: Users can cap the number of instruments in a package by checking the box and setting a maximum. If left unchecked, all available instruments are considered.
  - *Sectors*: Users can focus recommendations on specific sectors by selecting them after checking the box. Leaving it unchecked includes all sectors.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

In the Inkomati CS, when advanced functionalities are activated, users can customize the year of policy application. The advanced DSS provides additional recommendations with dynamic years only if a better solution is found; otherwise, only default-year recommendations are displayed.

When uncertainty is activated and set above 1, the DSS incorporates the stochastic SDM in its recommendations. An additional column in the expandable goal detail shows the range of goal achievement levels across 100 runs, offering insight into the impact of stochasticity. This applies only to policy goals, as footprint variables remain deterministic.

Comparison

Within the same case study, policy instruments, goals, variables, and indicators remain fixed. To enable the comparison of policy scenarios—whether within the same or a different reference scenario—the tool provides a functionality that allows users to add multiple policy packages to a simulation, forming a comparison. Figure 28 illustrates a policy package summary containing two policy packages. Users can add multiple policy packages, configure distinct policy scenarios, and run simulations. Each policy package requires a separate run. Additionally, policy packages can be removed by clicking the *cross icon* next to the *Run* button.

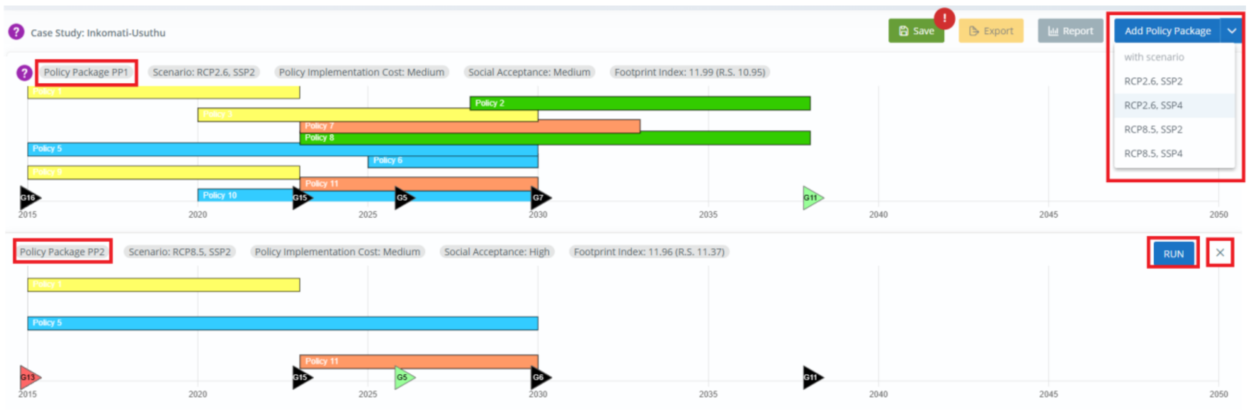
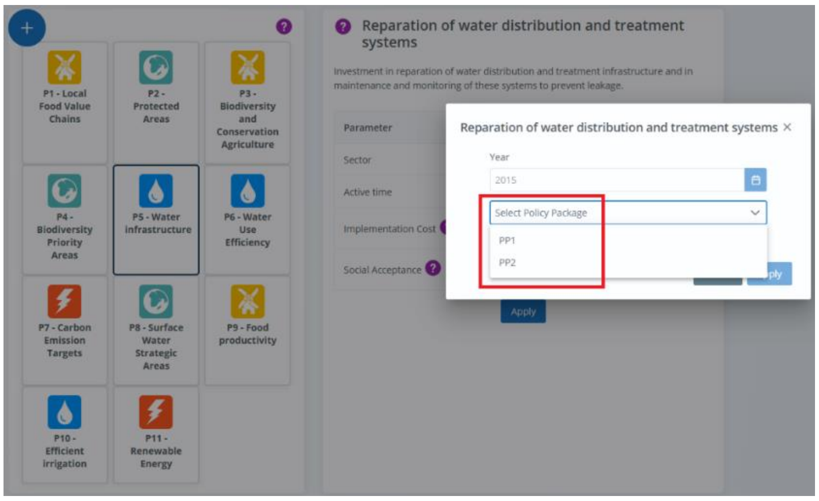


Figure 28. Managing and comparing policy packages in the NEPAT

To select policy instruments for policy package simulation, the user—now working with multiple PPs—must specify which PPs each policy instrument should be included in, as illustrated in Figure 29.


P1 - Local Food Value Chains
P2 - Protected Areas
P3 - Biodiversity and Conservation Agriculture
P4 - Biodiversity Priority Areas
P5 - Water infrastructure
P6 - Water Use Efficiency
P7 - Carbon Emission Targets
P8 - Surface Water Strategic Areas
P9 - Food productivity
P10 - Efficient irrigation
P11 - Renewable Energy
Reparation of water distribution and treatment systems
Parameter
Sector
Active time
Implementation Cost
Social Acceptance
Year
2015
Select Policy Package
PP1
PP2
Apply

Investment in reparation of water distribution and treatment infrastructure and in maintenance and monitoring of these systems to prevent leakage.

Figure 29. Selection of policy instruments for multiple policy packages

The different result views, such as policy goals, nexus footprint, and detailed view, display the outcomes for each policy package separately. For example, Figure 30 illustrates the policy goals view for two distinct policy packages.

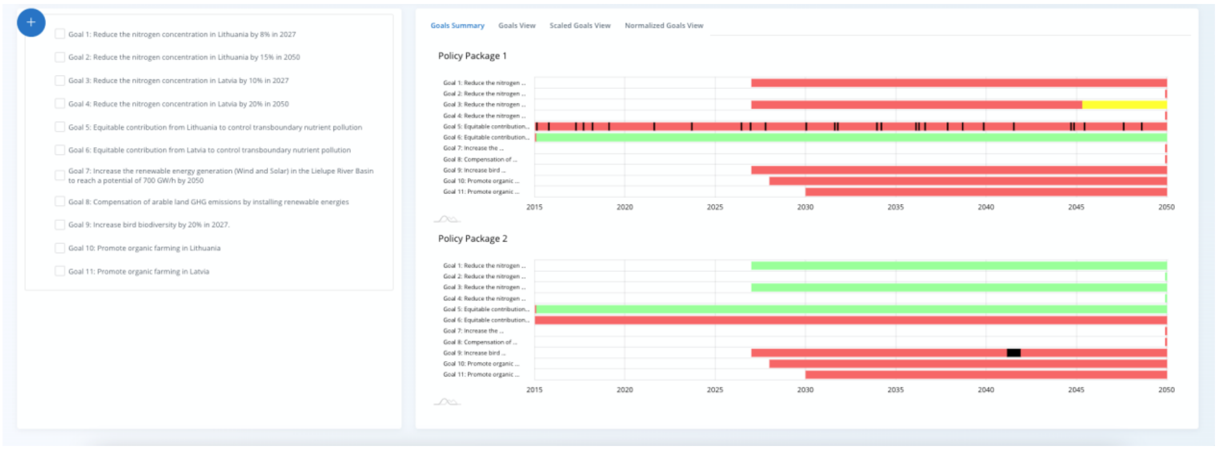

- Goal 1: Reduce the nitrogen concentration in Lithuania by 8% in 2027
- Goal 2: Reduce the nitrogen concentration in Lithuania by 15% in 2050
- Goal 3: Reduce the nitrogen concentration in Latvia by 10% in 2027
- Goal 4: Reduce the nitrogen concentration in Latvia by 20% in 2050
- Goal 5: Equitable contribution from Lithuania to control transboundary nutrient pollution
- Goal 6: Equitable contribution from Latvia to control transboundary nutrient pollution
- Goal 7: Increase the renewable energy generation (Wind and Solar) in the Lielupe River Basin to reach a potential of 700 GW/h by 2050
- Goal 8: Compensation of arable land GHG emissions by installing renewable energies
- Goal 9: Increase bird biodiversity by 20% in 2027.
- Goal 10: Promote organic farming in Lithuania
- Goal 11: Promote organic farming in Latvia

Figure 30. Policy goals view for two separate policy packages

Advanced functionalities

The advanced functionalities available in the comparison view are the same as those for a single policy package in each specific view. However, instead of displaying the results for a single policy package, each view presents the results of multiple policy packages.

Export and Report

In the simulation configuration, users have the option to import a previously existing simulation. This is only possible if the simulation has been exported beforehand, as the export process ensures it is saved in a compatible format that the system can recognize and successfully load. This functionality facilitates the sharing of simulations among users and enables access

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

from different accounts. Rather than being restricted to a single user account, simulations can be stored as files and imported as needed.

Additionally, to export results in a format that can be viewed and shared outside the tool, a reporting functionality has been developed. This feature allows users to generate a PDF from the results of a simulation, whether for a single policy package or a comparison of multiple policy packages. The PDF is generated using the NXG template and includes an introduction describing the case study for the simulation, followed by a simulation section that details each policy package included in the report. This section includes:

- **Reference scenario description**: Defines the scenario considered, based on a combination of RCP and SSP.
- **Policy package summary**: Outlines the policies included in the package.
- **Policy package impact**: Features the *Sunburst graph* illustrating the policy package's impact and the comparative *Sunburst graph* against the reference scenario.
- **Goals achievement**: Provides a description of the goals considered, along with the *Goals Summary* chart that uses colors to indicate different goal statuses.
- **WEFE Footprint Index**: Includes a brief introduction to the index and visual representations of its pillars, sub-pillars, and indicators.

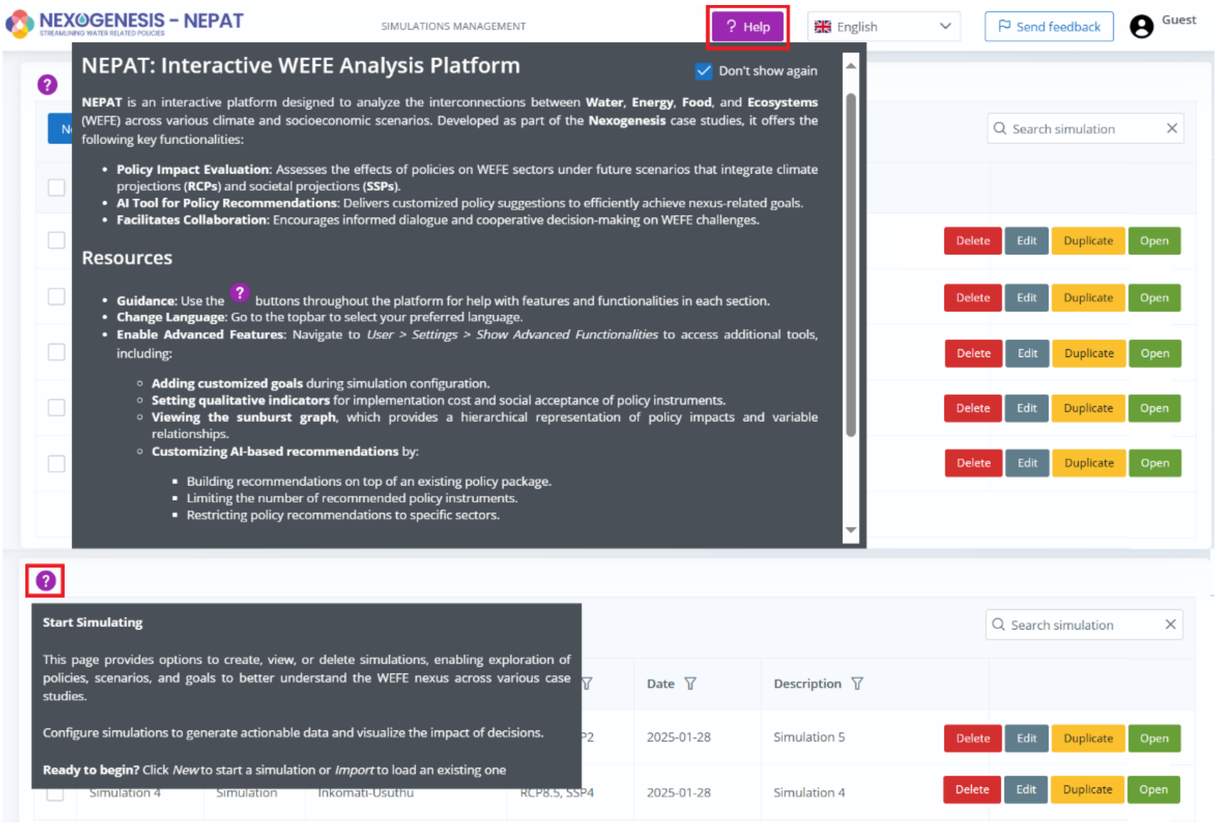
2.5.5. Help User Guide & Languages

Given the high complexity of NEPAT, which integrates numerous functionalities, including advanced features, user support is essential for effective navigation, interaction with the tool, and understanding of the available features. To address this, a **help functionality** has been incorporated across all views. Users can click on the ? icons, which provide relevant information about the type of data displayed in each view, the available user actions, and guidance on interpreting the results. Figure 31 provides an example of the information displayed by these help buttons in the management view.

Since users must navigate through different views to access each ? icon, it was deemed essential to also develop a user manual. This manual consolidates all necessary information across views, ensuring users have access to comprehensive guidance at any time. It includes detailed explanations along with visual examples to enhance understanding of NEPAT. The tutorial has been shared with all case studies to facilitate its distribution among stakeholders.

Furthermore, since each case study operates in a different language, NEPAT has been translated into all required languages: English, Greek, Bulgarian, Latvian, Lithuanian, Romanian, and Italian. These translations cover all explanations, descriptions, and help functionalities.
The translation process is handled by the Text Translator module from the NEPAT Core Service (see Figure 3). Figure 32 presents an example of a translated policy instruments view in the simulation interface, displayed in Romanian.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881



NEPAT: Interactive WEFE Analysis Platform

Don't show again

NEPAT is an interactive platform designed to analyze the interconnections between **Water**, **Energy**, **Food**, and **Ecosystems** (WEFE) across various climate and socioeconomic scenarios. Developed as part of the **Nexogenesis** case studies, it offers the following key functionalities:

- **Policy Impact Evaluation**: Assesses the effects of policies on WEFE sectors under future scenarios that integrate climate projections (**RCPs**) and societal projections (**SSPs**).
- **AI Tool for Policy Recommendations**: Delivers customized policy suggestions to efficiently achieve nexus-related goals.
- **Facilitates Collaboration**: Encourages informed dialogue and cooperative decision-making on WEFE challenges.

Resources

- **Guidance**: Use the ? buttons throughout the platform for help with features and functionalities in each section.
- **Change Language**: Go to the topbar to select your preferred language.
- **Enable Advanced Features**: Navigate to *User > Settings > Show Advanced Functionalities* to access additional tools, including:
  - **Adding customized goals** during simulation configuration.
  - **Setting qualitative indicators** for implementation cost and social acceptance of policy instruments.
  - **Viewing the sunburst graph**, which provides a hierarchical representation of policy impacts and variable relationships.
  - **Customizing AI-based recommendations** by:
    - Building recommendations on top of an existing policy package.
    - Limiting the number of recommended policy instruments.
    - Restricting policy recommendations to specific sectors.

Start Simulating

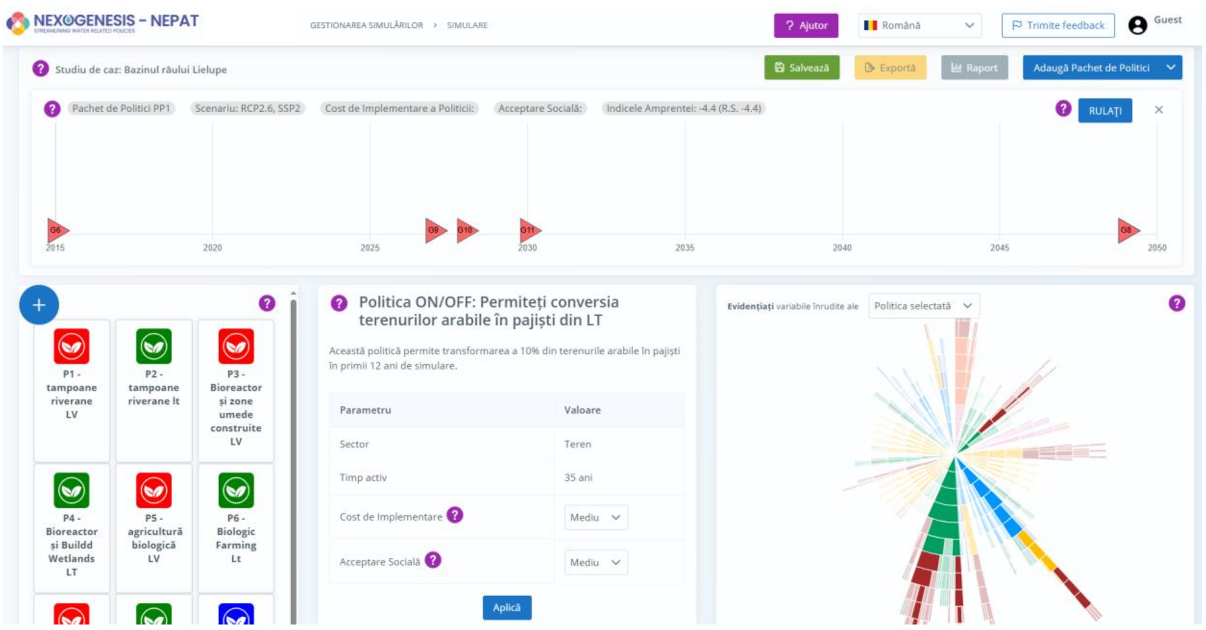
This page provides options to create, view, or delete simulations, enabling exploration of policies, scenarios, and goals to better understand the WEFE nexus across various case studies.

