D2.8 HiDALGO Dashboard and Services

Date: April 30, 2025

Document Identification			
Status	Final	Due Date	30/04/2025
Version	1.0	Submission Date	30/04/2025

Related WP	WP2	Document Reference	D2.8
Related Deliverable(s)	D2.1, D2.4, D2.7	Dissemination Level (*)	PU
Lead Participant	SZE	Lead Author	László Környei (SZE)
Contributors	ATOS, PSNC, SZE, USTUTT	Reviewers	Philippe Pincon (UNISTRA)
			Aristomenis Theodoridis (ICCS)

Keywords:
Operational Environment, Infrastructure Provisioning, Dashboard Architecture, Dashboard components, component interfaces

Document Information

List of Contributors	
Name	Partner
Christophe Prud'homme	UNISTRA
Jesus Gorroñogoitia Cruz	ATOS
Luis Torres	MTG
David Caballero	MTG
Sameer Haroon	USTUTT
László Környei	SZE

Document History			
Version	Date	Change editors	Changes
0.1	17/02/2025	SZE	Table of Contents
0.2	24/02/2025	Marcin Lawenda (PSNC), Rahil Doshi (FAU)	ToC, timeline and responsibilities approved
0.3	01/04/2025	László Környei	First contributions added
0.5	24/04/2025	László Környei	Ready for first review
0.6	28/04/2025	László Környei	Improvements after first review
0.9	29/04/2025	László Környei	Version ready for quality check
0.95	29/04/2025	Rahil Doshi	Quality assurance check
1.0	30/04/2025	Marcin Lawenda	Final check and improvements

Quality Control		
Role	Who (Partner short name)	Approval Date
Deliverable leader	László Környei (SZE)	29/04/2025
Quality manager	Rahil Doshi (FAU)	30/04/2025
Project Coordinator	Marcin Lawenda (PSNC)	30/04/2025

Table of Contents

List of Tables

List of Figures

List of Acronyms

Abbreviation / acronym	Description
API	Application Programming Interface
CI/CD	Continuous Integration and Continuous Deployment
DMS	Data Management System
Dx.y	Deliverable number y belonging to WP x

EC	European Commission
ECMWF	European Centre for Medium-Range Weather Forecasts
MathSO	Workflow orchestrator developed by SZE
PKCE	Proof Key for Code Exchange
QCG	Workflow orchestrator developed by PSNC
SPA	Single Page Application
SSO	Single Sign On
WFO	Workflow Orchestrator (MathSO, QCG)
WP	Work Package

Executive Summary

Deliverable D2.8 builds directly on the foundation laid by D2.7, extending the HiDALGO2 Dashboard from a proof-of-concept into a fully-fledged platform that unifies access to most consortium services. Grounded in the requirements analysis of D2.2 and aligned with data-management strategies in D4.1, this version introduces refined layouts, a responsive tile and list view, and an immersive details panel—all wrapped in the HiDALGO2 brand palette and typography. The dashboard's top ribbon, user-details area, filter pane, and central service grid have been optimized through iterative feedback, ensuring that icons, fonts, and interactive elements guide users to the tools they need with minimal friction.

Underneath its polished interface, the dashboard is powered by a Dockerized Node/Nginx deployment in the PSNC Cloud and secured end-to-end by Keycloak single-sign-on with silent-SSO and PKCE support. Sixteen core services—from workflow orchestrators (MathSO, QCG) and data-management portals (CKAN, JupyterHub) to collaboration tools (GitLab, Askbot) and upcoming HDFS/Spark interfaces—now most integrate seamlessly, with each connection level clearly documented. An upcoming four-tier authentication model (unauthenticated, external_user, internal_user, admin) will govern access, while administrators continue to manage user roles and organizations manually until automated onboarding is introduced in a future release.

Feedback from thirteen internal stakeholders demonstrates high satisfaction across design, layout, functionality, and performance, with average ratings around 4 out of 5. Participants praised the dashboard's speed and usability while identifying targeted areas for further improvement—most notably the detail-panel UX, iconography, and service taxonomy. All critical suggestions have already been addressed in this release, and remaining enhancements—such as deeper coupling of modelling workflows—are slated for upcoming milestones. D2.8 thus positions the HiDALGO2 Dashboard as an intuitive, secure, and extensible hub that will evolve alongside consortium needs and technical advancements.

1 Introduction

1.1 Purpose of the document

The deliverable “D2.8 HiDALGO2 Dashboard and Services (M28)” is a continuation of deliverable D2.7 of the same name. It elaborates the dashboard in terms of functionality and details how already delivered and existing backend services will be integrated with the HiDALGO2 platform including orchestration, data management, visualization and complimentary services. While the dashboard will cover expectations of both internal and external stakeholders, in this document internal stakeholders are considered.

1.2 Relations to other project work

This document – continuing D2.7 – is based on the rationale of integrating services and strategies from deliverables within HiDALGO2, in addition to updated requirements gathered in “D2.2 Requirements Analysis and Scenario Definitions” (M18). Additionally, it aligns with “D4.1 Data Management and Coupling Technologies” (M11) and provides a platform for both “D4.7 Visualization for Global Challenges” and “D2.5 Infrastructure Provisioning, Workflow Orchestration and Component Integration”.

1.3 Structure of the document

The document is structured into five main chapters. After introduction in Chapter 1,

Chapter 2 details the first release of the dashboard focusing on layout, design, functionalities and deployment.

Chapter 3 details the integration of HiDALGO2 services, starting with currently integrated ones, detailing single sign-on integration and addressing further services to be integrated. It also discusses user registration and management.

Chapter 4 details the ongoing process of integrating external services provided by partners and pilots, including data management, external frameworks and visualization systems.

Chapter 5 reports on the questionnaire conducted with internal stakeholders as described in D2.7. Questionnaire design and results are discussed, including results analysis.

Finally, Chapter 6 summarizes and concludes this deliverable while giving an outlook about the future objectives and challenges.

2 Dashboard first release

The first instalment of the HiDALGO2 dashboard was deployed in M23 following the descriptions and requirements described in D2.7. The functionalities, layout, and design were improved in many turns over time following steady discussion with internal stakeholders. In this chapter the current layout, design, functionality and deployment method is described in detail.

The main functionality of the dashboard is to gather HiDALGO2 services, providing a starting point for internal and external users. Layout and design follow the principles of presenting webpage structure following concepts of quick access and usability. In the functionality section specific functionalities of the dashboard are presented and discussed. The last section of this chapter describes dashboard deployment for the current HiDALGO2 ecosystem.

2.1 Layout and design

The layout and design of the dashboard focuses on pleasant user experience, quick overview of services and seamless transition to additional HiDALGO2 service providers. Design-wise, the HiDALGO2 colours and typography work together to guide the users' eyes, establish hierarchy, and communicate brand personality. The proper contrast supports readability and accessibility in cohesion with the existing HiDALGO2 brand identity. While background colours and neutral tones ensure content remains legible, accent colours are used sparingly to draw attention to call-to-action or important information. As far as fonts are concerned, the priority is legibility, by selecting typefaces with clear shapes and establishing a typographic hierarchy separating e.g. headings and body text. Font sizes, line-height, and weight variations are calibrated so that the page structure is instantly graspable on both desktop and mobile devices.

In the current layout, the screen is divided into four main areas, as depicted in Figure 1. There is the top ribbon section (1) and the user detail section (2). The main window consists of the filters bar (3) on the left-hand side, and the tile area in the centre (4). Following is a short description of the four sections:

Top ribbon (1)

The area consists of the HiDALGO2 logo, followed by a few central links. The underlined dashboard link indicates to the user that he is on the dashboard. Also, this link closes the detail page. Furthermore, it takes the user back to the central screen, when on other pages of the dashboard. The Project Website link takes the user to the central HiDALGO2 website. The last link is a "Contact Us" form, where users can contact HiDALGO2 staff.

User details (2)

The user details area consists of the login button, or the name of the signed-in user. The login button will take the user to login via HiDALGO2 Keycloak [1]. Also, by clicking on the dropdown menu, the user can log out. Further menu items, like settings, can be added to this menu later.