Configure simulations to generate actionable data and visualize the impact of decisions.

Ready to begin? Click *New* to start a simulation or *Import* to load an existing one

				Date	Description
			P2	2025-01-28	Simulation 5
Simulation 4	Simulation	Inkomati-Usuthu	RCP8.5, SSP4	2025-01-28	Simulation 4

Figure 31. Help functionality in the Management view


NEXOGENESIS – NEPAT
GESTIONAREA SIMULĂRILOR > SIMULARE
? Ajutor
Română
Trimite feedback
Guest
Salvează
Exportă
Raport
Adaugă Pachet de Politici
RULAȚI
G6
G9
G10
G11
G8
2015
2020
2025
2030
2035
2040
2045
2050
P1 - tampoane riverane LV
P2 - tampoane riverane lt
P3 - Bioreactor și zone umede construite LV
P4 - Bioreactor și Buildd Wetlands LT
P5 - agricultură biologică LV
P6 - Biologic Farming Lt
Aplică
Evidențiați variabile înrudite ale
Politica selectată

Studiu de caz: Bazinul râului Lielupe

Pachet de Politici PP1 · Scenariu: RCP2.6, SSP2 · Cost de Implementare a Politicii: · Acceptare Socială: · Indicele Amprentei: -4.4 (R.S. -4.4)

Politica ON/OFF: Permiteți conversia terenurilor arabile în pajiști din LT

Această politică permite transformarea a 10% din terenurile arabile în pajiști în primii 12 ani de simulare.

Parametru	Valoare
Sector	Teren
Timp activ	35 ani
Cost de Implementare	Mediu
Acceptare Socială	Mediu

Figure 32. Translated policy instruments in the Simulation View (Romanian)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

2.6. Self-learning engine and AI algorithms

The self-learning engine is the core component of the NEPAT and a key research and development outcome of the NXG project. During the project, the self-learning engine has been employed as a recommendation tool for decision-making in policy package validation. Currently, it is integrated into NEPAT, where it provides policy package recommendations to users based on specific requests.

In the first case, the self-learning engine has been applied within a co-creation framework cycle to generate optimal policy packages based on a predefined set of policies, policy goals, and targets established by SHs in each CS (see Figure 33). Once generated, these policy packages were validated within the CSs and assessed by SHs during the fifth Workshops and adjustments were made accordingly.

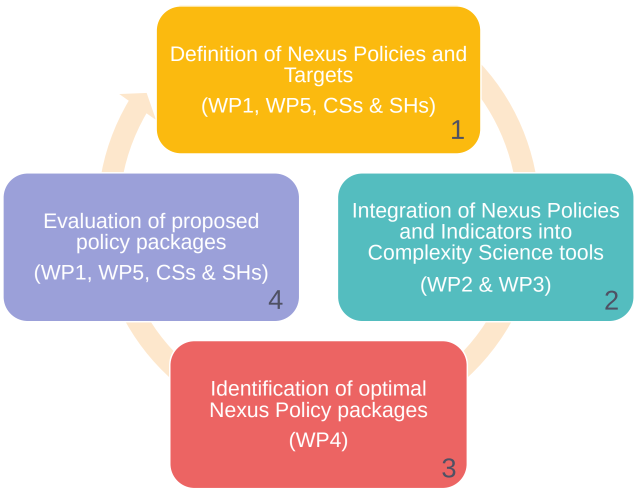

Definition of Nexus Policies and Targets
(WP1, WP5, CSs & SHs)
1
Integration of Nexus Policies and Indicators into Complexity Science tools
(WP2 & WP3)
2
Identification of optimal Nexus Policy packages
(WP4)
3
Evaluation of proposed policy packages
(WP1, WP5, CSs & SHs)
4

Figure 33. NXG co-creation framework for Nexus Policy packages identification

The second application of the self-learning engine is in the online DSS, which provides policy package recommendations to platform users.

This is a reduced version because these results have been/will be submitted for publication in a peer reviewed journal. For more information, please contact the authors - Nuria Nievas Viñals (Eurecat) nuria.nievas@eurecat.org.

2.7. Decision Support System

This is a reduced version because these results have been/will be submitted for publication in a peer reviewed journal. For more information, please contact the authors - Nuria Nievas Viñals (Eurecat) nuria.nievas@eurecat.org.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

2.8. Analytical Engine

The Analytical Engine plays a crucial role in supporting other SLNAE technical components by handling computational tasks efficiently. Its primary responsibility is to provide an environment equipped with the necessary Python libraries and hardware resources required for these tasks to run optimally. To achieve this, the Analytical Engine leverages Docker, using different requirements files tailored for specific tasks and selecting the appropriate base Docker images to ensure the required resources are available.

This is a reduced version because these results have been/will be submitted for publication in a peer reviewed journal. For more information, please contact the authors - Nuria Nievas Viñals (Eurecat) nuria.nievas@eurecat.org.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

3. Nexogenesis Data Sharing Tools

The Nexogenesis Data Sharing Tools constitute a digital ecosystem developed under Task T4.2 *Data Sharing Tools* to meet the internal and external data-sharing requirements of the NXG project. This ecosystem ensures robust system operations, including security, access management, and maintenance, while aligning with the project's commitment to Open Data and its DMP. It facilitates intra-project collaboration and external dissemination of knowledge, supporting Task T4.6 *Data Management Strategy*.

The NXG data pipeline defines the following roles (see Figure 34):

- **Data Providers**: WPs that generate data for use by other WPs.
- **Data Consumers**: WPs or external stakeholders that utilize data produced by providers.
- **Dual Role**: Certain WPs, such as WP3, function as both providers and consumers.

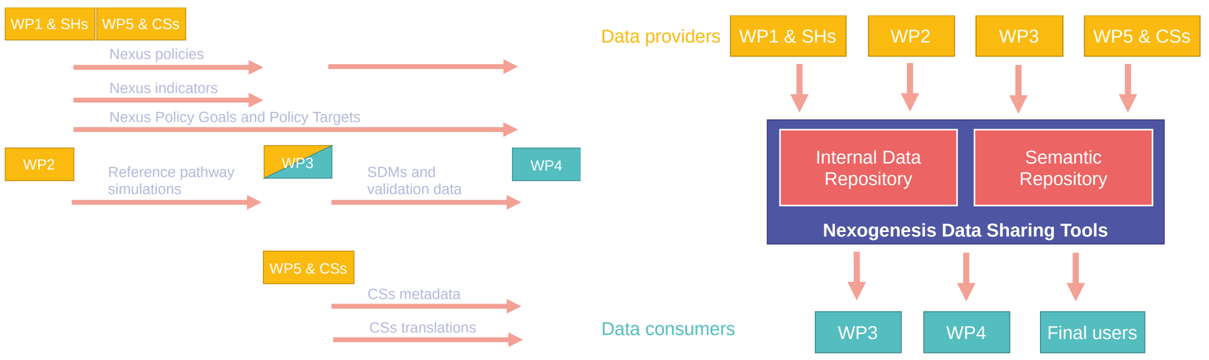


Figure 34. NXG cross-WP data pipelines in the Internal Data Repository

The **Knowledge Repository** serves as the core hub for data management in the NXG project. It is divided into two sub repositories:

Nexogenesis Data Lake (internal repository)

The Data Lake, implemented using Microsoft OneDrive, serves as a streamlined platform for data exchange between WPs, fostering seamless collaboration across project domains. It offers flexible schema-less data storage, enabling efficient file management without predefined structures. Its API-based retrieval capabilities support integration with other tools, while its scalable cloud infrastructure ensures reliable data synchronization and adaptability to varying project needs.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Nexogenesis Semantic Repository (external repository)

The Semantic Repository enhances the Knowledge Repository's capabilities by offering advanced semantic data management and visualization services, enabling data publication to external stakeholders and promoting nexus-related knowledge sharing. Its components include:

- **Nexogenesis Nexus Ontology**: A publicly accessible ontology that defines and organizes key concepts and relationships within the nexus framework.
  - **Access**: Nexus Ontology
- **SPARQL Server**: A query interface based on **Jena Fuseki**, enabling efficient retrieval and analysis of semantic data.
  - **Access**: SPARQL Server
- **Data Explorer and Visualizer**: An interactive platform implemented using **GraphDB**, providing tools for visualizing and analyzing complex data relationships through dashboards and graphical representations.
  - **Access**: Data Explorer and Visualizer

For further details, refer to Deliverable D4.2 *Data Lake for Data Sharing*.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

4. System Deployment

As previously discussed, NEPAT is a web-based tool composed of multiple interconnected components, as depicted in Figure 1. This section delves into the technical aspects of the development and deployment workflows that integrate these components into a cohesive system. It examines the architectural design, component interactions, and the methodologies employed to ensure seamless integration and robust performance.

4.1. Development and Deployment Environments

The development process was managed using Git[9], a free and open-source version control system, with all changes systematically tracked in a secure internal GitLab[10] repository. Git is designed to handle projects of all sizes with speed and efficiency, enabling robust tracking, seamless collaboration, and efficient branching and merging. To maximize development efficiency and minimize merge conflicts, a streamlined branching strategy was implemented. This strategy ensured clear separation of concerns for feature development, bug fixes, and releases, while facilitating smooth integration across different environments.

The project utilized three main branches to manage development and maintain a clear workflow:

- **Master/Main Branch**: This branch serves as the production-ready version of the project. All stable, tested, and approved features are merged into this branch, ensuring that it always represents the latest stable release of the software.
- **Development Branch**: All active development takes place here. New features, bug fixes, and enhancements are implemented within this branch. Periodic merges from the development branch to the master/main branch occur when features are completed and thoroughly tested, ensuring a smooth integration into the stable release.
- **UI Development Branch**: This branch is specifically dedicated to UI development, allowing front-end changes to be isolated from other development work. It enables focused UI improvements and testing without disrupting backend functionality. Once the UI changes are complete, they are merged into the development branch for further integration and testing.

As illustrated in Figure 35, these branches are carefully structured to ensure organized development, minimize conflicts, and streamline the integration process. Each branch serves a specific purpose, allowing for effective parallel development and maintaining the stability of the project throughout its lifecycle. While the primary branches cover specific

[9] https://git-scm.com/
[10] https://about.gitlab.com/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

responsibilities, temporary branches may be created from any development branch for larger changes that could cause conflicts if managed within the main branches.

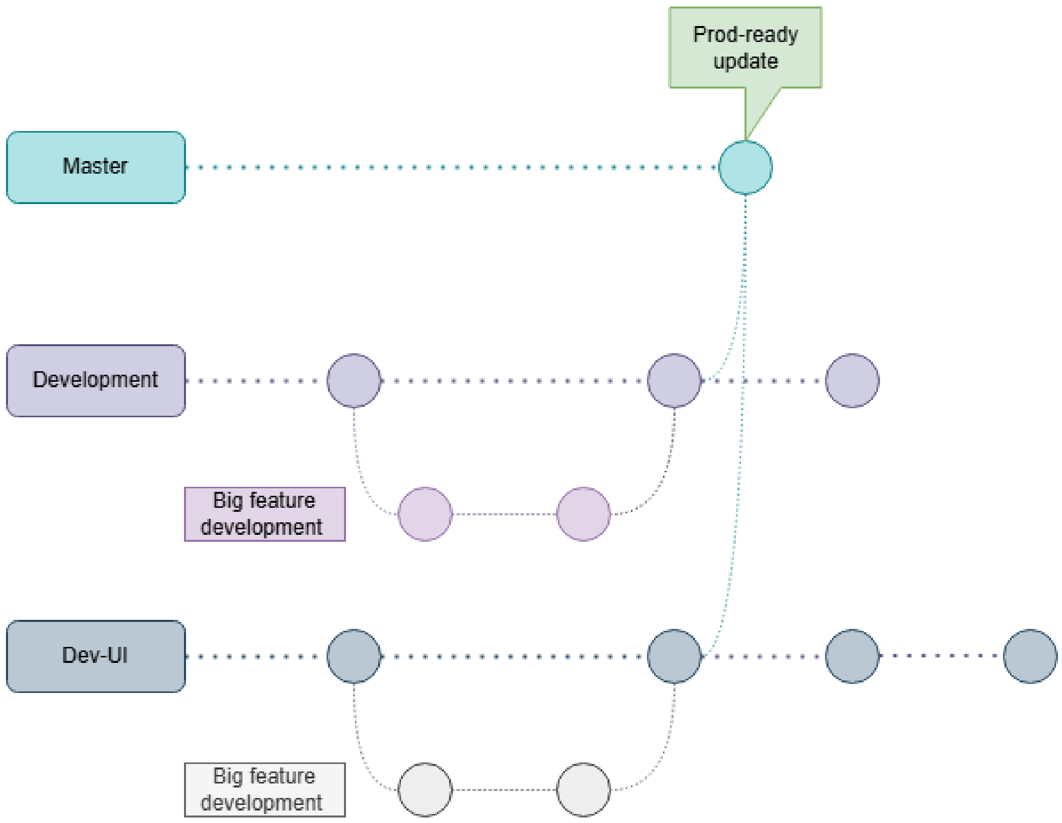


Figure 35. SLNAE Git Flow

This streamlined Git flow is complemented by a rigorous integration and testing process. Every new feature includes corresponding unit and integration tests before being pushed to GitLab. Upon push, the **CI/CD pipeline** is automatically triggered to run all tests, ensuring both new and existing components function correctly. If any tests fail, the team is immediately notified, and issues are promptly addressed, maintaining code stability throughout development.

The deployment of NEPAT involves two distinct environments to support both development and production workflows: **Dev** and **Pro**, as seen in the table. Dev is used for testing the latest features and validating API-GUI integration, while Pro hosts the stable, fully validated version for final deployment and user feedback.

Table 2. Deployment environments description

Aspect	Dev Environment	Pro Environment
Deployment server	Internal server.	Publicly accessible server.
Purpose	Testing interactions between API and GUI; partner demos.	Hosting the final, validated version for workshops & feedback.
Functionality	Latest features, integration testing, and minor issue fixes.	Stable, thoroughly tested version of NEPAT.
Pre-Testing	Pre-tested locally; newly found issues resolved in this environment.	Contains the stable release post full testing.
Audience	Developers, internal teams, and project partners	Project partners, stakeholders, and external users.
Updates	New features and functionalities are continuously added.	Only stable and validated versions deployed.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

4.2. Architecture

The architecture of the NEPAT tool is designed to provide high reliability, scalability, and streamlined deployment. Both the Development (Dev) and Production (Pro) environments adhere to the same architectural framework, as illustrated in Figure 36. The key distinction between these environments is their accessibility:

- The Dev environment is hosted on an internal server, ensuring restricted access to the development team and project partners for testing and validation purposes.
- The Pro environment is deployed on a public-facing server, enabling interaction with external users and stakeholders for final deployment and feedback collection.

All resources associated with the NXG project are centralized under the nexogenesis.eu domain, including the official project website at https://nexogenesis.eu. For NEPAT-related resources, two subdomains were established, reflecting the tool's names throughout the project lifecycle: https://nepat.nexogenesis.eu and https://slnae.nexogenesis.eu, both of which point to the Pro environment for public access.