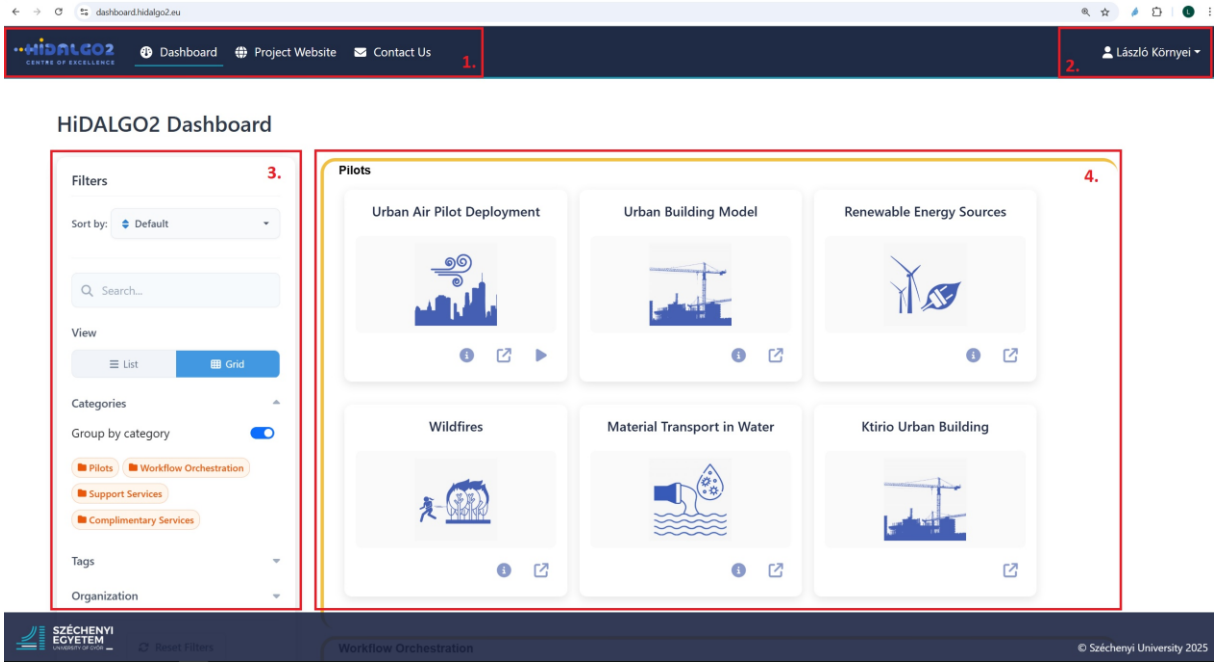

Figure 1. Screenshot of the default layout of the HiDALGO2 dashboard

Filter bars (3)

The filters here can be used to search for, sort or filter tile elements in the tile area. The tools here will be discussed in detail in section 2.2.

Tile area (4)

This area consists of elements representing the HiDALGO2 services. Services are represented by an icon and a title. The default action associated with the service is to connect to the appropriate site.

The basic layout of the dashboard is represented in the screenshot above. First time users may want to take time and get to know the services better or may be unfamiliar with some. To facilitate familiarization, two additional layouts are integrated into the dashboard. The first one, the list view, is activated by clicking the list icon in the view section of the filter bar. The layout can be seen on Figure 2. In list view – beside the service name and icon – a short description is also added to a widened tile of the service. Additionally, the organization and tags are also shown. All information here will be editable in the upcoming dashboard editor for users with appropriate access.

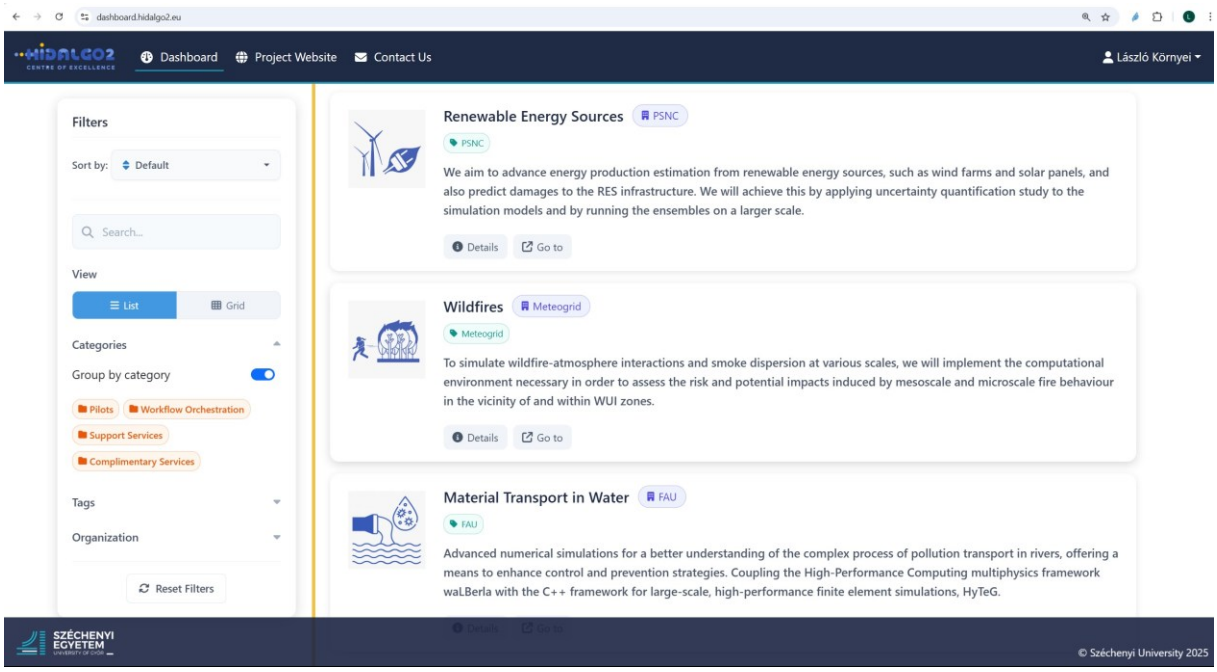

Figure 2. Screenshot of the list view of the HiDALGO2 dashboard

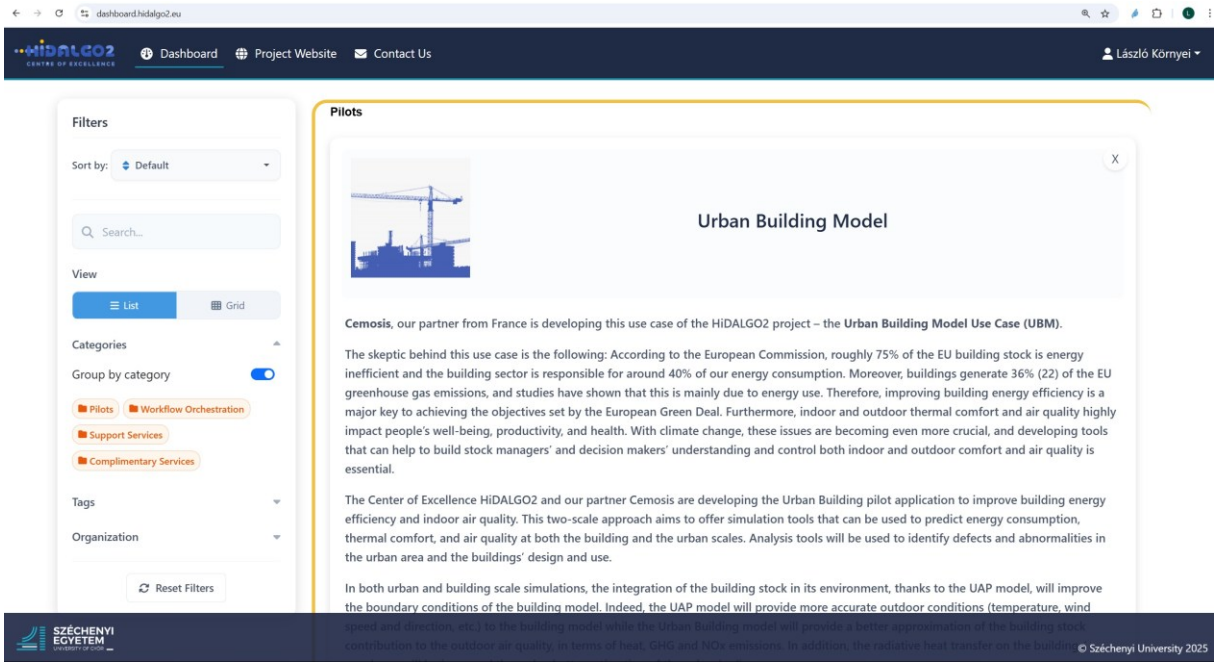

Figure 3. Screenshot of the detail view of the dashboard

The second additional layout is the detail view. Here, pilots or services are described in long form. This description can include embedded objects, like pictures, figures or external links. The long description is stored in markdown or HTML format and will be able to be edited within the dashboard editor for users with edit access.

2.2 Functionality

The primary focus of the dashboard is to gather HiDALGO2 services used by HiDALGO2 internal and external users and provide an intuitive, simple and seamless experience. In this section, currently implemented functionalities are addressed, while hinting some of the future development possibilities.

There are currently two main functionalities implemented into the dashboard. The first is aggregating services with optional detail and deployment functions, and the second is the search-filter-sort function on the left.

Service functionalities

The basic function of the dashboard is to display information about a service, and if clicked on, to take the user to the site of the service. This basic functionality is extended with properties and actions for various services. The current actions assigned to services are the following:

- Go to the service site
- Get details on the service
- Go to the pilot's appropriate WFO, and list associated applications for deployment to HPC.

Additionally, services have specific supplementary properties. Each service belongs to a specific organization and to a specific group. Also, each service can have several tags, which users can filter or search for. The groups currently defined are:

1. Pilots: This is the category where the pilots and pilot applications reside.
2. Workflow Orchestration: Access to the two supported WFO's, MathSO and QCG portals.
3. Complimentary services: Services directly attached to the workflow process, including pre- and post-processing, or data management
4. Support services: Services not directly attached to the workflow process, including documentations, training materials, and ticketing system.

Search-filter-sort functionalities

Selecting appropriate services from the tile area is supported by additional functionalities. The search-filter-sort pane on the left side of the site offers various facilities in this matter, as depicted on Figure 4. These tools are as follows:

1. Sort tiles by title, organization or category, in ascending or descending order.
2. Search for keywords in tile name or description.
3. Switch between list and grid formats.
4. Enable or disable grouping by categories.
5. Select filter by category, tags, or organization.

An additional reset button helps setting back filters to the default.

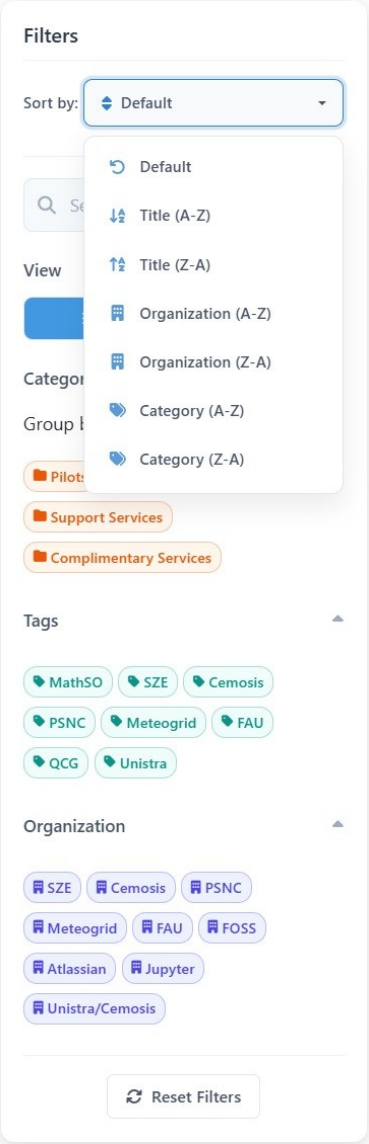

Figure 4. Screenshot of the search-filter-sort pane of the dashboard

2.3 Deployment

This section is focused on the deployment procedure of the HiDALGO2 dashboard. The focus is to follow the requirements and guidelines outlined in the earlier deliverable D2.4, as detailed in deliverable D2.7 section 4.3.

The current deployment is hosted at the PSNC Cloud, hostname sophora-60.man.poznan.pl, with alias to dashboard.hidalgo2.eu. The details of the virtual machine used can be found in deliverable D2.5 and are detailed in Table 1.

Table 1. Provisioning parameters for dashboard

Name	Dashboard		Prod	PSNC	IP	62.3.171.60	
Software	Nginx, JavaScript		Dev	N/A	IP	N/A	
Hardware	CPU cores	4	RAM	8 GB	Hard Disk	40 GB	
Domain	https://sophora-60.man.poznan.pl dashboard.hidalgo2.eu api.dashboard.hidalgo2.eu						
Description	The main dashboard instance for HiDALGO2.						

For the current versions, only one installation of the dashboard is used, as not all functionalities are integrated yet, including user rights management. The dashboard source code is hosted on the MathSO GitLab, while it is deployed within a docker virtual machine.

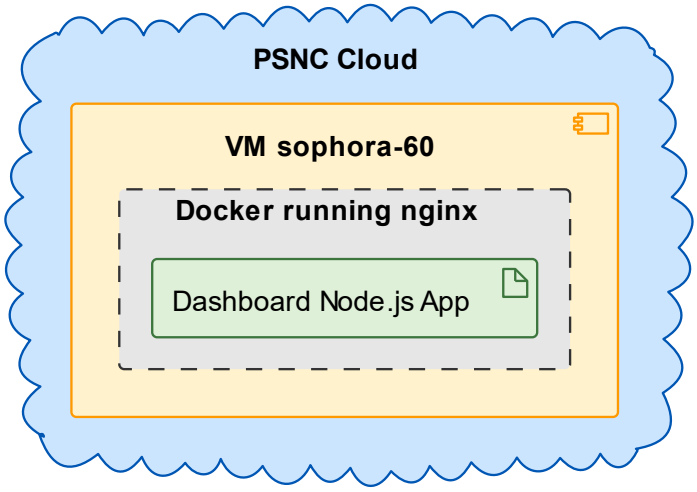

Figure 5. Dashboard deployment architecture

Currently, deployment is partially automated using docker for a reproducible build. Fully automated deployment and delivery (CI/CD) is planned through GitLab runners [2]. Implementation is planned to be completed by M36.

The Dockerfile uses a two-stage build [3]: first it pulls node:18, installs dependencies, and runs npm run build to output static assets into /app/dist. In the second stage it switches to nginx:stable-alpine, copies the built files into Nginx's HTML directory, injects a custom nginx.conf, and runs Nginx on port 1234 in the foreground to serve the dashboard app [4]. The modules and node versions used are listed in Table 2.