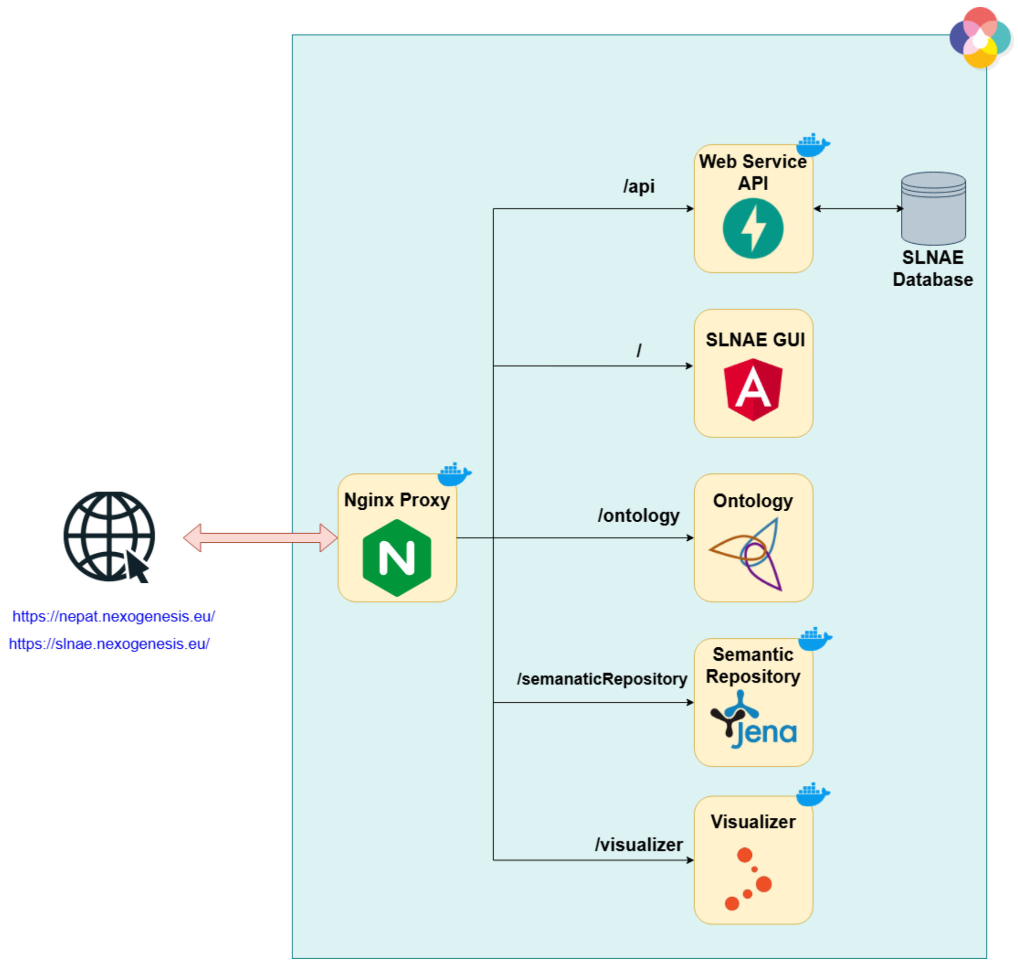


Figure 36. SLNAE Architecture

At the core of the architecture lies a proxy server that handles all incoming requests. Proxying is commonly used in modern architectures to act as an intermediary between clients and the

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

underlying services allowing to hide the internal structure from external clients, enabling load balancing, and allowing for better monitoring and logging of traffic.
In the NEPAT architecture, the proxy serves a dual purpose. First, it hosts and delivers the front-end Graphical User Interface to users, ensuring seamless access to the application. Second, it acts as an intermediary to redirect incoming requests to the appropriate back-end resources, all of which are deployed as Docker containers. This approach centralizes request handling and ensures that users can interact with the GUI while their actions are efficiently routed to the correct underlying service.

The Web Service API forms the backbone of the tool, implementing the business logic and managing data exchanges. It processes requests from the SLNAE GUI, returning data or triggering actions as required. The SLNAE GUI provides the user interface, delivering a responsive and intuitive user experience. In addition to the API and GUI, the architecture incorporates other of the project products like the Nexogenesis Ontology, Semantic Repository and the Nexogenesis Data Visualizer.
As seen in Figure 36, some of the components of the architecture are represented with the Docker logo, indicating that they are containerized. Docker[11] is a platform that automates the deployment, scaling, and management of applications within lightweight, portable containers. It is a cornerstone of the deployment strategy, ensuring consistency between environments by replicating identical runtime conditions and isolating services to prevent conflicts, simplifying both scaling and maintenance.
More specifically, for the Web Service API, which is subject to constant changes, Docker images with distinct tags for Dev and Pro environments are used. These tags include version identifiers and are saved in the container registry within the internal GitLab repository when necessary. This approach ensures quick rollbacks and allows for easy distinction between the versions deployed in Dev and Pro environments.

4.3. Hardware Resources

As outlined in previous sections, the Pro environment is deployed on a publicly accessible server, which hosts the architecture described earlier. This server is managed internally by Eurecat and is configured with the following specifications:

Table 3. NEPAT Pro Server Specifications

NEPAT Pro Server Specifications	
IP	84.88.76.21
Location	Barcelona, Catalonia, Spain.
OS	Ubuntu 22.04.5 LTS
CPU	8 Intel(R) Xeon(R) Gold 6442Y CPUs
RAM memory	15GB
Disk space	200GB

[11] https://www.docker.com/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

5. Testing and Validation

The NEPAT Platform schema, presented in section 2 (see Figure 1), outlines various modules, each designed to fulfill a specific responsibility within the system. To ensure the platform operates reliably and cohesively, testing every component is essential for identifying and resolving potential issues at every layer. Since the GUI relies on the API for data retrieval and manipulation, and the API depends on the Coordination Module for its core functionality, even a minor failure in one component could have cascading effects across the system.

Given the interdependence of these components, a multi-layered strategy was implemented to validate both API functionality and the underlying Coordination Module that supports it. The testing process focused on two key areas:

- **Web service API testing:** Comprehensive testing was conducted on all API methods to ensure they correctly handled incoming requests and returned accurate responses. This step was critical for verifying that the API seamlessly facilitated communication between the GUI and the Coordination Module while maintaining proper error handling and data consistency.
- **Coordination module testing:** The functionalities offered by the Coordination Module, which underpin the API's capabilities, were rigorously tested. This layer of testing ensured that the Coordination Module performed as expected, supported the API's operations, and provided reliable functionality for downstream processes.

This dual-layered testing approach guarantees that every interaction, whether initiated at the GUI level or within the backend, is fully validated. Detailed descriptions of the testing procedures for both the Web Service API and the Coordination Module are provided in the following sections.

Beyond functionality testing, the validation process extends to several other critical aspects to ensure the robustness and reliability of the NEPAT Platform:

- **Load testing:** Evaluates the system’s performance under varying levels of user demand to identify potential bottlenecks and ensure scalability.
- **Browser compatibility testing:** Verifies that the platform operates correctly across different web browsers to provide a consistent user experience.
- **Screen resolution testing:** Ensures the GUI is responsive and visually optimized for various screen sizes and resolutions.
- **Security testing:** Assesses vulnerabilities within the system to protect against potential threats, data breaches, and unauthorized access.
- **Resilience testing:** Tests the system’s ability to recover from failures and maintain functionality under adverse conditions.
- **Scalability testing:** Analyzes how well the platform can expand to accommodate increased user activity and data processing requirements.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

- **Interoperability testing:** Validates the system's capability to integrate with external applications and services seamlessly.

The following sections provide a detailed description of the testing procedures employed for each of these aspects, ensuring that the NEPAT Platform meets high standards of performance, security, and reliability.

5.1. Web Service API Testing

The Web Service API serves as the backbone of the NEPAT Platform, mediating communication between the GUI and the Coordination Module. As such, thorough testing of its endpoints is critical to guarantee seamless interactions and the accurate exchange of data. Following up the API description from section 2.3, this section provides a detailed account of the tests conducted on the Web Service API endpoints to evaluate their functionality, reliability, and robustness.

Each endpoint was tested to validate several key aspects:

- Functionality: Ensuring the endpoint performs its intended operations correctly.
- Data validation: Verifying that the input and output data conform to expected formats and constraints.
- Error Handling: Assessing the API's ability to manage unexpected inputs and conditions gracefully.

The tests were conducted using pytest[12], a robust and widely adopted Python testing framework. To simulate a real-world environment while maintaining controlled conditions, a dedicated testing database was utilized. This approach ensured that the tests could be performed in isolation without affecting the production database or live data.

Like the API description, the testing will be divided in three sections, corresponding to the main responsibilities of the Web Service API: authentication, simulation management and other functionalities.

5.1.1. Authentication

The API authentication endpoints support three methods of user authentication, each of which is tested in detail in this section.

Common user-password authentication

The first step in this authentication method is to register a new user account using a username, email and password. As such, the first endpoint to be tested is /auth/signup. As seen in Table 4, three tests were made to validate it works properly in a case where everything is correct and also ensure a correct management of each of the possible errors.

[12] https://docs.pytest.org/en/stable/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Table 4. Signup tests

Test name	Scenario	Expected Outcome	Notes
Test signup	User signs up with valid details.	Status code **200** Response: {"message": "User successfully registered!"} Email sent to the user.	Validates basic functionality and email notification on successful signup.
Test signup with existing email	User tries to sign up using an already registered email.	Status code **409** Only one email is sent from the initial registration.	Verifies duplicate email handling and prevents multiple accounts with the same email.
Test signup with missing parameters	User attempts to sign up with missing required fields (username, email or password)	Status code **422** No email sent.	Ensures the API enforces required parameters and returns appropriate error responses.

After the user is registered, it verifies its account, which triggers a call to the endpoint /auth/verify-email. This method was tested as detailed in Table 5.

Table 5. User verification tests

Test name	Scenario	Expected Outcome	Notes
Test verify user	User verifies its account providing a correct verification code.	Status code **200** User verified in the database.	Validates the basic functionality of the email verification system.
Test verify user already verified	User tries to verify its account more than once.	Status code **403** Response: {“detail”: “Invalid verification code or account already verified"}	Validates the verification is only done once, ensuring a consistent state in the database.
Test verify user with expired code	User tries to verify its account after code expiration.	Status code **403** Response: {“detail”: “Verification code expired"}	Validates that verification is only performed before the expiration time.

The email includes a button to resend the email in case the code has expired. This button makes an API call to /auth/resend-verification-email. This basic functionality is tested in Table 6.

Table 6. User verification email test

Test name	Scenario	Expected Outcome	Notes
Test resend verification email	User clicks on button to get send the verification email again.	Status code **200**. Response: {“message”: “Re sent verification email”}	Verifies the basic functionality of this endpoint.

After this thorough process, the user can log in using its credentials and making a call to /auth/login. To thoroughly ensure that the endpoint works correctly, many situations are tested: a correct login, a login with non-registered credentials, a login with unverified credentials and a login with an incorrect password. These tests are showcased in Table 7.

Table 7. User password login tests

Test name	Scenario	Expected Outcome	Notes
Test login	User logs in with correct and verifier	Status code **200**	Verifies the correct flow of a login operation.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

	email and password credentials.	Response: {"username":username, "access_token":access_token, "token_type":"Bearer"}	
Test login nonexistent user	User tries to log in with a non-registered email.	Status code **404** Response: {"detail": "User with email does not exist"}	Validates that only registered users can log in the tool.
Test login unverified user	User tries to log in with registered but not verified credentials.	Status code **403** Response: {"detail": "User not verified. Please, verify your account with the email received."}	Ensures only verified users can log in the tool.
Test login wrong password	A registered and verified user tries to log in but writes an incorrect password.	Status code **401** Response: {"detail": "Invalid details passed".}	Ensures that only users with correct credentials can log in.

Password recovery is also available in case the user forgets or loses their credentials. The method to be used is /auth/password-recovery by providing the email address and new password. To ensure the method works correctly three scenarios were tested (seen in Table 8): a successful scenario, a scenario where the email provided isn't registered and a scenario where the user didn't verify their account.

Table 8. Password recovery tests

Test name	Scenario	Expected Outcome	Notes
Test password recovery	User tries to recover their password by providing a valid email and a new password.	Status code **200** Response: {"message": "Password recovery email sent"}	Verifies the correct flow of a password recovery operation.
Test password recovery nonexistent user	User tries to recover their password with a non-registered email.	Status code **404** Response: {"detail": "The user with this email does not exist in the system."}	Validates that only registered users can recover their passwords.
Test password recovery unverified user	User tries to recover the password with a registered but not verified email.	Status code **403** Response: {"detail": "User not verified. Please, verify your account with the email received."}	Ensures only verified users can change their password.

Tests were also implemented for the process of password update. This corresponds to the endpoint /auth/reset-password, triggered by the button "Reset password" in the password recovery email. The tests are shown in Table 9.

Table 9. Password reset tests

Test name	Scenario	Expected Outcome	Notes
Test reset password	User tries to reset their password by providing a valid email and a new password.	Status code **200** Response: {"message": "Password updated successfully"}	Verifies the correct flow of a password reset operation.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Test reset password nonexistent user	User tries to reset their password with a non-registered email.	Status code **404** Response: {“detail”: “The user with this email does not exist in the system.”}	Validates that only registered users can reset their passwords.
Test reset password with invalid rest token	User tries to reset the password issuing an invalid token for the operation.	Status code **404** Response: {“detail”: "Invalid token"}	Ensures only the email owner can confirm the operation, being the only one with the valid token.

Guest Login

The guest login process is designed to be straightforward, where a request is made and a new user is provided automatically. Given its simplicity, the basic functionality of this process has been tested under normal circumstances as seen in Table 10, since there are no unforeseen errors.

Table 10. Guest login tests

Test name	Scenario	Expected Outcome	Notes
Test guest login	User logs in using the guest log in.	Status code **200** Response: {"username": “guest”, "access_token": access_token, "token_type": "Bearer"}	Verifies the correct flow of a guest login operation.

Login with Google

Similar to the guest login, the testing for the Google login is relatively straightforward. The Google authentication process relies primarily on the OAuth2 library provided by Google, which has already been thoroughly tested. As a result, the focus of our testing was not on the Google verification itself, but rather on validating the behavior of the rest of the endpoint. To achieve this, a mock was used for the authorization process, ensuring that the endpoint functions as expected without directly interacting with Google's OAuth2 service. The tests made are showcased in Table 11.

Table 11. Google login tests

Test name	Scenario	Expected Outcome	Notes
Test google login	User tries to log in using a Google account.	Status code **200** Response: {"username”: google_username, "access_token": access_token, "token_type": "Bearer"}	Verifies the correct flow of a Google login operation.

5.1.2. Simulation Management

As detailed in section 2.3.2, the most relevant functionalities of the Web Service API revolve around the Simulations management, which is implemented in the API through the methods of simulation wrappers and simulations. Access to these endpoints requires user authentication, which is managed by the SLNAE Core Service.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

The testing of the SLNAE Core Service will be covered in the following section. For the purposes of this testing phase, authentication was mocked to ensure the focus remained on validating the functionality of the API endpoints themselves. This approach allowed the testing environment to isolate and rigorously evaluate the specific operations and behaviors of the Web Service API without introducing dependencies on the authentication system. A similar approach was applied for the text translation, as the actual translation of entities is also handled by the SLNAE Core Service and will be tested separately, like the authentication.

Additional responsibilities of the SLNAE Core Service, such as running SDMs (via the SPF Core Service) and computing footprints, were also mocked for these tests. These functionalities impact key methods such as: POST /simulation-wrappers, GET /simulation-wrappers/{id}, POST /simulations, and POST /simulations/{id}/run.

This section provides a comprehensive account of the testing conducted on each endpoint, detailing the methods and results used to validate their functionality, reliability, and robustness. For more details on the description and parameters of each of the overviewed endpoints see Annex I: Simulation endpoints.

Firstly, there is the GET request to /simulation-wrappers, used to obtain a list of user simulation wrappers. For this endpoint, a scenario where no simulation wrappers existed was tested to confirm that the API returns an empty list, and another scenario with a few existing wrappers was tested to ensure proper retrieval and formatting of data as seen in Table 12.

Table 12. Simulation wrappers retrieval tests

Test name	Scenario	Expected Outcome	Notes
Test retrieve simulation wrappers empty	The user tries to retrieve the list of their simulation wrappers without having any simulation wrapper created before.	Status code **200** Response: []	Validates that an empty list is returned when no persisted simulation wrappers of the user exist.
Test retrieve simulation wrappers	The user tries to get the list of their simulation wrappers after having created some persisted simulations.	Status code **200** Response: [{“id”: …}, {“id”: ..}]	Checks that all the user’s persisted simulation wrappers are retrieved using this method.

Simulation wrappers can also be created by running a POST request to /simulation-wrappers. This method requires providing the name of the wrapper and, for each simulation it contains, a case study identifier and a scenario identifier. As simulation wrappers serve as high-level containers for grouping multiple simulations, this section focuses exclusively on testing functionalities specific to this method. Tests for creating individual simulations, which are included within simulation wrappers, will be addressed later in this section. Details of the specific tests for the /simulation-wrappers endpoint are outlined in Table 13.

Table 13. Simulation wrappers creation tests

Test name	Scenario	Expected Outcome	Notes
Test create wrapper	The user tries to create an empty simulation wrapper providing the name and description.	Status code **200** Response: new simulation wrapper	Validates the basic functionality of this endpoint.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

For persisted simulation wrappers, users can make a GET request to /simulations-wrappers/{id} to fetch the information of a specific saved simulation wrapper. Tests for this method covered both a successful scenario as well as some error cases, including users attempting to access a simulation wrapper owned by another user or providing a non-existent wrapper ID. Details of these tests are outlined in Table 14.

Table 14. Simulation wrapper retreival tests

Test name	Scenario	Expected Outcome	Notes
Test retrieve simulation wrapper	The user tries to retrieve a simulation wrapper, providing correct request parameters.	Status code **200** Response: simulation wrapper	Validates the correct functioning of the endpoint under normal circumstances.
Test retrieve non-existent simulation wrapper	The user tries to retrieve a simulation wrapper but provides a simulation id that doesn't exist.	Status code **404** Response: {"detail": "Simulation wrapper with specified id does not exist"}	Validates the corresponding error is returned if the simulation wrapper id is incorrect.
Test retrieve simulation wrapper of another user.	The user tries to retrieve a simulation wrapper but provides the id of a simulation where the owner is another user.	Status code **403** Response: {"detail": "Must be the owner of the Simulation wrapper!"}	Ensures that users can only retrieve their own simulation wrappers.