Table 2. Node modules and versions for Dashboard

Package	Version	Category
@fortawesome/fontawesome-svg-core	^6.6.0	dependencies
@fortawesome/free-brands-svg-icons	^6.6.0	dependencies
@fortawesome/free-regular-svg-icons	^6.6.0	dependencies
@fortawesome/free-solid-svg-icons	^6.6.0	dependencies
@fortawesome/vue-fontawesome	^3.0.8	dependencies
@tiptap/extension-hard-break	^2.11.0	dependencies
@tiptap/extension-image	^2.11.0	dependencies
@tiptap/extension-text-align	^2.11.0	dependencies
@tiptap/extension-underline	^2.11.0	dependencies
@tiptap/pm	^2.11.0	dependencies
@tiptap/starter-kit	^2.11.0	dependencies
@tiptap/vue-3	^2.11.0	dependencies
@vue/compat	^3.4.29	dependencies
bootstrap	^5.3.3	dependencies
bootstrap-vue-next	^0.24.6	dependencies
cropperjs	^1.6.2	dependencies
dompurify	^3.2.3	dependencies
file-saver	^2.0.5	dependencies
keycloak-js	^26.1.0	dependencies
marked	^15.0.6	dependencies
mitt	^3.0.1	dependencies
tiptap-extension	^1.0.1-alpha.0	dependencies
vue	^3.4.29	dependencies
vue-axios	^3.5.2	dependencies
vue-router	^4.3.3	dependencies

Package	Version	Category
vuex	^4.1.0	dependencies
vuex-persist	^3.1.3	dependencies
@vitejs/plugin-vue	^5.0.5	devDependencies
sass-embedded	^1.77.8	devDependencies
unplugin-vue-components	^0.27.4	devDependencies
unplugin-vue-fontawesome	^1.0.6	devDependencies
vite	^5.3.1	devDependencies

3 Dashboard services

The current section covers the services currently installed and planned in the HiDALGO2 ecosystem, with details on the level of integration of the various services. In the second part authentication topics are covered, including planned authentication levels and user registration, as well as a description of the Single Sign-On and Silent Sign-On authentication features already implemented.

3.1 Currently integrated services

The HiDALGO2 development and user environment consists of a number of services, which are hosted at different partner sites, and each with a separate link for access. The Dashboard will bring together and integrate access to all the required services, so that it is convenient for users and developers to use them. A list of services, and their current level of integration with the Dashboard is provided in Table 3.

Table 3. List of services

	Service	Description	Type	Status
1	Website	Project Website		Integrated as a link on the top bar
2	MathSO Portal	Workflow Orchestrator		Integrated, with single sign on and silent sign on
3	QCG Portal	Workflow Orchestrator		Integrated, with single sign on and silent sign on
4	Prototype	Compute Cluster		Available through WFO for simulation deployment
5	JupyterHub [5]	Jupyter Notebooks	Complimentary	Integrated, with Single Sign On through IDM page supported
6	IDM	Identity Management		Not to be directly integrated, but used through access to other services
7	Askbot	User Forums	Support	Integrated, with Single Sign On through IDM page supported
8	CKAN [6]	Data Management System	Complimentary	Integrated, with Single Sign On through IDM page supported
9	Bitbucket	Git Repository	Complimentary	Integrated – to be removed and replaced by GitLab
10	Zammad [7]	User Support	Support	Integrated, with Single Sign On through IDM page

				supported
11	Wiki [8]	Knowledge Management	Support	Integrated, with Single Sign On through IDM page supported
12	Open Project	Project Management	Complimentary	Integrated
13	Moodle [9]	Learning Platform	Support	Integrated, with Single Sign On through IDM page supported
14	SEMS	Energy Monitoring		not yet integrated
15	HedgeDoc [10]	Collaborative Note Taking	Complimentary	Integrated, with Single Sign On through IDM page supported
16	GitLab	Git Repository		to be integrated, HiDALGO2 Keycloak supported

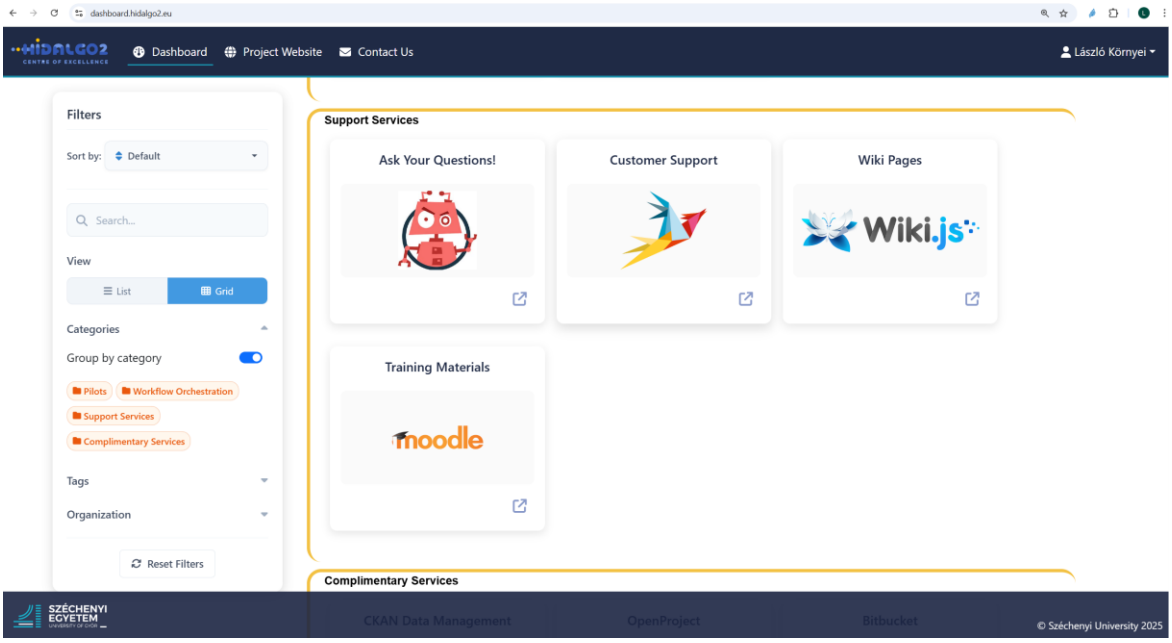

Figure 6. Screenshot of integrated support services

The integrated services have been divided into support and complimentary services. Complimentary services directly facilitate workflow, deployment and post processing, while support services provide features not directly related to the production workflow, such as documentation and training. Integrated services are shown on Figure 6 for support, and Figure 7 for complimentary services.

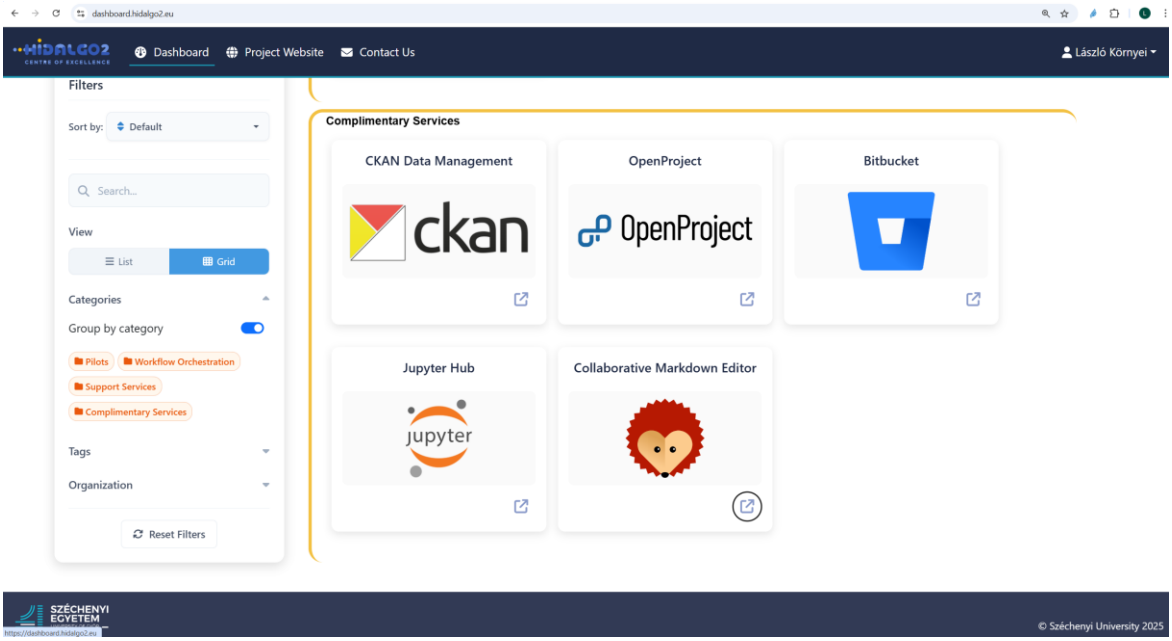

Figure 7. Screenshot of complimentary services

3.2 Authentication

The HiDALGO2 service ecosystem uses Keycloak for primary authentication. Keycloak provides fine-grained authorization by defining roles, policies, and permissions that govern access to application resources. Realm- and client-level roles can be assigned to users or groups, and adapters enforce these roles at runtime on HTTP endpoints, UI components, or API calls. For more advanced control, Authorization Services enable modeling of resources and scopes, the creation of role-based, time-based, or script-driven policies, and the binding of those policies into permissions—all managed centrally within the Keycloak server. For further details refer to D2.7 section 3.2 and [11].

3.2.1 Authentication levels

A four-tier authentication model is planned within the single HiDALGO2 Keycloak realm. The four core roles, **unauthenticated, external_user, internal_user** and **admin** will be treated separately.

1. The basic, **unauthenticated** role represents any visitor, who has not signed in. Visitors may get access to information, documentation or public training sites, that can be visited without need of authorization.
2. The **external_user** role will represent users, who have been authenticated via HiDALGO2 Keycloak system. They will be able to use applications for training,

testing, or production use. However, they will not be able to create or administer own services.

3. Authenticated staff or consortium members will receive the **internal_user** role, who will be able to create and administer services within the dashboard. This level will be able to configure services or sites to be available for unauthenticated, external or internal users. Internal users will be assigned to organizations and will be able to administer services of their own organization.
4. The **admin** role will be focusing on granting or withdrawing user rights. Also, it will inherit the right of an internal user.

Clients and resource servers are protected by Keycloak's built-in role enforcement or by its Authorization Services. Service ownership will be organization based, so any internal user within a given organization will be able to administer services of the given organization.

3.2.2 Single Sign On integration

This section outlines the end-to-end integration of Keycloak SSO [12] into a JavaScript application – like HiDALGO2 services – using the keycloak-js adapter. It covers the initial client setup in Keycloak, the authentication lifecycle (including silent SSO checks and PKCE security [13]), robust token management with proactive refreshing and error handling, seamless HTTP request handling via Axios interceptors, and runtime behaviour that keep sessions alive and secure. Together, these components ensure smooth user experience without manual re-authentication while maintaining strong security guarantees.

Keycloak Configuration

For client initialization, the integration uses the official keycloak-js library, a client-side JavaScript adapter based on OpenID Connect Keycloak. The Keycloak client is instantiated with the identity provider's URL, the target realm, and the unique client ID:

```
const keycloak = new Keycloak({
url: "https://idm.hidalgo2.eu",
realm: "hidalgo2",
clientId: "client-id-here"
});
```

Figure 8. Instantiating Keycloak client using target realm and unique client ID

Additionally, the following settings must be made. In the Keycloak Admin Console, the client must be set as public (Client authentication OFF) and only Standard flow enabled to suit a browser-based application. Valid Redirect URIs and Web Origins need precise configuration to prevent unauthorized redirects and ensure proper CORS handling.

Authentication Flow

Within the initial authentication, on application startup, calling *keycloak.init({ onLoad: 'check-sso' })* triggers Keycloak to detect an existing session without forcing a login redirect if none exists. Silent SSO is enabled by specifying *silentCheckSsoRedirectUri* during initialization, which loads a minimal HTML page in a hidden *iframe* to check authentication state without reloading the SPA. After the *iframe* receives tokens, it posts them back to the parent window, keeping the user's view uninterrupted. For enhanced security in public clients, Proof Key for Code Exchange (PKCE) is enabled by default in Keycloak v23+, which protects the authorization code flow against interception attacks by using a code challenge and verifier. The adapter automatically handles generating and sending the *code_challenge* and *code_verifier* parameters in the auth requests.

Token Management

Upon successful login, Keycloak exchanges the authorization code for an access token, refresh token, and associated expiry times, storing them in the adapter instance. The adapter's *kc.updateToken(minValidity)* method proactively refreshes the access token *minValidity* seconds before expiry, preventing mid-request expirations. By leveraging the *onTokenExpired* callback, the app can trigger additional refresh logic or user logout flows when refreshing fails. If the application receives a 403 (Unauthorized) response, the implementation retries fetching a fresh token via *updateToken()* before reissuing the original request. This approach ensures that transient authentication lapses are corrected automatically without user disruption.

Axios Integration

An Axios request interceptor checks the access token's validity prior to each HTTP call, invoking *kc.updateToken()* as needed, then attaches the latest bearer token in the Authorization header. The response interceptor watches for authentication failures (e.g. 401/403). On detection, it temporarily ejects itself to avoid infinite loops, calls a clean instance to refresh the token, updates the original request's headers, and re-issues it. After retrying, it re-attaches the interceptor to resume normal operation.

Runtime Behaviour

When the SPA loads, *keycloak.init()* runs silently, checks for existing sessions, and, if authenticated, retrieves user profile data for in-app use. Background silent checks and proactive refreshes run at configured intervals, keeping sessions valid without manual re-login. Protected routes only become accessible once a valid token is present, preventing unauthorized access to secured components.

By combining Keycloak's JavaScript adapter features—public client setup, silent SSO checks, PKCE security, and built-in token lifecycles—with Axios interceptor logic for HTTP resilience, this integration provides seamless, secure single-sign-on capabilities in modern SPAs. Continuous token renewal and error-driven retries ensure that sessions remain alive and that users enjoy uninterrupted access to protected resources.

3.2.3 User registration and management

Although automated user registration will become essential once the platform scales beyond a modest user base, the current number of accounts remains small enough for manual management. As a result, all aspects of user administration are centralized under the administrator role to ensure consistency and oversight. Presently, administrators handle new user creation through a secure console, manually inputting basic profile information and initial credentials. Once an account exists, administrators assign appropriate realm-level roles—such as "external_user" or "internal_user"—to control access to public, partner, or internal resources.

In addition to role assignment, administrators link each internal user to their respective organization within the consortium. Because every internal user is formally affiliated with one of the partner institutions, administrators verify organizational membership by consulting the designated organization leader before provisioning access. This manual confirmation step guarantees that only legitimate consortium members receive internal permissions. As the user community grows, these tasks will be automated—streamlining new-user onboarding, role allocation, and organization mapping—without sacrificing the rigorous access controls established during the initial rollout.

4 External services

The following chapter consists of services provided by consortium partners outside of the core provisioning teams. The services detailed here are not yet completely ready to be integrated into the dashboard, however, will be added later on.

4.1 Data management system

4.1.1 Data movement subsystems

The HiDALGO2 project provides infrastructure and solutions for the management of large datasets consumed (as input) and produced (as output) by pilot's simulations executed in HPC clusters and their movement across different data storages. As reported in [D4.1], the HiDALGO2 infrastructure is based on:

- **CKAN** [6]: a dataset-oriented data management system with user-friendly facilities for data sharing, and
- **Hadoop HDFS** [14]: a highly scalable, largely distributed, big-size dataset-oriented repository for fast reading (i.e. Write once, read many) operations.

Complementing the distributed Hadoop storage, HiDALGO2 also provides a highly distributed and scalable computing infrastructure, over the HDFS storage, based on YARN [15] and Spark [16], aiming at conducting HPDA analytics (see D4.3) through the HiDALGO2 Jupyter Hub (see D2.4 and D2.5).

The HiDALGO2 Dashboard already integrates the CKAN data management system for dataset publication and sharing, and the Jupyter Hub, for HPDA analytics. In future releases, the Dashboard could also integrate other services of the HiDALGO2 data management system, mostly aiming

- Administrators of the data management system to:
  - Supervise and admin the Hadoop HDFS,
  - Supervise HPDA analytics submitted to Spark and YARN, and
- Advance users to
  - Manage their storage on the HDFS,
  - Supervise their HPDA analytics.