The deletion of simulation wrappers was tested using the DELETE method on /simulation-wrappers/{id}. Tests (see Table 15) confirmed that users could delete only their own simulation wrappers and that appropriate errors were returned when attempting to delete a wrapper they did not own or one that was not present in the database.

Table 15. Simulation wrapper deletetion tests

Test name	Scenario	Expected Outcome	Notes
Test delete simulation wrapper	The user tries to delete a simulation wrapper, providing correct request parameters.	Status code **200** Response: recommendations list	Validates the correct functioning of the endpoint under normal circumstances.
Test delete non-existent simulation wrapper	The user tries to delete a simulation wrapper but provides a simulation id that doesn't exist.	Status code **404** Response: {"detail": "Simulation wrapper with specified id does not exist"}	Validates the corresponding error is returned if the simulation wrapper id is incorrect.
Test delete simulation wrapper of another user.	The user tries to delete a simulation wrapper but provides the id of a simulation where the owner is another user.	Status code **403** Response: {"detail": "Must be the owner of the Simulation wrapper!"}	Ensures that users can only delete their own simulation wrappers.

Modifications to simulation wrappers were tested using the PUT method on /simulation-wrappers. The tests for this endpoint (Table 16) validated that changes to the wrapper's name and description, as well as updates to the simulations list, were correctly reflected in the database. Error cases included users attempting to modify wrappers they did not own or providing invalid wrapper IDs.

Table 16. Simulation wrapper modification tests

Test name	Scenario	Expected Outcome	Notes
Test update simulation	The user provides a valid simulation wrapper id and	Status code **200**	Checks name and description can be successfully modified.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

wrapper name and description	tries to modify its name and description.	Response: {"message": "Simulation wrapper updated successfully!"}	
Test update simulation wrapper simulations	The user provides a valid simulation wrapper id and tries to modify the list of simulations wrapped.	Status code **200** Response: {"message": "Simulation wrapper updated successfully!"}	Checks that simulations are updated with the old ones deleted and the new ones saved in the wrapper.
Test delete non-existent simulation wrapper	The user tries to retrieve a simulation wrapper but provides a simulation id that doesn't exist.	Status code **404** Response: {"detail": "Simulation wrapper with specified id does not exist"}	Validates the corresponding error is returned if the simulation wrapper id is incorrect.
Test delete simulation wrapper of another user.	The user tries to retrieve a simulation wrapper but provides the id of a simulation where the owner is another user.	Status code **403** Response: {"detail": "Must be the owner of the Simulation wrapper!"}	Ensures that users can only modify their own simulation wrappers.

For the simulation endpoints, the creation of simulations was tested using the POST method on /simulations. Here, different possible scenarios were considered with different parameters available upon creation, with details provided in Table 17.

Table 17. Simulation creation tests

Test name	Scenario	Expected Outcome	Notes
Test create Simulation	The user tries to create a simulation specifying correct case study and scenario ids.	Status code **200** Response: new simulation	Checks the correct creation of simulations with valid parameters.
Test create simulation with missing parameters	The user tries to create a simulation but doesn't provide the case study id or scenario id.	Status code **422** Response: {"detail": {"loc":..}, {"msg":"missing param..",..}}	Validates the correct error is generation when a required parameter isn't provided.
Test create simulation with invalid parameters	The user tries to create a simulation with a non-existent case study id or scenario id.	Status code **404** Response: {"detail": "Provided a nonexistent id!"}	Validates the correct error is generation when a parameter is incorrect.
Test create simulation with fixed goals year	The user attempts to create a simulation and specifies a target year to filter all case study goals accordingly.	Status code **200** Response: new simulation with filtered policy goals	Validates that the goals year parameter functions correctly by returning a simulation containing only the policy goals specified for that year.
Test create simulation with custom goals	The user tries to create a simulation specifying a custom goal.	Status code **200** Response: new simulation with case study and custom goals.	Validates custom goals are correctly created with the simulation.
Test create simulation with custom goal missing parameters	The user tries to create a simulation with a custom goal but it has some required parameters missing.	Status code **422** Response: {"detail": {"loc":..}, {"msg":"missing param..",..}}	Validates the correct error is generation when a required parameter isn't provided.
Test create simulation with custom goal invalid parameters	The user tries to create a simulation with a custom goal that has a non-existent indicator.	Status code **404** Response: {"detail": "provided a non-existent goal indicator"}	Validates an error is generated when a non-existent parameter is included in the request.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

The execution of simulations was tested using the POST method to /simulations/{id}/run. Tests (Table 18) ensured that simulation executions were only made with the correct parameters and authorship and existence of the simulation. As explained in the beginning of this section, the actual running of SDMs is performed by the Core service and was mocked in these tests since will be tested separately in the following section.

Table 18. Simulation run tests

Test name	Scenario	Expected Outcome	Notes
Test run simulation	The user runs a simulation with different correct run configurations.	Status code **200** Response: { "var_tree_values": {}, "footprint_values": {}, "footprint": 0}	Tests basic functionality of the endpoint.
Test run simulation with missing parameters	The user runs a simulation without providing the policy package.	Status code **422** Response: {“detail”: {“loc”:..}, {“msg”:“missing param..”,..}}	Validates the appropriate error is generated when required parameters are missing.
Test run simulation with incompatible policies	The user runs a simulation with a policy package that contains incompatible policies.	Status code **422** Response: {“detail”: "Incompatible policies applied at the same time"}	In case there are policy incompatibilities, this test ensures the API identifies them and returns an appropriate error.
Test run simulation with invalid policy package	The user runs a simulation with a policy package where a policy is applied in a year that isn’t between 2015 and 2050.	Status code **422** Response: {“detail”: " Year of application must be between 2015 and 2050"}	Validates an error is generated when incorrect parameters that may cause errors in the simulation run are provided.
Test run simulation with non-existent simulation id	The user tries to run a simulation but provides a simulation id that doesn’t exist.	Status code **404** Response: {“detail”: “Simulation with specified id does not exist”}	Validates the corresponding error is returned if the simulation id is incorrect.
Test run simulation of another user	The user tries to run a simulation but provides the id of a simulation where the owner is another user.	Status code **403** Response: {“detail”: “Must be the owner of the Simulation!”}	Ensures that users can only manage their own simulations.

Finally, the POST method to /simulations/{id}/advice was tested to validate the API’s ability to provide recommendations. Based on the endpoint definition, both success and possible errors were tested. The interaction with the DSS is handled by the SLNAE Core service, which will be tested in the following section along with the other functionalities discussed.

Table 19. Simulation advice tests

Test name	Scenario	Expected Outcome	Notes
Test get advice	The user tries to get advice from the DSS, providing correct request parameters.	Status code **200** Response: recommendations list	Validates the correct functioning of the endpoint under normal circumstances.
Test get advice with non-existent simulation id	The user tries to get advice from the DSS but provides a simulation id that doesn’t exist.	Status code **404** Response: {“detail”: “Simulation with specified id does not exist”}	Validates the corresponding error is returned if the simulation id is incorrect.
Test get advice with a	The user tries to get advice from the DSS but provides	Status code **403**	Ensures that users can only manage their own simulations.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

simulation id of another user	the id of a simulation where the owner is another user.	Response: {“detail”: “Must be the owner of the Simulation!”}	

5.1.3. Other functionalities

The API includes several key functionalities that support the core features of the NEPAT platform. Like with the simulation management, all these endpoints require user authentication. Each endpoint was thoroughly tested to ensure proper data retrieval and feedback submission.

Firstly, the /case-studies, which provides a list with information about the available case studies, was tested to verify that it returns accurate and complete data in every possible scenario as seen in Table 20.

Table 20. Case study retrieval tests

Test name	Scenario	Expected Outcome	Notes
Test retrieve case studies empty	The user tries to retrieve information about case studies.	Status code **200** Response: []	Validates that an empty list is returned when no case studies exist in the database.
Test retrieve case studies	The user tries to get the case studies information.	Status code **200** Response: [{“id”: …}, {“id”: ..}]	Checks that when the database is not empty, a list of case studies’ information is returned.
Test retrieve case study details	The user tries to retrieve case study details, but now with the query parameter detail set to true.	Status code **200** Response: [{“id”: .., “scenarios”: …, “policy_goals”:…}, …]	Validates that when “detail” is set to true a list of extended (including scenarios and policy goals) case study information is returned.

The /case-studies/{id}/scenarios/{scenario_id}/indicators endpoint enables users to retrieve scenario-specific SDM variables associated with a given case study and scenario combination. This functionality depends on the SLNAE Core Service to fetch the list of variables. To ensure the endpoint's functionality is tested in isolation, a mock was utilized to simulate the behavior of the Core Service. The tests made are the ones shown in Table 21.

Table 21. Case study-scenario variable retrieval tests

Test name	Scenario	Expected Outcome	Notes
Test retrieve SDM variables	The user tries to retrieve the list of SDM variables of a valid case study-scenario combination.	Status code **200** Response: [“var1”, “var2”]	Validates that a list of SDM variables is returned for the specified case study and scenario.
Test retrieve SDM variables not found	The user tries to retrieve the list of SDM variables of a nonexistent case study-scenario combination.	Status code **404** Response: {"detail": "Provided a non-existent cs scenario combination"}	Validates the appropriate error is raised when an invalid case study scenario combination is requested.

Finally, the endpoint to provide feedback /bug-reports is tested to ensure it works correctly when correct parameters are given and also validate appropriate error messages are given in other cases as seen in Table 22.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Table 22. Feedback endpoint tests

Test name	Scenario	Expected Outcome	Notes
Test submit bug report	The user tries to submit a bug report providing the correct parameters	Status code **200** Response: {"message": "Bug report created successfully!"} Email with report generated and sent to the administrator.	Checks that the reports are correctly generated when correct parameters are given.
Test submit bug report missing parameters	The user tries to submit a bug reports but doesn't provide the title or the report parameter.	Status code **422** Response: {"detail": {"loc":..}, {"msg":"missing param..",..}} Email not generated	Validates the correct error is generated in case there's any missing required parameter.

5.2. Core Service Testing

The Core Service has strong dependencies with the API as they work in union to provide the responses to the SLNAE GUI. Section 2.1 provides a detailed overview of the Core Service components and their interactions. This section will focus on the testing procedures for each Core Service component that was mocked during the API tests and plays a critical role in the system, following the same structured approach as Section 2.1.

Data Manager

The Data Manager is responsible for loading and processing key components such as policies, goals, targets, and indicators from the provided Excel files. This process primarily involves reading and structuring the extracted data for further use within the system.
To ensure the correctness of this functionality, a test (Table 23) was conducted to verify that the data loading process works as expected. Specifically, the test checks whether the policy cards, policy goals, and nexus indicators are correctly initialized when loading data for a given case study.

Table 23. Data Manager tests

Test name	Scenario	Expected Outcome	Notes
Load data test	Load data for a given SDM name	Policy cards, policy goals, and nexus indicators are correctly assigned	Ensures that the Data Manager correctly processes and stores the loaded data

Text Translator

Text translation primarily relies on the Google API for processing. However, it also includes validation of the selected languages and proper handling of errors related to the translation file or API. Table 24 introduces the set of tests used to validate the translation process. These tests ensure proper language selection, accurate handling of errors related to translation files, and the robustness of the Google API integration.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Table 24. Text translation tests

Test name	Scenario	Expected Outcome	Notes
Load Languages Successfully	JSON file with valid language codes is provided	Returns a dictionary of languages	Ensures correct language file parsing
Handle Missing JSON File	Language JSON file is missing	Raises ValueError	Ensures proper error handling for missing file
Handle Invalid JSON Format	JSON file contains invalid structure	Raises ValueError	Ensures the JSON file has the correct format
Translate to Multiple Languages	Translate text with multiple language codes	Returns translated text for each language	Ensures correct translation results
Handle Translation API Failure	Translation service is unavailable	Raises ValueError	Ensures error handling for API failures

Authentication and Authorization Module

For the Authentication and Authorization Module, tests were performed for each of the three responsibilities it holds: JWT token generation, validation, and password encryption and validation.

For the validation of JWT generation, the tests outlined in Table 25 were executed to verify the correct creation of tokens, proper handling of expiration time, and token uniqueness. Regarding token validation, since it is utilized as a FastAPI dependency, both valid scenarios and various error scenarios (as detailed in Table 26) were tested to ensure the appropriate HTTP exceptions are triggered. For password encryption and verification, both the encryption process and the verification mechanism were rigorously tested, as demonstrated in Table 27.

Table 25. JWT generation tests

Test name	Scenario	Expected Outcome	Notes
Generate Token for Valid User	A valid username is provided	A JWT token is generated successfully	The token should contain the correct user information and expiration
Verify Token Structure	A token is generated for a user	The token follows the standard JWT structure	Ensure the token is a valid JWT format
Expiration Claim Validation	A token is generated	The expiration time is correctly set (e.g., 48h from creation)	Decode token and check the expires claim
Ensure Token Signing	Generate a token and verify its integrity	The token signature is valid when decoded with the secret key	A tampered token should fail validation
Token Uniqueness	Generate multiple tokens for the same user at different times	Each generated token should be unique	Ensure token regeneration is independent

Table 26. JWT verification tests

Test name	Scenario	Expected Outcome	Notes
Valid Token Verification	A valid JWT is provided	The function returns the decoded payload successfully	Ensure claims like user and expires are correct
Token Without Expiration	A token missing the expires claim is provided	Exception 400 Bad request is raised	Ensure missing expiration is detected

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Expired Token	A JWT with a past expiration time is used	Raises HTTP exception 403 Forbidden - Token expired!	Test expiration logic with an expired timestamp
Tampered Token	A JWT with a modified payload is provided	Raises 400 Bad Request - Invalid Token	Test signature verification
Token with Wrong Signature	A token signed with a different secret key is used	Raises 400 Bad Request - Invalid Token	Ensure tokens signed with the wrong key fail

Table 27. Password management tests

Test name	Scenario	Expected Outcome	Notes
Password Hashing	A user provides a password for hashing	A hashed string is returned	Hash should be non-empty and different from the plain password
Password Verification (Correct Password)	A user provides the correct password for verification	Function returns True	Ensure the password is correctly verified
Password Verification (Incorrect Password)	A user provides an incorrect password	Function returns False	Ensure verification fails when wrong password is used
Different Hashes for Same Password	A user hashes the same password twice	The generated hashes should be different	Ensures salting mechanism is working

Simulation Utilities Module

For the Simulation Utilities Module, tests were also conducted for each of the tasks it is responsible for: SDM variable list retrieval, SDM execution and Footprint calculations.

The SDM variable list retrieval, it is performed by dynamically loading a JSON file based on the provided SDM name. Tests were conducted to verify that the correct file is loaded, the fallback mechanism works as intended, and errors such as missing or malformed files are properly handled as seen in Table 28.

Table 28. SDM variable retrieval tests

Test name	Scenario	Expected Outcome	Notes
Test retrieve SDM variables	Try to retrieve the SDM variables list for an existing SDM name	Returns a dictionary of variables	Ensures the correct file with the variables is loaded.
Test retrieve SDM variables failed	Try to retrieve the SDM variables list with incorrect SDM name	Raises FileNotFoundError	Ensures proper error handling when the SDM asked for doesn’t exist.
Invalid variables file format	The JSON file with the variables is malformed	Raises JSONDecodeError	Ensures JSON integrity

To run simulations, the Core Service executes the appropriate SDM based on the simulation configuration, considering the number of runs, random seed, and whether a baseline execution is required. Tests were conducted to ensure that the correct SDM file is selected based on the number of runs, the appropriate translation file is applied, and the baseline execution works as expected when enabled. These are detailed in Table 29.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Table 29. Simulation run tests

Test name	Scenario	Expected Outcome	Notes
Test single run without random seed	Execute simulation with a single run and no random seed.	The function should return a dictionary with a key "PP", including the run results of the provided policy package.	Ensures a single run was made with the provided policies and using the SDM average scenario.
Test multiple runs without random seed	Execute simulation with multiple runs (3) and no random seed.	The result should be a list of 3 quartiles (Q1, Q2, Q3), and sdm_mode="normal" should be called.	Ensures a single run was made with the provided policies and using the SDM original scenario.
Test multiple runs with random seed	Execute simulation with multiple runs (3) and a random seed.	The result should contain lists and quartiles, with the correct use of the random seed in run_sdm.	Verifies random seed usage and result format.
Test invalid SDM name	Execute simulation with an invalid SDM name.	The function should raise an exception due to an invalid SDM name or configuration.	Tests error handling for invalid input.

For the footprint calculations, as mentioned earlier, the normalization used was provided by JAWS, the testing therefore was focused on the integration of the provided normalization, without delving into testing the normalization logic itself. The tests made are the following:

Table 30. Footprint computations tests

Test name	Scenario	Expected Outcome	Notes
Test compute footprint results	Given the run results of an SDM, get footprint normalization	Returns the footprint pillars, sub-pillars and indicator values.	Test the correct integration with the SDM run results.
Test compute footprint with empty run results	Run footprint calculations with empty run results	The function raises a ValueError with the message "run_results should not be empty."	Checks that the according error is returned when run results are not provided.

DSS Manager

The DSS Management made by the Core Service was also thoroughly tested. Specifically, tests include two scenarios: a test for obtaining recommendations for both the policy goals and the footprint. These are detailed below in Table 31.