The following HiDALGO2 services provides the aforementioned functionality to administrators and advanced users:

- **The HDFS Portal:** http://sophora-42.man.poznan.pl:9870/.
  - This portal enables administrators (and advance users) to check the status and configuration of the HDFS storage and users to browse and manage their content, offering a rudimentary file browser.
- **The YARN Portal:** http://sophora-42.man.poznan.pl:8088/

  - This portal enables administrators (and advanced users) to check the status and configuration of the YARN computing infrastructure and the state of submitted applications.
- **The Livy Portal:** http://sophora-42.man.poznan.pl:8998/ui
  - This portal enables administrators (and advance users) to check the status of sessions opened from applications triggered within the Jupyter Hub notebooks and to access to the Spark Portal for inspecting application execution details.
- **The Spark Portal:** accessible through the Livy Portal.
  - This portal enables administrators (and advance users) to inspect the resources and execution details of Spark jobs submitted from the Jupyter Hub notebooks

To users, HiDALGO2 offers a portal for HDFS:

- **The HUE portal:** http://prunus-106.man.poznan.pl/
  - This portal enables users to browse and manage their content in the HDFS storage. Other additional Hadoop services could be enabled on demand.

All the HDFS portals are secured with Kerberos, which is not straightforward compatible with Keycloak. Therefore, a different authentication mechanism is required, beforehand, to get access to the services. Fortunately, this authentication can be performed by end-users on their computers, using the Kerberos CLI client, being granted for 12 hours to get access to any Hadoop service, including all the aforementioned portal services.

4.1.2 Document management systems

UNISTRA developed a document management system built on Antora [17] and ASCIIdoc. Reports generated by the urban building simulation and feelpp.benchmarking workflows are written in ASCIIdoc and version-controlled in Git. Antora automatically converts these ASCIIdoc documents into web pages, ensuring that the website remains continuously updated.

Key features include:

- **Centralized Version Control**: Simulation reports, including urban energy scenarios and benchmarking results, are stored in Git, ensuring meticulous change tracking and traceability.
- **Automatic Web Site Generation**: Antora converts the ASCIIdoc content into dynamic web pages that are automatically deployed and kept up to date.

- **Enhanced Accessibility and Transparency**: Automatically published documents provide immediate access to critical simulation data and performance analyses for users and partners, thereby supporting informed decision-making and efficient collaboration.

The websites are:

- https://cases.ktirio.fr: use cases report for the cities of the project and more
- https://bench.ktirio.fr: benchmarking reports of the pilot application on EuroHPC systems.

While the UNISTRA DMS platform is a standalone platform on its own, it is currently adapted to and integrated into the HiDALGO2 ecosystem.

4.2 Ktirio

The **Ktirio Platform** is a suite of tools developed by UNISTRA under the CoE HiDALGO2 initiative that facilitates the simulation and analysis of urban building energy performance. The platform integrates various modules that enable scenario preparation, simulation execution, and performance analysis.

4.2.1 Key Components and Integrations

- **Ktirio.cases**: A web-based module designed to manage and explore simulation results. It automatically aggregates outputs from urban building scenarios and organizes the data for statistical and comparative analysis.
- **feelpp.benchmarking**: This component evaluates simulation performance on HPC infrastructures. It collects and presents benchmarking results (including detailed performance analyses on EuroHPC machines), ensuring continuous optimization of simulation workflows. feelpp.benchmarking is part of the Feel++ platform on which Ktirio is built.
- **Ktirio GUI**: An interactive graphical interface (available as both a desktop application and via WebAssembly) that allows users to configure simulation scenarios, inspect input data, and visualize simulation outcomes.
- **Ktirio Urban Building**: the pilot application for building energy simulation deployed via containers on EuroHPC systems.

4.2.2 Workflow

1. **Report Preparation**:
    - Simulation reports (for urban building scenarios) and benchmarking results are prepared in ASCIIdoc.

- These documents are version-controlled in Git to ensure efficient management and historical tracking.

2. **Antora Automation**:

   - Ktirio.cases and feelpp.benchmarking integrate automated processes to transform these ASCIIdoc documents into web pages.

   - Antora continuously builds and deploys the website, making sure that the latest information and analyses are always available.

3. **Simulation Execution and Analysis**:

   - Simulation data are prepared by Ktirio GUI and deployed on our data management platform (Girder and CKAN).

   - The platform utilizes HPC resources and containerized environments (Singularity, Apptainer) to run energy modeling simulations.

   - Simulation outputs are processed and visualized using Ktirio GUI as well as Paraview.

4.2.3 Benefits

- **Continuous Updates**: Automation via Antora ensures that all reports and analyses are always current, reflecting the latest simulation advancements and results.

- **Transparency and Traceability**: Centralizing documentation in ASCIIdoc and versioning it via Git provides clear, secure management of numerical simulations.

- **Collaborative Optimization**: The integrated modules facilitate efficient information exchange among teams, promoting ongoing improvements in energy simulation and performance evaluation methods.

The Ktirio Platform is a strategic tool for urban building energy modeling, offering an integrated, automated, and collaborative solution. The figure below illustrates the Ktirio platform.

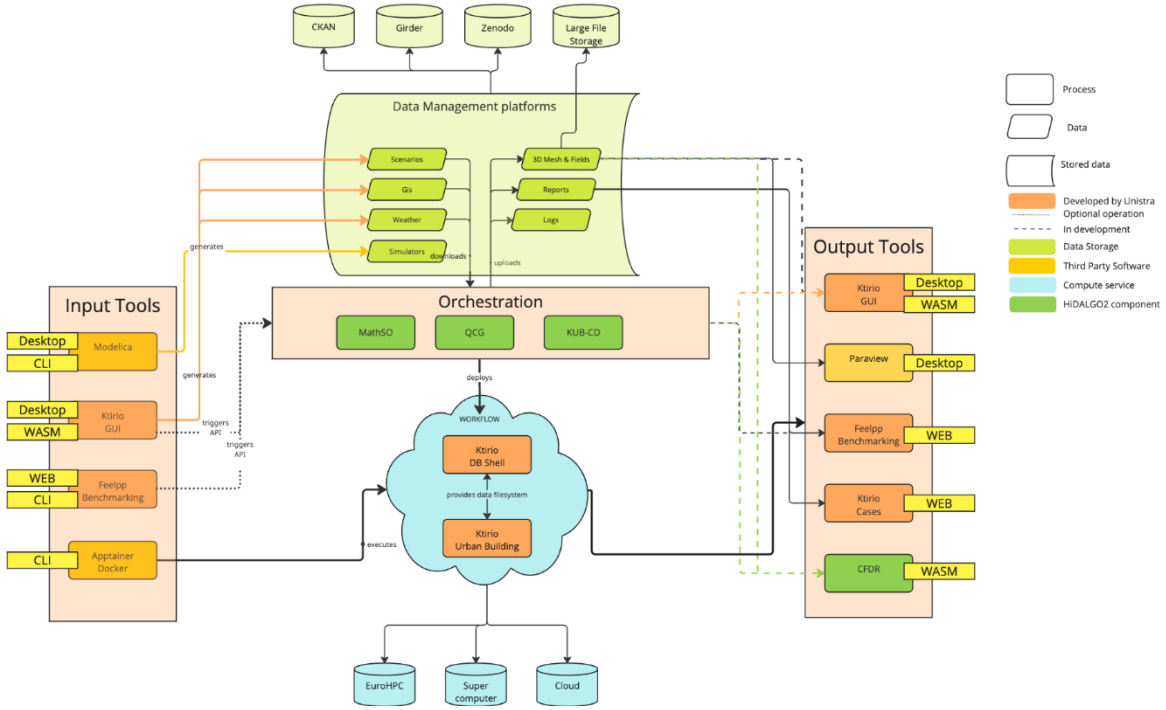

Figure 9. Architecture diagram of the Ktirio platform

While the Ktirio platform is a standalone platform on its own, it is currently adapted to and integrated into the HiDALGO2 ecosystem.

4.3 Visualization System

While visualization related tasks are reported in Deliverable D4.7 within WP4, this section addresses the details regarding integration of the visualization subsystem into the HiDALGO2 ecosystem. The primary method is tied into the MathSO or QCG workflow orchestrators, as visualization is tightly coupled to the post processing part of a simulation workflow. Additionally, a map preview functionality is to be integrated to facilitate the selection of simulation boundaries.

4.3.1 Visualization in the CKAN system

The aim of this section is to describe the service for visualization of simulation results after the results have been removed from the WFO. Most visualization workflow consists of two steps: analysis of the bulk simulation result data, to create visualization primitives, and the actual on-screen representation of the primitives at hand. While most 3D simulations may have extended amounts of output data, moving this amount to the client machine for post processing might not be the most effective way. It is less usual, that full scale 3D data gets visualized, so it is useful to generate slices, surface cuts, streamlines, or other geometries, which take much less space.

Pre-generated, or pre-processed visualization data can be stored on a data management system, like CKAN. It is useful to visualize this data independently, so an appropriate visualization service is to be deployed. The service is to be reachable from the dashboard and connect data repositories with visualization facilities.

4.3.2 Map preview functionality

In addition to the basic functionality of cloning simulations already carried out in the HiDALGO2 workflow orchestrators, by selecting from a list of available simulations the one of interest to the user, new simulations will be able to be carried out through the Dashboard.

The possibility of launching new simulations in the system will be conditioned by the available dates with meteorological data. In case the user wishes to perform a simulation for a date that is not available, the download of the meteorological information for that date must be done through CKAN and wait for the system to show the availability of the data once downloaded from the ECMWF [18]. This process may take one or several days.

Once the date has been selected, the area of interest will be selected. To do so, the system will display a map of Europe and allow the selection of the area of interest. The map will display reference information from Open Street Map [19] or some other similar service and will provide the possibility to modify the zoom level and to move the displayed area to the right, left, up or down. There will be two ways to select the target area:

- By entering the central coordinate of the area on the map, the system will allow entering the latitude and longitude coordinates of the central point; once this value is entered the system will display on the map the selected point and the surrounding area with a zoom level suitable to identify the main elements of interest on the map (rivers, lakes, bodies of water, relief, roads, population areas). A square centred on the selected point and with a side of 20km will be displayed on the map.
- By selecting the area directly on the map, the system will display a 20x20 km square and allow to move it to the user's area of interest. The displayed map will have the possibility of modifying the zoom level and displacement of the selected area by dragging with the mouse.

Once the area of interest has been selected, the system will allow the introduction of the outbreaks of the fire to be simulated, by introducing the geographic coordinates of the outbreak (LAT/LON) together with the date and time at which the ignition started at that point (yyyy/mm/dd hh/mm/ss). The entered sources shall be displayed on the preview map.

5 Feedback, suggestions and additional requirements

As detailed in the deliverable D2.7, two main campaigns were planned for the collection of the requirements of the dashboard. Up to the time of this deliverable, the first campaign was completed, focusing on collecting feedback from internal partners, including their opinions and requirements. The internal collection was performed by a shared questionnaire. The following section details the design, creation and management of the questionnaire. Afterwards the results are summarized, followed by the analysis of the results.

It is important to consider that the dashboard presented in the current document has already been improved according to the feedback provided here.

5.1 Questionnaire design

The questionnaire was created and conducted using google forms. Links to the form was delivered by the HiDALGO2 mailing list to reach all partners. It was important to have at least one response from every partner.

The questionnaire was short, focusing on four key parts of the dashboard: design, layout, functionalities and usability and responsiveness. These parts were defined in the questionnaire in the following way:

- **Design**
  Design consists of graphical elements, like icons, fonts, pictograms, style of visual elements, like boxes.
- **Layout**
  Layout consists of the placement and sizing of the various elements, including management of empty space.
- **Functionalities**
  Functionalities include the ability to search and filter tiles, the support for a detail page, tags and sorting. Later functionalities will include user roles and editing. Who has access to editing, please also include its rating here.
- **Usability and responsiveness**
  Usability means that its functionalities efficiently support daily work and activities. Responsiveness refers to the speed of the portal.

All key parts were to be rated from 1 to 5, with the various values defined on the questionnaire in the following way:

1. It is not acceptable this way, and the following issues need to be addressed as soon as possible.
2. It is acceptable for now, but the following issues need to be addressed in a timely manner.

3. It is OK; however, I do have the following suggestions, which could significantly improve this.
4. It is good; however, I think that it could be improved. – Please give feedback on what you feel that could be improved.
5. It is perfect as it is, it should not be changed. – Please give feedback on what you liked best.

Participants were required to fill out an additional long format text box for thoughts, opinions and suggestions, one for detailing each valuation. Participants were also asked if there were any services to be added additionally to the dashboard, of if there were any services to be removed. Participants were also to provide their e-mail address, so that any inconsistencies can be addressed afterwards.

As the dashboard has already been improved since the questionnaire, issues that were already been addressed will be marked as such.

5.2 Questionnaire results

Altogether, 13 answers were submitted. In this section, basic statistics of the answers will be shown, including key responses from long text sections. Eight partners were submitting answers. Most partners were represented by one answer, while MeteoGrid and PSNC submitted two, and HLRS submitted four.

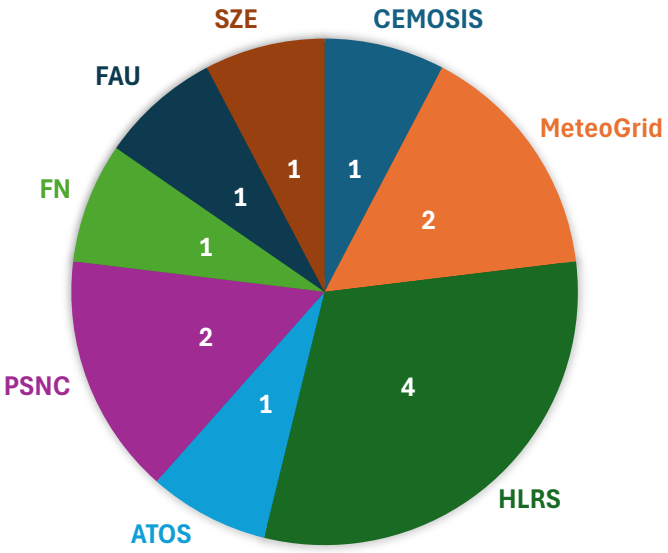

Figure 10. Distribution of questionnaire answers from partners

For the four major aspects that were analysed, results are presented as simple pie-charts, showing only number of selections per answer. While all answers will be listed

in “Annex I. Internal feedback on the Dashboard”, some selected feedback will be presented here. Please note that while not all issues are addressed in this document, all reported issues and opinions will be considered.

Design

The first topic is design. Results for numerical valuation are shown on Figure 11. While most of the votes (11) are 4 – good or 5 – perfect, some votes (2) are 2 – acceptable.

Figure 11. Questionnaire results from internal partners evaluating design

The remarks regarding acceptable valuation are the following, in bullet points, focusing on improvement suggestions. Some of these recommendations are already considered in the current implementation:

- Some optimizations are needed for the look & feel,
- The interface could benefit from a more polished, “fancier” visual style,
- Category sections currently feel cramped and collide with one another,
- Reduce the font size of the description text that appears after clicking the “Details” button.

Some recommendations are considered for the next implementation:

- Better differentiation between groups is recommended (e.g., add descriptive text, not just boxed elements),
- Replace the generic pilot icons with screenshots of the actual simulations/visualizations,
- Use real simulation images to improve relevance and context.

Some of the most important remarks from the other opinions:

- Fix the item-renaming bug so editing a name updates the existing item instead of creating a new one,
- Allow custom logos and replace generic pilot icons with real simulation/visualization screenshots,
- Tone down the “Details” description font size and structure this section with clear text and diagrams or images,
- Visually differentiate category sections (e.g. rethink the coloured frames, add headers or brief descriptions) so they don’t collide,
- Standardize the “Go to” button behaviour for logged-in and logged-out users (or provide a separate post-login portal link).