Table 31. DSS Manager tests

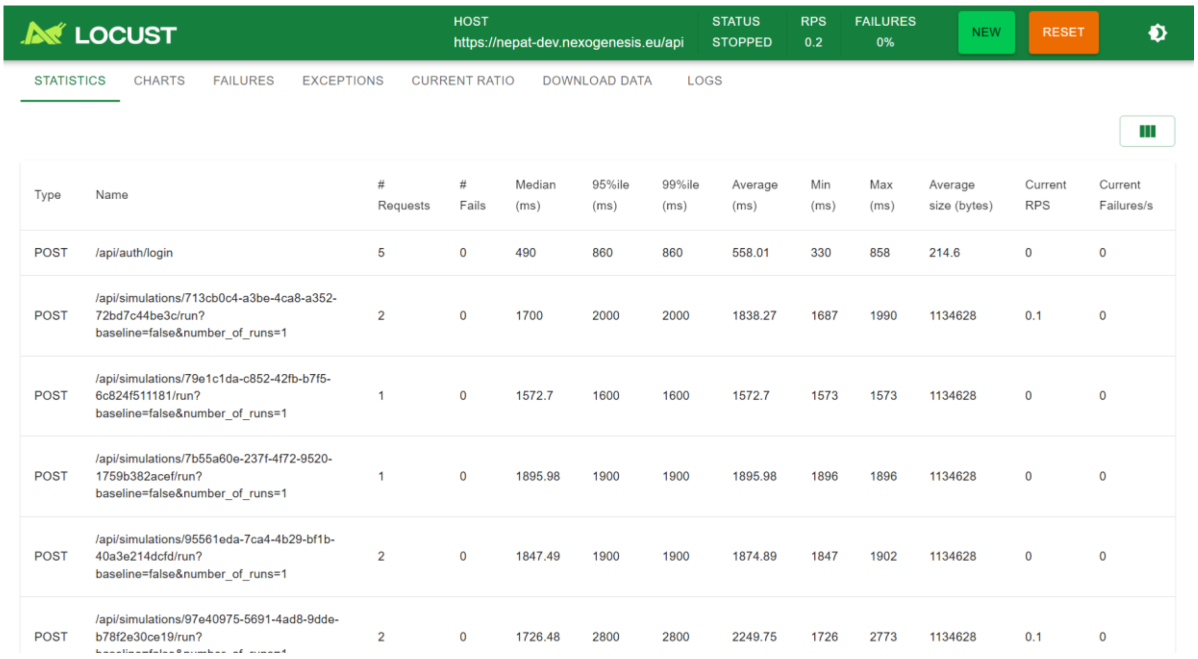
Test name	Scenario	Expected Outcome	Notes
Test get advice	The user asks for recommendations considering some policy goals	Returns a list of recommendations including the policies, overall achievement, individual policy goal achievement and tradeoffs and synergies.	Test the correct management of the different engines of the DSS.
Test get advice for footptint indicators	The user asks for recommendations considering some footprint indicators		

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

5.3. Load Testing

Load testing is a crucial aspect of performance evaluation that assesses how the NEPAT Platform behaves under varying levels of demand. By doing these tests, a good approximation of how many users can the system hold concurrently is obtained. During the lifetime of the project, load testing was given special attention, specially before workshops to ensure a smooth interaction with the tool.

To perform the load testing, the framework Locust[13] was used. Locust offers a scalable and efficient way to simulate user interactions, enabling comprehensive assessment of system performance under realistic workloads. A key advantage of this framework is its simplicity—it allows tests to be set up using a straightforward Python script while providing an intuitive interface for extracting and analyzing performance statistics (Figure 37). This ease of use makes Locust an effective tool for evaluating system scalability and identifying potential bottlenecks.



Type	Name	# Requests	# Fails	Median (ms)	95%ile (ms)	99%ile (ms)	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	Current RPS	Current Failures/s
POST	/api/auth/login	5	0	490	860	860	558.01	330	858	214.6	0	0
POST	/api/simulations/713cb0c4-a3be-4ca8-a352-72bd7c44be3c/run?baseline=false&number_of_runs=1	2	0	1700	2000	2000	1838.27	1687	1990	1134628	0.1	0
POST	/api/simulations/79e1c1da-c852-42fb-b7f5-6c824f511181/run?baseline=false&number_of_runs=1	1	0	1572.7	1600	1600	1572.7	1573	1573	1134628	0	0
POST	/api/simulations/7b55a60e-237f-4f72-9520-1759b382acef/run?baseline=false&number_of_runs=1	1	0	1895.98	1900	1900	1895.98	1896	1896	1134628	0	0
POST	/api/simulations/95561eda-7ca4-4b29-bf1b-40a3e214dcfd/run?baseline=false&number_of_runs=1	2	0	1847.49	1900	1900	1874.89	1847	1902	1134628	0	0
POST	/api/simulations/97e40975-5691-4ad8-9dde-b78f2e30ce19/run?	2	0	1726.48	2800	2800	2249.75	1726	2773	1134628	0.1	0

Figure 37. Locust dashboard

In the context of NEPAT, most user interactions with the graphical user interface (GUI) involve:

- Creating and running simulations – The primary function of the platform, which users frequently execute.
- Requesting recommendations from the Decision Support System – An action performed occasionally but still important for system evaluation.

User behavior in real-world scenarios is inherently unpredictable, as interaction frequency varies among users. However, for load testing purposes, we simulate a worst-case scenario by

[13] https://locust.io/

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

modelling users who continuously perform all key interactions. This approach ensures that system limits are effectively tested.

With Locust, load tests are configured by defining the number of concurrent users and specifying an interaction flow that they continuously execute until the test completes. For NEPAT, the user interaction flow was structured as follows:

1. Authenticate – Log into the system.
2. Create a simulation – Generate a new simulation instance.
3. Run the simulation – Execute the simulation 10 times to simulate real-world computational demand.
4. Request a DSS recommendation – Query the Decision Support System for recommendations.

To ensure a robust performance evaluation, load tests were executed across all case studies within the NEPAT platform. This approach allowed for a comprehensive assessment of system performance under varying conditions, ensuring that optimizations and scalability improvements were based on real-world workloads. The parameters used for each of the tests are described in Table 32.

Table 32. Load tests description

Test	Number of users	Ramp up (users started/s)	Duration
Test 1	10	1	5m
Test 2	50	1	5m
Test 3	100	1	5m

To test the limits of the system and evaluate the response times with different scenarios, tests were conducted for 10, 50 and 100 users. The ramp up and duration of the test were configured such that the peak of requests is reached and maintained for some time, to have good metrics on the average times per type of request.
The results of the load tests, summarized in Table 33, Table 34, and Table 35 provide valuable insights into the system's performance under different conditions.

Table 33. Load test 1 average response times

	Nestos	Lielupe	Jiu	Adige	Inkomati
Login	0.48 s	0.46 s	0.47 s	0.46 s	0.38 s
Simulation creation	1.08 s	1.83 s	1.25 s	0.94 s	1.22 s
Simulation run	3.88 s	2.34 s	2.09 s	1.74 s	1.95 s
DSS recommendation	12.42 s	11.40 s	2.09 s	2.16 s	12.48 s

Table 34. Load test 2 average response times

	Nestos	Lielupe	Jiu	Adige	Inkomati
Login	0.63 s	0.69 s	0.42 s	0.42 s	0.58 s
Simulation creation	3.45 s	3.78 s	1.42 s	1.02 s	3.56 s
Simulation run	37.56 s	13.44 s	2.85 s	2.43 s	11.64 s
DSS recommendation	39.79 s	35.64 s	3.15 s	3.38 s	33.10 s

Table 35. Load test 3 average response times

	Nestos	Lielupe	Jiu	Adige	Inkomati
Login	3.51 s	3.84 s	0.42 s	2.31 s	4.33 s
Simulation creation	7.89 s	7.83 s	4.94 s	3.05 s	8.56 s
Simulation run	43.39 s	25.79 s	17.32 s	15.26 s	25.55 s
DSS recommendation	32.48 s	41.41 s	19.53 s	15.56 s	47.93 s

When analyzing the response times across different endpoints and considering the average across all case studies, a clear trend emerges where some methods experience a significant increase in response times as the number of users increases. As shown in Figure 38, the login and simulation creation do not show a major increase unless the number of concurrent users reaches 100. This indicates that the authentication process is not significantly affected by increased demand, due to its relatively lightweight nature compared to other operations. Similarly, simulation creation is a process that is optimized through caching, particularly for each unique case study scenario combination, which reduces the computational load and makes it less resource-intensive and more scalable.

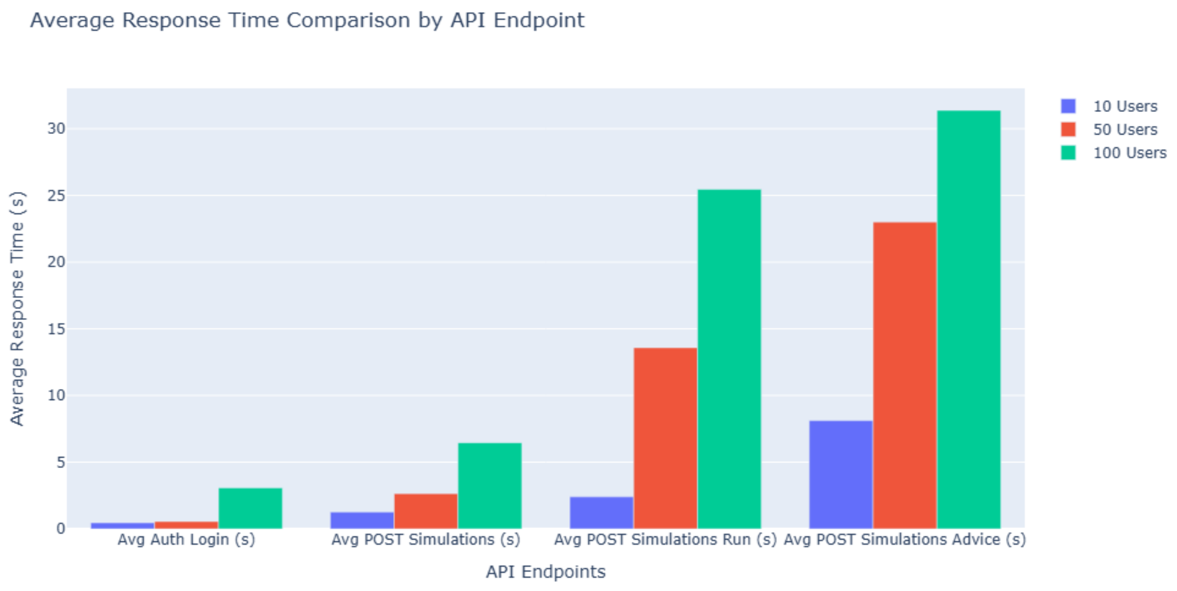


Figure 38. Average response time comparison by API endpoint

In contrast, the time required for simulation management increases significantly as more users are added, as illustrated in Figure 38. This suggests that the computational load associated with creating and running simulations places a heavier burden on system resources, leading to longer response times when multiple users are interacting with the platform simultaneously. This trend was expected, as the execution of simulations involves both processing and data storage operations, both of which can become limiting factors as demand scales up.

A further comparison across case studies, depicted in Figure 39, highlights an interesting observation: when summing the average response times for each step in the user interaction

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

flow, i.e. calculating the average time to complete the initially presented user flow, a variation emerges depending on the case study.

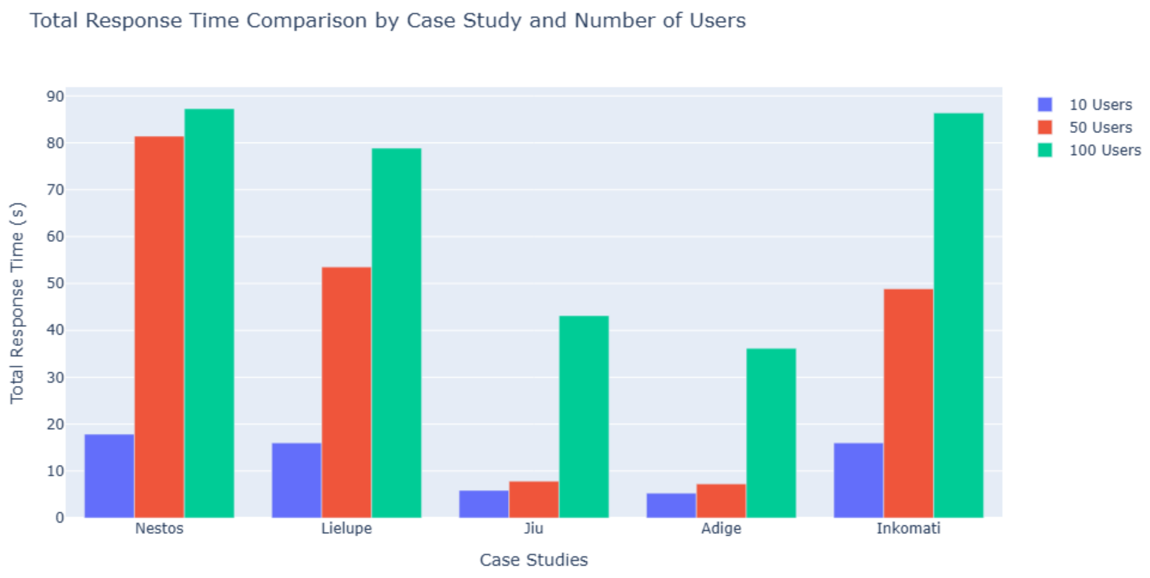


Figure 39. Total user flow time comparisona cross case studies

This discrepancy is particularly evident with smaller numbers of users, although it persists across different user counts. Notably, Jiu and Adige demonstrate faster response times, while others, such as Nestos, exhibit slower performance. This is a reasonable outcome given the differing complexity of the underlying SDMs. For example, Nestos, which features thousands of variables, naturally requires more processing time than other case studies with only a few hundred variables. The increased complexity of such models translates into longer execution times and, consequently, a slower completion of the user interaction flow.

The results overviewed in this section represent a worst-case scenario, where the system is under maximum load with all users performing intensive actions simultaneously. In reality, such extreme conditions are unlikely to occur frequently, and performance is expected to be better under typical usage patterns. However, these results provide valuable insight into the system's performance under stress, helping to identify potential limitations in extreme situations. Overall, the observed metrics seem reasonable and align with the current project's requirements. Therefore, no immediate changes to the server infrastructure are expected, unless future demands necessitate adjustments.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

5.4. Browser Compatibility

The NEPAT GUI has been developed using Angular, a popular framework for building dynamic, single-page applications. According to the official Angular documentation, the website supports a range of modern web browsers that adhere to current web standards. Specifically, the application is compatible with the following browsers:

Table 36. Angular 15 browser compatibility

Browser	Supported versions
Chrome	2 most recent versions
Firefox	latest and extended support release (ESR)
Edge	2 most recent major versions
Safari	2 most recent major versions
iOS	2 most recent major versions
Android	2 most recent major versions

This ensures that users accessing the website through these browsers will have an optimal experience, with full functionality and performance. Legacy browsers, such as older versions of Internet Explorer, are not supported due to their lack of support for essential modern web features.

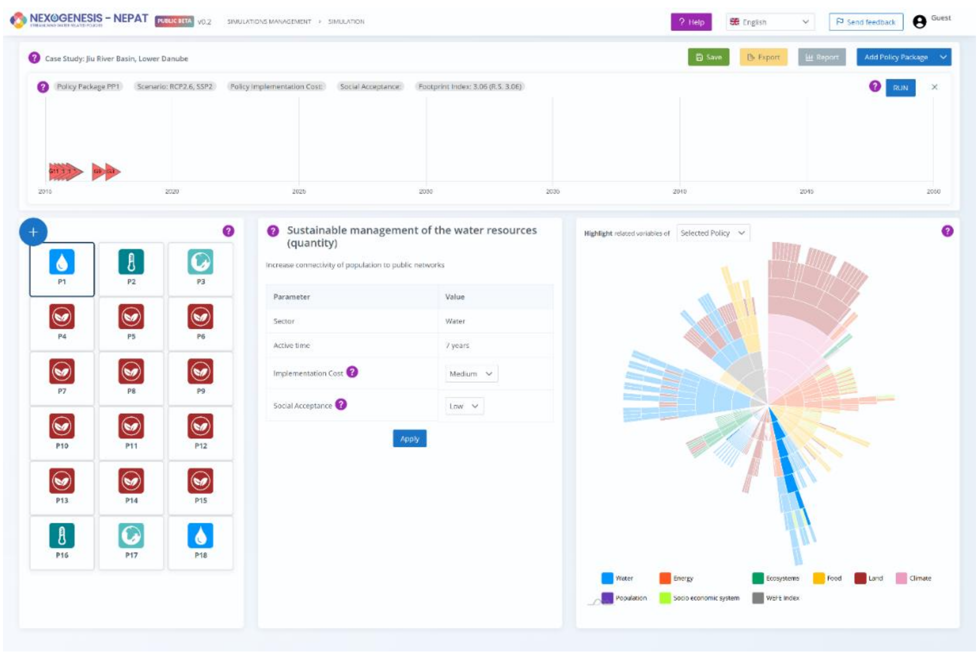
5.5. Screen Resolutions

Responsiveness was considered during the design and implementation of the User Interface, allowing the tool to be accessed from various devices, including computers, tablets, and mobile phones. As a web-based application, it adjusts to different screen sizes to provide a consistent experience. However, due to the complexity of the visualizations and data-heavy charts, interpreting certain elements on smaller screens may be challenging.

For that reason, although it is possible to be used with any of the forementioned devices, it is recommended to use a laptop or computer for optimal experience. Larger screens provide better readability and allow users to fully engage with the charts, diagrams, and interactive elements without the constraints of limited screen space. When using a mobile phone, viewing the tool in horizontal (landscape) orientation is advised, as it provides a better layout and improves the visibility of visualizations.

To illustrate this, the following figures showcase how the tool adapts across different devices, highlighting the differences in usability and visibility between desktops, tablets, and mobile screens.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881



Case Study: Jiu River Basin, Lower Danube

Policy Package PP1 | Scenario: RCP2.6, SSP2 | Policy Implementation Cost: | Social Acceptance: | Footprint Index: 3.06 (R.S. 3.06)

Sustainable management of the water resources (quantity)

Increase connectivity of population to public networks

Parameter	Value
Sector	Water
Active time	7 years
Implementation Cost	Medium
Social Acceptance	Low

Apply

Figure 40. NEPAT Policies view in computer

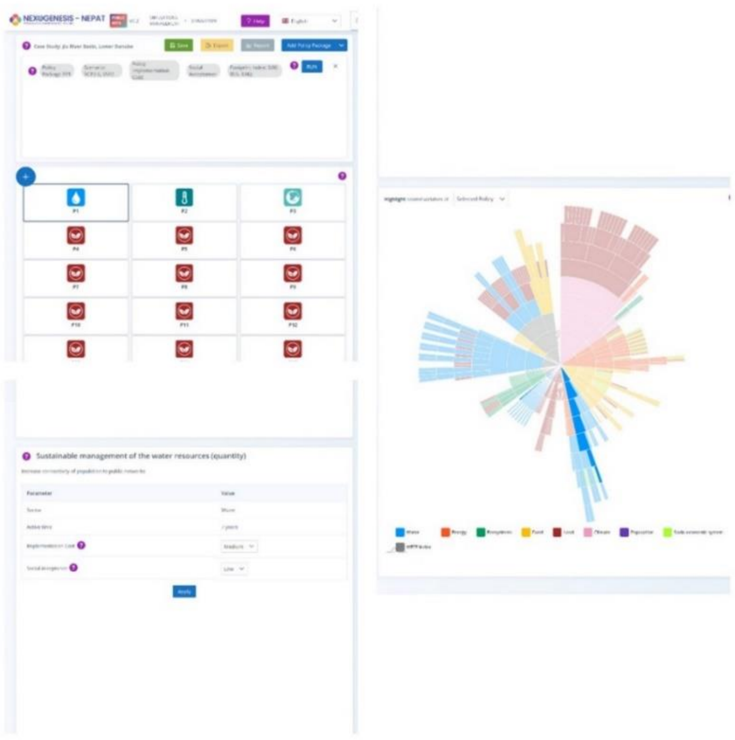
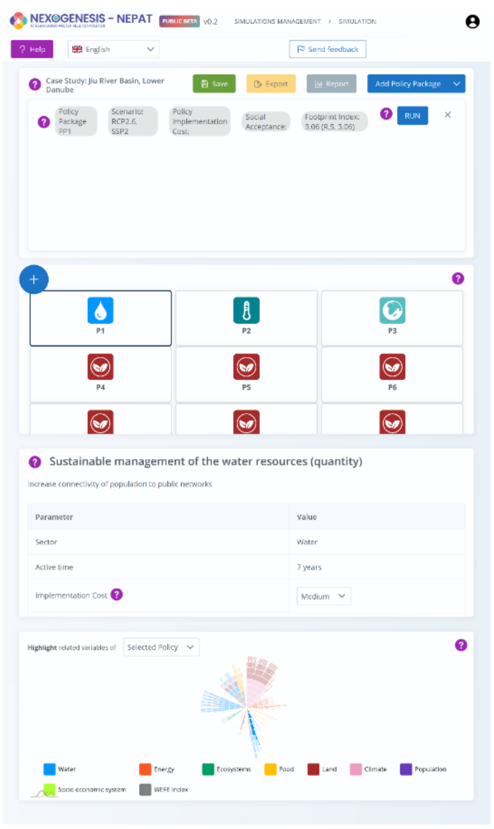


Figure 41. NEPAT Policies view in tablet



Case Study: Jiu River Basin, Lower Danube

Policy Package PP1 | Scenario: RCP2.6, SSP2 | Policy Implementation Cost: | Social Acceptance: | Footprint Index: 3.06 (R.S. 3.06)

Sustainable management of the water resources (quantity)

Increase connectivity of population to public networks

Parameter	Value
Sector	Water
Active time	7 years
Implementation Cost	Medium

Highlight related variables of: Selected Policy

Figure 42. NEPAT Policies view in mobile

5.6. Security

Security is a fundamental aspect of the development process of the NEPAT. The application is designed with multiple layers of protection to ensure that user data is safe from common web vulnerabilities. The NEPAT GUI, implemented using Angular, by default, provides automatic protection against Cross-Site Scripting (XSS) attacks by sanitizing inputs. This feature ensures that any potentially dangerous content submitted by users is filtered, preventing malicious scripts from being injected into the application’s Document Object Model (DOM). As a result, the website is less susceptible to unauthorized script execution, which could otherwise lead to data breaches or unexpected behaviors.

For secure authentication, the application utilizes modern protocols such as OAuth 2.0 and JSON Web Tokens (JWT). These methods ensure that user credentials are protected during authentication and that secure, encrypted tokens are used for session management. By leveraging techniques like HTTP-only cookies for storing session tokens, the website minimizes the risk of session hijacking and cross-site scripting vulnerabilities that could compromise user authentication.

In terms of session management, the application uses secure token storage and enforces automatic expiration of session tokens after a predefined time. Refresh tokens and other

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

measures are in place to provide secure, persistent sessions without compromising security. This minimizes the risk of unauthorized access while maintaining a smooth user experience.

All communication between the client and the server is encrypted through the use of HTTPS, ensuring that all data is transmitted securely. HTTPS protects against man-in-the-middle attacks by encrypting the data exchanged between the user’s browser and the server, ensuring the integrity and privacy of all information.

To further secure the application, input validation and output encoding are enforced in the Web Service API. This prevents malicious input from reaching the backend, such as in the case of SQL injection or other types of injection attacks. Only sanitized data is accepted, and any dynamic content rendered in the application is encoded, reducing the risk of malicious payloads being executed.

The website also implements a CORS (Cross-Origin Resource Sharing) middleware in FastAPI, which controls how resources can be accessed from different origins. This security measure helps prevent unauthorized cross-origin requests, reducing the risk of malicious interactions with the server. By specifying allowed origins, methods, and headers, CORS ensures that only trusted sources can communicate with the site, adding an extra layer of protection.

Finally, regular security audits and timely updates are essential to maintaining the application’s security posture. The development team consistently monitors third-party dependencies for known vulnerabilities and ensures that patches are applied promptly to protect against emerging threats. These practices, combined with the built-in security features of Angular, provide a robust framework for ensuring the safety and integrity of the application.

5.7. Resilience

Ensuring resilience is a critical component of the NEPAT’s design and implementation. The application is built to maintain availability, continuity, and proper functionality, even under adverse conditions or in the event of failures. Several key strategies have been employed to ensure the system is resilient to both expected and unexpected challenges.

One of the fundamental principles of resilience in the NEPAT is fault tolerance. The system has been designed to gracefully handle failures by isolating problematic components and preventing them from affecting the overall user experience. This is achieved through mechanisms such as error handling and retry logic at various points within the application and its services.

Load balancing is another critical feature that supports resilience. By distributing incoming traffic across multiple servers or instances, the application can effectively manage high traffic volumes, preventing any single server from becoming a bottleneck. This ensures consistent performance under varying loads, improving the user experience even during periods of high

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

demand. Section 5.3 demonstrates this with tests conducted under varying user loads, validating the effectiveness of this approach.

In the event of an unexpected failure or system outage, the NEPAT leverages disaster recovery practices to ensure quick recovery and data integrity. Backups of the database are taken regularly, and systems are designed to automatically restore from these backups in case of data loss or corruption.

Lastly, the system is designed to support continuous improvement through iterative testing and validation. Regular functionality testing and load testing ensure that the application can handle extreme conditions while maintaining a consistent and reliable service.

5.8. Scalability

The NEPAT architecture, introduced in section 4.2, has been designed with scalability in mind, enabling efficient handling of increased traffic or demand by distributing and replicating various components. This architecture can be deployed across multiple servers for horizontal scaling, but it is also flexible enough to scale within a single server by replicating the API running multiple Gunicorn workers.

The Web Service API is already optimized for scalability, currently operating with six Gunicorn workers to handle multiple concurrent requests. As demand grows, the number of workers can be increased dynamically within the same server, provided sufficient resources are available. Additionally, like the other components of the NEPAT architecture, the API can be replicated across multiple servers, allowing for even greater scalability and resilience.

Together, these modular and distributed components ensure that the NEPAT application can scale efficiently to meet increasing demands, whether it’s adding more workers to the API or replicating the database across multiple servers. This approach offers both flexibility and performance, providing a robust architecture capable of adapting to future growth.

5.9. Interoperability

The NEPAT architecture is built with interoperability as a fundamental principle, ensuring seamless integration with external systems, third-party services, and diverse technology stacks. By adhering to widely accepted standards and protocols, NEPAT facilitates efficient communication across different environments while maintaining flexibility, security, and scalability.

The Web Service API follows RESTful principles, leveraging standard HTTP methods such as GET, POST, PUT, and DELETE to enable consistent and predictable interactions. Data

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

exchange is structured using JSON, a lightweight and widely supported format, ensuring compatibility with web applications, mobile clients, and external services. This standardized approach simplifies integration, allowing different systems to interact with NEPAT efficiently.

Furthermore, NEPAT is designed to support multi-platform interactions, enabling seamless communication across web and mobile applications. Its modular and decoupled architecture allows for easy adaptability, whether deployed as a monolithic system or as part of a distributed microservices ecosystem. This ensures that NEPAT can function as a standalone solution or integrate smoothly into larger, more complex infrastructures.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

6. Co-Creation Process and User Feedback

The NEPAT beta version, incorporating the WEFE policy framework, goals, targets, indicators, nexus complexity modeling, and DSS functionalities, underwent validation through co-creation workshops with CSs and SHs. Feedback from these sessions drove substantial enhancements to the platform's GUI and informed the refinement of its functionalities, including the addition, removal, and improvement of features. This iterative process facilitated the transition from the beta version to the final version of NEPAT, detailed in Section 2.

6.1. Co-Creation in NEPAT Refinements

The refinements to NEPAT can be categorized into three primary areas: enhancement to existing functionalities, development of pending functionalities, and inclusion of new functionalities.

Enhancements to existing functionalities

Features from the beta version were iteratively improved based on user feedback. These enhancements addressed both functional aspects and visualization elements (GUI), ensuring the platform better aligned with user expectations.

Development of pending functionalities

Functionalities planned but not included in the beta version were further developed through a comprehensive process. This involved creating detailed mock-ups, conducting workshops to finalize their features and visualization requirements, and ultimately integrating these functionalities into the platform.

Inclusion of new functionalities

Additional functionalities, not initially planned, were incorporated in response to the evolving needs and priorities of CSs and SHs. These new features were designed through a co-creation process, ensuring the platform effectively adapted to user requirements.

The workflow for integrating user feedback and refining the NEPAT tool, addressing both functionality and visualization improvements, is illustrated in Figure 43. The iterative refinement process of the NEPAT tool identified several key issues through the feedback received, which were subsequently addressed with targeted solutions. Below is a summary of these issues and the corresponding improvements.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

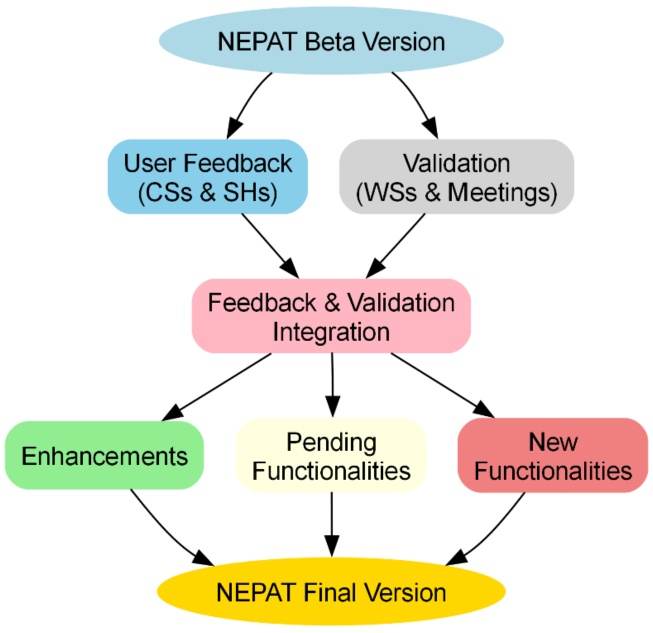


Figure 43. Workflow of user feedback and tool refinement

Case study and scenario descriptions

- **Feedback**: Case study and reference scenario descriptions were outdated and unclear.
- **Solution**: Descriptions for multiple case studies were updated, and new Representative Concentration Pathways and Shared Socioeconomic Pathways descriptions, provided by WP2, were integrated into the tool for improved clarity.

Tool intuitiveness

- **Feedback**: Users found the tool unintuitive and overwhelming due to excessive information.
- **Solution**: Advanced functionalities, such as the sunburst graph for policy impacts, uncertainty visualizations, and AI-based requests, were moved to an optional advanced view. A simplified interface was introduced for general users, retaining advanced functionalities for in-depth analysis. The advanced functionalities are described in detail in the GUI module in Section 2.

Graphics, legends, and user navigation

- **Feedback**: Graphics lacked units, legends were unclear, colors were inconsistent, and variable names were ambiguous.
- **Solution**: Enhancements implemented include adding units to graphics, improving legends, renaming variables, and introducing a variable search feature to improve user accessibility and navigation.

Simulation save feature

- **Feedback:** The save prompt appearing with every *RUN* press made the simulation save function confusing.
- **Solution:** The save prompt was removed to streamline the process. Users can save simulations by clicking the *Save* button whenever desired.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Expanding sunburst graphs

- **Feedback**: There was a suggestion to expand the sunburst graph to analyze complete policy packages and compare scenarios with reference scenarios.
- **Solution**: New sunburst graphs have been added to the *Policy Instruments* view to analyze the impacts of policy packages and enable comparisons between reference and policy scenarios.

Case study-specific customizations

- **Feedback**: Some case studies, such as Lielupe and Inkomati, required tailored functionalities to address unique needs, including dynamic policy applications and joint policy implementation. Additionally, the Nestos case study required a customized option for selecting the target year for its goals.
- **Solution**: Customizations were developed to meet these specific requirements. These enhancements ensure that the platform accommodates the distinct needs of each case study effectively.

User guidance

- **Feedback**: Users reported difficulties navigating the tool due to insufficient guidance.
- **Solution**: Sections were renamed with clearer titles, and help features were added to explain views and functionalities. A comprehensive tutorial was shared with CSs, and an online video tutorial (accessible via NEPAT Video) is available in English with multilingual subtitles.

WEFE Footprint understanding

- **Feedback**: The WEFE footprint index was difficult to interpret.
- **Solution**: In collaboration with WP3, the WEFE footprint index was redefined and scaled between -100 and 100 for better intuitiveness. Positive values indicate beneficial impacts compared to the baseline year (2015), while negative values indicate detrimental impacts.

Validated policy packages

- **Feedback**: During the co-creation work on validated policy packages, optimal policy packages identified through reinforcement learning were found to require further validation, refinement, and updates to the policy instruments, goals, and targets.
- **Solution**: Reference scenario data and modifications to SDMs, policy instruments, goals, and indicators were updated across all case studies.

Decision Support System

- **Feedback**: The DSS was unclear, and results were difficult to interpret.
- **Solution**: The DSS was restructured into two components: one for policy goals and another for footprint-relevant variables. Case studies selected their preferred integration, with Lielupe using only the policy goals component, while others incorporated both. Each DSS now recommends policy packages based on user requests, ranking them by overall objective achievement. Detailed information on individual objectives is accessible via a dropdown menu.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Language support

- **Feedback**: Limited language options restricted accessibility.
- **Solution**: The tool now supports seven languages: English, Greek, Bulgarian, Latvian, Lithuanian, Romanian, and Italian.

Uncertainty analysis

- **Feedback**: The lack of uncertainty analysis limited usefulness for technical users.
- **Solution**: Uncertainty analysis was incorporated as an advanced functionality, allowing users to specify stochastic runs and view statistical results across all result sections.

Simulation comparisons

- **Feedback**: Stakeholders highlighted the importance of simulation comparisons, particularly for analyzing policy impacts within individual case studies.
- **Solution:** A dedicated comparison view was created, enabling users to analyze policy impacts within the same case study across different reference scenarios. Cross-case comparisons were excluded since the WEFE footprint was the only shared variable, and focusing on within-case analysis better aligned with stakeholder needs.

Duplicate, Import, Export, and Report features

- **Feedback**: Duplicate, import, export and report functionalities were pending.
- **Solution**: All functionalities were successfully implemented.

The feedback for the NEPAT platform, which was used to refine the tool, is summarized in Table 37. It indicates whether each feature represents an improvement to an existing functionality, a refinement of a functionality that was previously under development, or the addition of a new functionality.

Table 37. Summary of feedback, enhancements, and new NEPAT features

Refinement to NEPAT	Improvement	Development	New
Case study and scenario descriptions	✓		
Tool intuitiveness	✓		
Graphics, legends, and user navigation	✓		
Simulation save feature	✓		
Expanding sunburst graphs			✓
Case study-specific customizations			✓
User guidance			✓
WEFE Footprint understanding	✓		
Validated policy packages	✓		
Decision Support System	✓		
Language support		✓	
Uncertainty analysis		✓	
Simulation comparisons		✓	
Duplicate, Import, Export, and Report features		✓	

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

6.2. Feedback Collection

Feedback was gathered from CSs, SHs, and potential users through internal project co-creation meetings, discussions, and external workshops with SHs. This process included insights from the fifth workshops of the Nestos, Lielupe, and Jiu case studies, with hands-on sessions held during the Lielupe and Jiu workshops.

While most feedback has already been integrated into the platform, delays in the fifth workshops for the Inkomati and Adige case studies mean their feedback has not yet been incorporated. These additional inputs will inform future updates as they become available, ensuring the platform evolves based on comprehensive user insights.

To support ongoing improvements, the NEPAT platform includes a built-in feedback mechanism. A *Send Feedback* button, located in the upper-right corner of the interface, directs users to a feedback form where they can report issues or provide suggestions (see Figure 44). All submissions are systematically reviewed and considered for inclusion in future updates, maintaining the platform's adaptability and user-centric design.

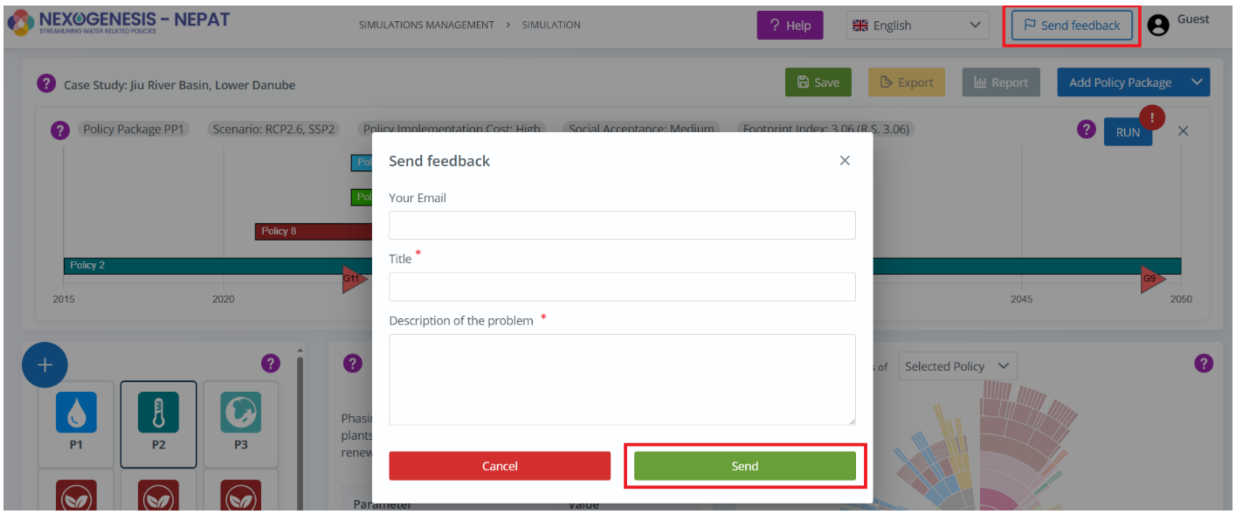


Figure 44. Built-in feedback mechanism in NEPAT platform

Hands-On Sessions

The hands-on sessions provided stakeholders, including potential future users of the NEPAT tool, with a valuable opportunity for direct, practical engagement. In earlier workshops, participants were introduced to NEPAT through mock-ups, the initial deployment of the online platform, and a video guide outlining its functionality and usability. However, the hands-on sessions marked the first instance of active, in-depth interaction with the tool. During these sessions, NEPAT's features were demonstrated, explored, and tested, allowing stakeholders to actively engage with the tool by experimenting with policy packages aimed at achieving the WEFE goals for the respective river basin.

Below, we summarize the two hands-on sessions conducted and their key contributions to NEPAT's development.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Hands-on session in Jiu

The fifth Jiu Case Study workshop was held on September 17, 2024, in Bucharest, Romania, with 18 stakeholders, including representatives from policy-making units, educational institutions, and civil society. The workshop aimed to advance the WEFE Nexus approach in Romania and encourage practical use of the NEPAT tool.

The first session emphasized the relevance of the WEFE Nexus to Romania's SDGs, providing updates on project achievements and initiatives from River Basin authorities. The NEPAT tool was introduced with a demonstration of its application to selected WEFE policies in the Jiu River Basin, showcasing its potential for informed water resource management.
The second session focused on hands-on training and gathering feedback on the NEPAT tool. Stakeholders were provided with an information package prior to the workshop to help them understand the tool's purpose and functionality.
The session began with a brief introduction and tutorial on NEPAT. Participants then engaged in a practical exercise, with its topic carefully selected based on the specific interests and concerns of the Jiu CS. The exercise focused on addressing the increasing reliance on irrigation for crop production due to erratic weather patterns and reduced rainfall. Working individually or in pairs, participants were tasked with improving water use efficiency in the Jiu River Basin. They developed strategies to enhance crop yields while balancing broader development goals by applying policy instruments within NEPAT and analyzing the results through its visualization tools.

In the final part of the exercise, participants utilized the tool's DSS to request recommendations and compared the suggested strategies with their own results. The activity employed the *WEFE_Index__Crop_per_drop* indicator to assess irrigation efficiency and optimize water use in the context of water scarcity and climate unpredictability.

The session concluded with a discussion where participants provided feedback and suggestions for improving the tool:

- Participants appreciated the tool's interactive design and its relatively easy-to-understand structure.
- A general recommendation was made to integrate additional explanations for the various visual elements to ensure accessibility and clarity for users outside the project community.

Hands-on session in Lielupe

The fifth Lielupe Case Study workshop took place on October 2, 2024, in Riga, Latvia, with the participation of 11 stakeholders. Attendees included policymakers at various levels, representatives from universities, NGOs, agriculture, and municipalities. The workshop aimed to introduce stakeholders to the NEPAT tool and engage them in hands-on exploration and testing of its functionalities to enhance their understanding of nexus interlinkages and policy-making in the Lielupe River Basin.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

The NEPAT exploration session began with a guided introduction to the tool, highlighting its key functionalities. Participants were shown how the tool could simulate the impacts of case study-related policies, monitor and evaluate these impacts across WEFE goals and indicators under various future scenarios, and provide policy recommendations to achieve these goals. Relevant terminology and features specific to the Lielupe CS were also explained.

Participants were divided into four working groups, each supported by 1–2 project consortium experts, to explore and test NEPAT's capabilities. Supplementary materials were provided to facilitate the exercise, including information on the active reference scenario, policy goals and targets for the Lielupe Case Study, and a set of policy instruments available for testing within the tool.

During the session, participants underscored the importance of cross-border cooperation between Latvia and Lithuania to improve water quality in the Lielupe River Basin. However, they also identified challenges, such as differing methodologies for water quality assessment and the preparation of river basin management plans at the national level, which hinder effective collaboration.

Stakeholders offered valuable feedback to improve the NEPAT tool's user experience:

- They emphasized the need for a user manual to provide clarity on background information, abbreviations, and available options within the tool.
- The newly introduced comparison view was well-received.
- They highlighted the need for further clarification of table indicators and legends, including more detailed descriptions of policy instruments and goals.
- They suggested incorporating evaluations of policy implementation costs and social acceptance to enhance the tool's support for informed decision-making.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

7. Conclusions

The current document details the successful development, deployment, and validation of NEPAT as an advanced decision-support system for policy simulation and analysis. The NEPAT platform integrates a comprehensive simulation policy framework, a robust core service, and a web service API to ensure seamless interaction and data management. The interactive graphical user interface has been designed to enhance user experience, incorporating advanced functionalities, multilingual support, and a user-friendly help system. Additionally, the self-learning engine and AI-driven recommendation algorithms—including deterministic, stochastic, and dynamic policies recommendation engines—have been implemented to provide data-driven insights for decision-making.

Beyond its core functionalities, NEPAT ensures efficient system deployment, adhering to high standards of scalability, security, resilience, and interoperability. Extensive testing and validation have confirmed the reliability and performance of the platform, covering API functionality, load capacity, browser compatibility, and system robustness across different environments. Furthermore, the co-creation process has played a key role in refining the platform, integrating direct stakeholder feedback to ensure the tool meets real-world policy analysis and decision-making needs.

NEXOGENESIS STREAMLINING WATER RELATED POLICIES

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Future Work/Next Steps

As reported in this document, all NEPAT functionalities have been successfully implemented, with the final version now available. The next steps focus on refining and enhancing the tool based on stakeholder feedback from upcoming workshops, integrating updates to SDMs, policies, and goals as needed, and further improving functionalities such as language translations and help features to enhance usability and clarity.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

Annexes

Annex I: Web Server API Documentation

Authentication endpoints

Location	/auth/signup
Methods	POST
Params	-
Headers	-

Body	{ "username": "string",
	  "email": "user@example.com",
	  "password": "string"}

Response	Message

	[200, OK]
	{
	  "message": "User successfully registered!"
	}
	[409, Conflict Error]
	{“detail”: "User with supplied email already exists"}
	[422, Validation Error]
	{
	  "detail": [
	    {
	      "loc": [
	        "string",
	        0
	      ],
	      "msg": "string",
	      "type": "string"
	    }
	  ]
	}

	Cookies
	-
Description	This endpoint manages the registration of new users. It requires the username, email, and password as mandatory parameters. Upon successful registration, an email is sent to the user for validation.

Location	/auth/verify-email/{verification_code}		
Methods	POST		
Params	Name	Type	Description
	verification_code	String	Code generated for the user to verify its email authenticity.
Headers	-		
Body	-		
Response	Message		
	[200, OK] Redirection to NEPAT login page. [403, Forbidden Error] {“detail”: “Invalid verification code or account already verified”} {“detail”: “Verification code expired”} [422, Validation Error] { "detail": [		

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

	{ "loc": ["string", 0], "msg": "string", "type": "string" }] }		
	Cookies		
	-		
Description	This endpoint validates the authenticity of a user's email. If the provided code is correct, the user is redirected to the login page, allowing the verified user to access the tool.		

Location	/auth/resend-verification-email /{verification_code}		
Methods	POST		
Params	Name	Type	Description
	verification_code	String	Code generated for the user to verify its email authenticity.
Headers	-		
Body	-		
Response	Message		
	[200, OK] {"message": "Re sent verification email"} [422, Validation Error] { "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }		
	Cookies		
	-		
Description	This endpoint generates and sends a new verification email to the user, including a valid verification code. It is designed for cases where the user's previous verification code has expired or is no longer valid.		

Location	/auth/login
Methods	POST
Params	-
Headers	-
Body	{ "email": "user@example.com", "password": "string"}
Response	Message
	[200, OK] Login successful [401, Unauthorized error] {“detail”: “Invalid details passed.} [403, Forbidden Error]

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

	{“detail”: “User not verified. Please, verify your account with the email received.”}
	[404, Not found Error]
	{“detail”: “User with email does not exist”}
	[422, Validation Error]
	{ "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }
	Cookies
	[200, OK]
	{ "username”:username, "access_token":access_token, "token_type":"Bearer" }
Description	This endpoint handles the login process for a registered user, verifying the provided email and password. Upon successful authentication, it returns a JWT token that grants access to functionalities requiring user authentication. If the credentials are invalid, an appropriate error message is returned to inform the user.

Location	/auth/password-recovery
Methods	POST
Params	-
Headers	-
Body	{ "email": "string", "new_password": "string" }
Response	Message
	[200, OK]
	{"message": "Password recovery email sent"}
	[403, Forbidden error]
	{“detail”: “User not verified. Please, verify your account with the email received.”}
	[404, Not found error]
	{“detail”: “The user with this email does not exist in the system.”}
	[422, Validation Error]
	{ "detail": [{ "loc": ["string", 0], "msg": "string", "type": "string" }] }
	Cookies
	-

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Description	This endpoint allows a user to reset their password in case of loss. To confirm the user’s identity, an email is sent to validate the operation before changing the password.

Location	/auth/reset-password		
Methods	POST		
Params	Name	Type	Description
	Token	String	Password renewal token generated in /auth/password-recovery
	New_password	String	Encrypted new password.
Headers	-		
Body			
Response	Message		

	[200, OK]
	{"message": "Password recovery email sent"}
	[404, Not found error]
	{“detail”: “Invalid token.”}
	{“detail”: “The user with this username does not exist in the system.”}
	[422, Validation Error]
	{
	  "detail": [
	    {
	      "loc": [
	        "string",
	        0
	      ],
	      "msg": "string",
	      "type": "string"
	    }
	  ]
	}		

	Cookies		
	-		
Description	This endpoint resets the user’s password after verifying the password recovery token is correct.		

Location	/auth/google-login
Methods	POST
Params	-
Headers	-
Body	-
Response	Message
	[200, OK] Login successful [400, Bad request error] {“detail”: “Unable to validate social login.”}
	Cookies

	[200, OK]
	{
	    "username”:username,
	    "access_token":access_token,
	    "token_type":"Bearer"
	}

Description	This endpoint handles the login process for user wishing to log in with a Google account. Upon successful authentication, it returns a JWT token that grants access to functionalities requiring user authentication.

Location	/auth/guest-login
Methods	POST

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Params	-
Headers	-
Body	-
Response	Message
	[200, OK] Login successful
	Cookies

	[200, OK]
	{
	"username”:username,
	"access_token":access_token,
	"token_type":"Bearer"
	}

Description	This endpoint handles the login process for user wishing to log as a guest without providing any credentials. It registers a new user and provides its JWT token to interact with the endpoints that require authentication. All data generated with a guest user will be lost upon the token’s expiration.

Location	/profile
Methods	GET
Params	-
Headers	Authorization: Bearer <token>
Body	-
Response	Message
	[200, OK] {"username": user.username, "email": user.email} [401, Unauthorized error] {“detail”: “Could not validate credentials.”} [403, Forbidden Error] {“detail”: “Sign in for access.”}
	Cookies
	-
Description	This endpoint returns the current authenticated user’s username and email, used to be shown in the profile section of the tool.

Feedback reports endpoint

Location	/bug-reports
Methods	POST
Params	-
Headers	Authorization: Bearer <token>

Body	report= {
	"email": "user@example.com",
	"title": "string",
	"report": "string"
	}

Response	Message
	[200, OK] {"message": "Bug report created successfully!"} [401, Unauthorized error] {“detail”: “Could not validate credentials.”} [403, Forbidden Error] {“detail”: “Sign in for access.”}
	Cookies
	-

Description	This endpoint allows users to report issues or provide feedback about the tool. The provided information is used to send an email to the development team. Only the title and report fields are mandatory.

Case study endpoints

Location	/case-studies		
Methods	GET		
Params	Name	Type	Description
	Detail	Bool	Whether to get a detailed response including also the scenarios and goals for each case study.
Headers	Authorization: Bearer <token> Accept-language: [en, el, bg, lv, lt, ro, it]		
Body	-		
Response	Message		

	[200, OK]
	Detail=true
	[
	{
	"id": 0,
	"name": "string",
	"site": "string",
	"short_description": "string",
	"long_description": "string",
	"scenarios": [],
	"policy_goals": [
	{
	"label": "string",
	"description": "string",
	"indicator": "string",
	"year": 0,
	"target": "string",
	"goal_id": 0
	}
	]
	}
	]
	Detail=false
	[
	{
	"id": 0,
	"name": "string",
	"site": "string",
	"short_description": "string",
	"long_description": "string",
	}
	]
	[401, Unauthorized error]
	{“detail”: “Could not validate credentials.”}
	[403, Forbidden Error]
	{“detail”: “Sign in for access.”}		

	Cookies		
	-		
Description	This endpoint returns the list of case studies, optionally with each case study’s policy goals and available scenarios.		

Location	/case-studies/{id}/scenarios/{scenario_id}/indicators
Methods	GET

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Params	Name	Type	Description
	Id	Str	Case study id
	Scenario_id	Str	Scenario id
Headers	Authorization: Bearer <token> Accept-language: [en, el, bg, lv, lt, ro, it]		
Body	-		
Response	Message		
	[200, OK] [“var_name_1”, “var_name_2”, …] [401, Unauthorized error] {“detail”: “Could not validate credentials.”} [403, Forbidden Error] {“detail”: “Sign in for access.”}		
	Cookies		
	-		
Description	This endpoint returns the list of nexus variables there are for a specific case study and scenario fixed by the parameters {id} and {scenario_id}.		

Simulation endpoints

Location	/simulations-wrappers
Methods	GET
Params	-
Headers	Authorization: Bearer <token> Accept-language: [en, el, bg, lv, lt, ro, it]
Body	-
Response	Message

	[200, OK]
	[
	  {
	    "name": "string",
	    "description": "string",
	    "creation_date": "2025-01-21T08:55:25.946Z",
	    "modification_date": "2025-01-21T08:55:25.946Z",
	    "case_studies": "string",
	    "scenarios": "string",
	    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
	    "simulations": [
	      {
	        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
	        "case_study": {
	          "name": "string"
	        },
	        "scenario": {
	          "id": 0,
	          "label": "string"
	        },
	        "policy_packages": [
	          {
	            "policy_links": [
	              {
	                "policy": {
	                  "id": 0,
	                  "policy_id": 0,
	                  "short_name": "string",
	                  "name": "string",
	                  "description": "string",

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

```
                  "sector": "string",
                  "active_time": 0,
                  "permanent": true,
                  "multiple": true,
                  "implementation_cost": "string",
                  "social_acceptance": "string",
                  "sdm_variables": [
                    "string"
                  ],
                  "policy_field": {
                    "name": "string",
                    "field_unit": "string",
                    "field_type": "string",
                    "field_from": "string",
                    "field_to": "string",
                    "step_size": 0,
                    "min_value": 0,
                    "max_value": 0,
                    "id": 0
                  },
                  "fixed_year": 0,
                  "incompatible_policies_ids": [
                    0
                  ],
                  "incompatible_policies_db_ids": [
                    0
                  ],
                  "depends_on": 0,
                  "requires": [
                    0
                  ]
                },
                "year": 0,
                "value": 0
              }
            ]
          }
        ]
      }
    ]
  }
]
```

[401, Unauthorized error]
{“detail”: “Could not validate credentials.”}
[403, Forbidden Error]
{“detail”: “Sign in for access.”}

	Cookies		
	-		
Description	This endpoint enables users to retrieve all their persisted simulations. It is specifically designed for use within the management view. The language parameter is required only for GET requests.		
Location	/simulations-wrappers		
Methods	POST		
Params	Name	Type	Description
	Persist	Bool	Whether to save the simulation created.
Headers	Authorization: Bearer <token> Accept-language: [en, el, bg, lv, lt, ro, it]		
Body	{		

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

```
  "name": "string",
  "description": "string",
  "simulations": [
    {
      "case_study_id": 0,
      "scenario_id": 0,
      "policy_packages": [
        [
          {
            "policy_id": 0,
            "year": 0,
            "value": 0
          }
        ]
      ],
      "custom_goals": [
        {
          "label": "string",
          "description": "string",
          "indicator": "string",
          "year": 0,
          "target": "string"
        }
      ],
      "goals_year": 0
    }
  ]
}
```

Response

Message

```
[200, OK]
[
  {
    "name": "string",
    "description": "string",
    "creation_date": "2025-01-21T08:55:25.946Z",
    "modification_date": "2025-01-21T08:55:25.946Z",
    "case_studies": "string",
    "scenarios": "string",
    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "simulations": [
      {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "case_study": {
          "name": "string"
        },
        "scenario": {
          "id": 0,
          "label": "string"
        },
        "policy_packages": [
          {
            "policy_links": [
              {
                "policy": {
                  "id": 0,
                  "policy_id": 0,
                  "short_name": "string",
                  "name": "string",
                  "description": "string",
                  "sector": "string",
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

```
                  "active_time": 0,
                  "permanent": true,
                  "multiple": true,
                  "implementation_cost": "string",
                  "social_acceptance": "string",
                  "sdm_variables": [
                    "string"
                  ],
                  "policy_field": {
                    "name": "string",
                    "field_unit": "string",
                    "field_type": "string",
                    "field_from": "string",
                    "field_to": "string",
                    "step_size": 0,
                    "min_value": 0,
                    "max_value": 0,
                    "id": 0
                  },
                  "fixed_year": 0,
                  "incompatible_policies_ids": [
                    0
                  ],
                  "incompatible_policies_db_ids": [
                    0
                  ],
                  "depends_on": 0,
                  "requires": [
                    0
                  ]
                },
                "year": 0,
                "value": 0
              }
            ]
          }
        ]
      }
    ]
  }
]
[401, Unauthorized error]
{“detail”: “Could not validate credentials.”}
[403, Forbidden Error]
{“detail”: “Sign in for access.”}
[404, Not found Error]
{“detail”: “Provided a nonexistent id!”}
[422, Validation Error]
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

	Cookies
	-
Description	This endpoint allows users to create a new simulation wrapper by specifying its name, description, and a list of simulations to include. If persist is set to false, the simulation wrapper is not saved to the database until explicitly persisted using the PUT method.

Location	/simulations-wrappers/{id}		
Methods	GET, DELETE		
Params	Name	Type	Description
	Id	uuid	Id of the simulation wrapper
	Baseline	Bool	Whether to include in the run results the reference scenario results.
Headers	Authorization: Bearer <token> Accept-language: [en, el, bg, lv, lt, ro, it]		
Body	-		
Response	Message		
	[200, OK] GET result		

```
[
  {
    "name": "string",
    "description": "string",
    "creation_date": "2025-01-21T08:55:25.946Z",
    "modification_date": "2025-01-21T08:55:25.946Z",
    "case_studies": "string",
    "scenarios": "string",
    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "simulations": [
      {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "case_study": {
          "name": "string"
        },
        "scenario": {
          "id": 0,
          "label": "string"
        },
        "policy_packages": [
          {
            "policy_links": [
              {
                "policy": {
                  "id": 0,
                  "policy_id": 0,
                  "short_name": "string",
                  "name": "string",
                  "description": "string",
                  "sector": "string",
                  "active_time": 0,
                  "permanent": true,
                  "multiple": true,
                  "implementation_cost": "string",
                  "social_acceptance": "string",
                  "sdm_variables": [
                    "string"
                  ],
                  "policy_field": {
                    "name": "string",
                    "field_unit": "string",
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

```
                    "field_type": "string",
                    "field_from": "string",
                    "field_to": "string",
                    "step_size": 0,
                    "min_value": 0,
                    "max_value": 0,
                    "id": 0
                  },
                  "fixed_year": 0,
                  "incompatible_policies_ids": [
                    0
                  ],
                  "incompatible_policies_db_ids": [
                    0
                  ],
                  "depends_on": 0,
                  "requires": [
                    0
                  ]
                },
                "year": 0,
                "value": 0
              }
            ]
          }
        ]
      }
    ]
  }
]
DELETE result
{"message": "Simulations wrapper deleted successfully"}
[401, Unauthorized error]
{“detail”: “Could not validate credentials.”}
[403, Forbidden Error]
{“detail”: “Sign in for access.”}
{“detail”: “Must be the owner of the Simulations Wrapper!”}
[404, Not found Error]
{“detail”: “Simulations wrapper with specified id does not exist.”}
[422, Validation Error]
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
      "type": "string"
    }
  ]
}
```

	Cookies
	-
Description	For persisted simulation wrappers, the GET method allows retrieval of a specific wrapper using its ID. The baseline parameter can be used to specify whether to include reference scenario results along with the policy package results. Additionally, a specific simulation wrapper can be deleted by providing its ID.

Location	/simulations-wrappers/{id}

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Methods	PUT		
Params	Name	Type	Description
	Id	uuid	Id of the simulation wrapper
Headers	Authorization: Bearer <token>		

Body	{
	"name": "string",
	"description": "string",
	"simulations_policy_packages": {
	"additionalProp1": [
	[
	{
	"policy_id": 0,
	"year": 0,
	"value": 0
	}
	]
	],
	…
	}
	}		

Response	Message		

	[200, OK]
	{"message": "Simulation wrapper updated successfully!"}
	[401, Unauthorized error]
	{“detail”: “Could not validate credentials.”}
	[403, Forbidden Error]
	{“detail”: “Sign in for access.”}
	{“detail”: “Must be the owner of the Simulations Wrapper!”}
	{“detail”: “Must be the owner of the Simulations!”}
	{“detail”: “Simulation belong to another Simulation wrapper!”}
	[404, Not found Error]
	{“detail”: “Simulations wrapper with specified id does not exist.”}
	{“detail”: “Simulation with specified id does not exist.”}
	[422, Validation Error]
	{
	"detail": [
	{
	"loc": [
	"string",
	0
	],
	"msg": "string",
	"type": "string"
	}
	]
	}		

	Cookies		
	-		
Description	This endpoint allows updating the name, description, or simulation list of an existing simulation wrapper. If a new simulation list is provided, the old list is modified accordingly by performing the necessary create or delete operations. All updates are immediately persisted in the database.		

Location	/simulations		
Methods	POST		
Params	Name	Type	Description
	Persist	Bool	Whether to save the simulation created.
Headers	Authorization: Bearer <token>		

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

	Accept-language: [en, el, bg, lv, lt, ro, it]

Body	{
	  "case_study_id": 0,
	  "scenario_id": 0,
	  "policy_packages": [
	    [
	      {
	        "policy_id": 0,
	        "year": 0,
	        "value": 0
	      }
	    ]
	  ],
	  "custom_goals": [
	    {
	      "label": "string",
	      "description": "string",
	      "indicator": "string",
	      "year": 0,
	      "target": "string"
	    }
	  ],
	  "goals_year": 0
	}

Response	Message

	[200, OK]
	[
	{
	  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
	  "case_study": {
	    "id": 0,
	    "name": "string",
	    "baseline_footprint": 0,
	    "policies": [
	      {
	        "id": 0,
	        "policy_id": 0,
	        "short_name": "string",
	        "name": "string",
	        "description": "string",
	        "sector": "string",
	        "active_time": 0,
	        "permanent": true,
	        "multiple": true,
	        "implementation_cost": "string",
	        "social_acceptance": "string",
	        "sdm_variables": [
	          "string"
	        ],
	        "policy_field": {
	          "name": "string",
	          "field_unit": "string",
	          "field_type": "string",
	          "field_from": "string",
	          "field_to": "string",
	          "step_size": 0,
	          "min_value": 0,
	          "max_value": 0,
	          "id": 0
	        },

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

```
        "fixed_year": 0,
        "incompatible_policies_ids": [
          0
        ],
        "incompatible_policies_db_ids": [
          0
        ],
        "depends_on": 0,
        "requires": [
          0
        ]
      }
    ],
    "nexus_indicators": [
      {
        "label": "string",
        "description": "string",
        "sdm_variable": "string",
        "id": 0
      }
    ],
    "table_indicators": {}
  },
  "scenario": {
    "id": 0,
    "label": "string"
  },
  "policy_goals": [
    {
      "label": "string",
      "description": "string",
      "indicator": "string",
      "year": 0,
      "target": "string",
      "goal_id": 0,
      "id": 0,
      "indicator_var": 0,
      "decrease": true,
      "init_month": 0,
      "goal_targets": [
        0
      ]
    }
  ],
  "goals_year": 0,
  "policy_packages": [
    {
      "policy_links": [
        {
          "policy": {
            "id": 0,
            "policy_id": 0,
            "short_name": "string",
            "name": "string",
            "description": "string",
            "sector": "string",
            "active_time": 0,
            "permanent": true,
            "multiple": true,
            "implementation_cost": "string",
```

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101003881

```
            "social_acceptance": "string",
            "sdm_variables": [
              "string"
            ],
            "policy_field": {
              "name": "string",
              "field_unit": "string",
              "field_type": "string",
              "field_from": "string",
              "field_to": "string",
              "step_size": 0,
              "min_value": 0,
              "max_value": 0,
              "id": 0
            },
            "fixed_year": 0,
            "incompatible_policies_ids": [
              0
            ],
            "incompatible_policies_db_ids": [
              0
            ],
            "depends_on": 0,
            "requires": [
              0
            ]
          },
          "year": 0,
          "value": 0
        }
      ]
    }
  ],
  "footprint_tree": {},
  "footprint_values": {},
  "footprint_baseline_values": {},
  "footprint_weights": {},
  "footprint": 0,
  "dss_footprint_vars": [
    "string"
  ],
  "variable_tree": {},
  "variable_indexes": {},
  "var_tree_values": {}
}
[401, Unauthorized error]
{“detail”: “Could not validate credentials.”}
[403, Forbidden Error]
{“detail”: “Sign in for access.”}
[404, Not found Error]
{“detail”: “Provided a nonexistent id!”}
[422, Validation Error]
{
  "detail": [
    {
      "loc": [
        "string",
        0
      ],
      "msg": "string",
```

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

	"type": "string" }] }
	Cookies
	-
Description	This endpoint facilitates the creation of a new simulation. The minimum required parameters are the case study ID, scenario ID, and policy_packages. Optional parameters, such as custom goals or the target year for filtering specific case study goals, can also be specified to refine the simulation setup.

Location	/simulations/{id}/run		
Methods	POST		
Params	Name	Type	Description
	Id	uuid	Id of the simulation.
	Baseline	Bool	Whether to include in the run results the reference scenario results.
	Number of runs	Int	Number of times to run the policy package. When >1 is runs stochastic SDM executions.
	Random seed	Int	Seed for stochastic runs.
Headers	Authorization: Bearer <token> Accept-language: [en, el, bg, lv, lt, ro, it]		
Body	[{ "policy_id": 0, "year": 0, "value": 0 }]		
Response	Message		

	[200, OK]
	{
	"var_tree_values": {},
	"footprint_values": {},
	"footprint": 0
	}
	[401, Unauthorized error]
	{“detail”: “Could not validate credentials.”}
	[403, Forbidden Error]
	{“detail”: “Sign in for access.”}
	{“detail”: “Must be the owner of the Simulation!”}
	[404, Not found Error]
	{“detail”: “Simulation with specified id does not exist!”}
	[422, Validation Error]
	{
	"detail": [
	{
	"loc": [
	"string",
	0
	],
	"msg": "string",
	"type": "string"
	}
	]
	}
	{“detail”: "Incompatible policies applied at the same time"}
	{“detail”: "Year of application must be between 2015 and 2050"}		

	Cookies		
	-		

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Description	This endpoint is used to simulate the results of executing a specific policy package. It provides multiple parameters to customize the type of run and the desired results: • **baseline**: Determines whether the reference scenario results should be included. ○ **Run Type:** ○ **Deterministic**: Set number of runs to 1. ○ **Stochastic**: Set number of runs to a value greater than 1. Optionally, specify a random seed using the random seed parameter. ○ **Variable Tree Values**: In case studies with a large number of SDM variables, var_tree_values are not returned directly. These must be retrieved individually using the endpoint simulations/{id}/var-tree-values/{var_id}.

Location	/simulations/{id}/var-tree-values/{var_id}		
Methods	POST		
Params	Name	Type	Description
	Id	uuid	Id of the simulation.
	var_id	Int	Id of the variable.
Headers	Authorization: Bearer <token>		
Body	-		
Response	Message		
	[200, OK] {…} [401, Unauthorized error] {“detail”: “Could not validate credentials.”} [403, Forbidden Error] {“detail”: “Sign in for access.”}		
	Cookies		
	-		
Description	Endpoint to retrieve the run results computed using the RUN endpoint for a specific variable. Used for case studies where the number of variables is too large to send simultaneously.		

Location	/simulations/{id}/advice		
Methods	POST		
Params	Name	Type	Description
	Id	uuid	Id of the simulation.
	Only footprint	Bool	Set to get recommendations on footprint indicators optimization.
Headers	Authorization: Bearer <token>		

Body	{
	  "policy_package": [
	    {
	      "policy_id": 0,
	      "year": 0,
	      "value": 0
	    }
	  ],
	  "num_policies_limit": 0,
	  "policy_goals_weights": [
	    0
	  ],
	  "desired_policies_sector": [
	    "string"
	  ]
	}		

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881

Response	Message

	[200, OK]
	[
	 {
	   “policies”: [],
	   “achievement_lvl”: 0,
	   “goals_completion”: [],
	   “tradeoffs_and_sinergies”: {}
	]
	[401, Unauthorized error]
	{“detail”: “Could not validate credentials.”}
	[403, Forbidden Error]
	{“detail”: “Sign in for access.”}
	{“detail”: “Must be the owner of the Simulation!”}
	[404, Not found Error]
	{“detail”: “Simulation with specified id does not exist!”}
	[422, Validation Error]
	{
	  "detail": [
	    {
	      "loc": [
	        "string",
	        0
	      ],
	      "msg": "string",
	      "type": "string"
	    }
	  ]
	}

	Cookies
	-
Description	This endpoint retrieves recommendations from the Decision Support System (DSS). Users can customize the recommendations using the following parameters: • **only footprint**: Determines whether recommendations focus on policy goals or footprint indicators. • **policy_goals_weights**: Specifies the importance of achieving each policy goal, guiding the recommendation process. • **Policy Package Restrictions**: ○ **num_policies_limit**: Limits the number of recommended policies. ○ **desired_policies_sector:** Specifies the sectors that recommended policies should belong to.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 101003881