Layout

Next topic is layout. Results for numerical valuation are shown on Figure 12. Most valuations (11) are 4 – good or 5 – perfect, however there is one vote on 1 – not acceptable.

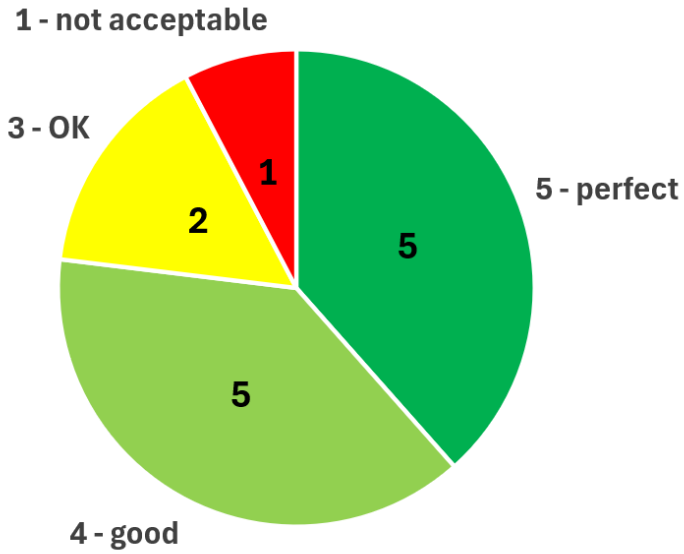

Figure 12. Questionnaire results from internal partners evaluating layout

The remark regarding not acceptable valuation is the following:

- Spacing needs to be improved: the paddings/margins are very often too small.

Upon further discussion with the respondent, some more visual clarification was presented, as shown on Figure 13, focusing on the following:

- more spacing between tile icon and action button,
- more spacing between tile title and tile icon,
- inappropriate size of font and button for action button,

- more spacing between group title and group separator.

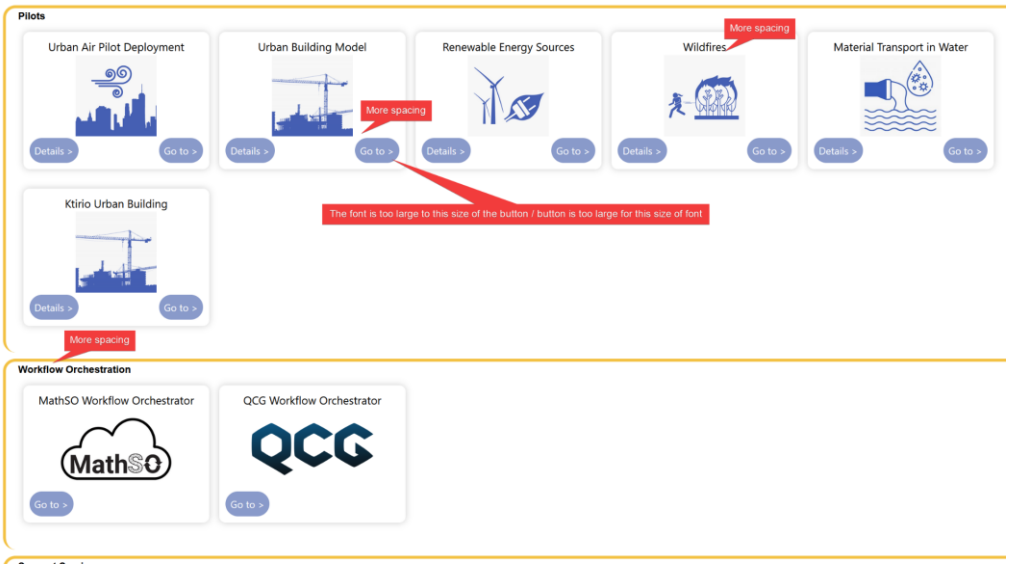

Figure 13. Screenshot of the dashboard (previous version) pinpointing issues with spacing between elements

In the current version, the above issues have been treated as shown on Figure 14. The issues have been treated as follows:

- spacing between tile title and icon has been increased – see arrow 2.
- action buttons were redesigned and repositioned, moving text to mouse over help. – see arrow 3.
- Spacing between separator and group title has been adjusted – see arrow 1.

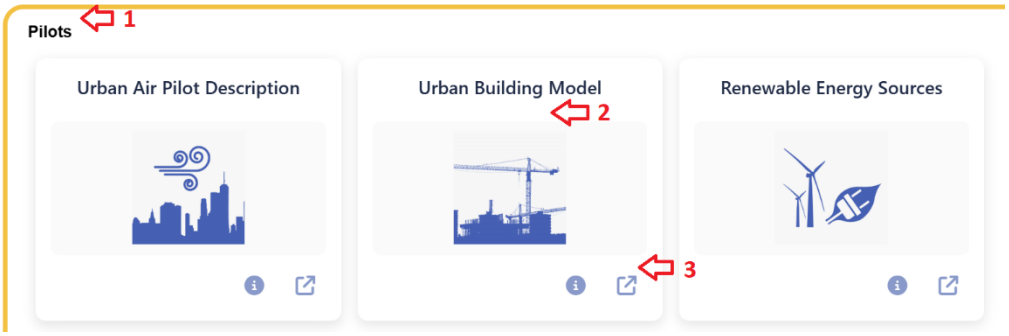

Figure 14. Screenshot of the dashboard (current version) pinpointing issues addressed

Summary of the most important remarks from the other opinions:

- Fix item-renaming bug. Editing an item's name should update the existing entry instead of creating a duplicate,
- Enable custom logos & improve icons. Let users upload their own logos and replace generic pilot icons with high-resolution, context-relevant visuals,

- Polish the “Details” view. Reduce the description font size and organize content with clear headings, concise text, and supportive diagrams or images,
- Differentiate category sections. Rethink or refine the coloured frames around groups; add distinct headers or brief descriptions so blocks don’t visually blend,
- Standardize “Go to” navigation. Ensure the “Go to” button behaves the same whether users are logged in or not or provide a clear secondary link for post-login portal access.

Functionalities

Next topic is functionalities. Results for numerical valuation are shown on Figure 15. Most valuations (9) are 4 – good or 5 – perfect, however there is one vote on 2 – acceptable.

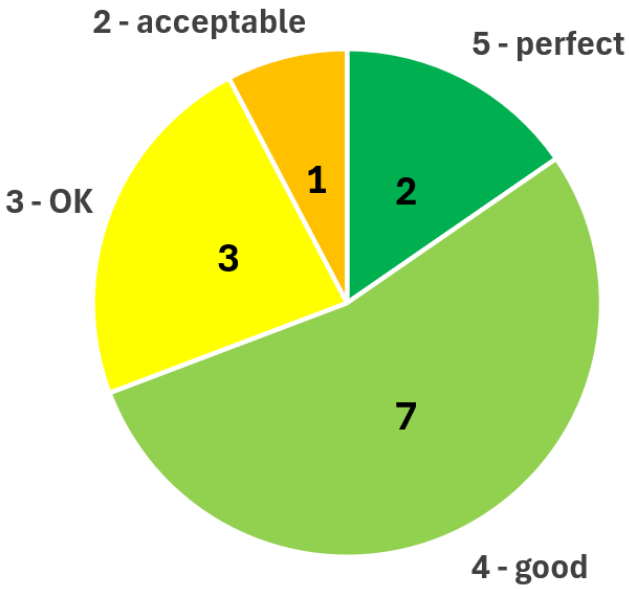

Figure 15. Questionnaire results from internal partners evaluating functionalities

The remark regarding acceptable valuation is the following:

- For the detail view:
  - Back navigation unavailable. The browser’s Back button cannot be used to exit the Details view,
  - No new tab opened. Details open in-place, so the context of the main dashboard is lost,
  - Unresponsive logo and Dashboard button. Clicking the HiDALGO2 logo or the “Dashboard” button does nothing,

    - Hidden close control. The Details panel is mistaken with the whole page; the only way to go back is the small “X” close button.
- Suggestion for fixing Details panel:
    - Make the panel visually distinct (e.g., smaller size, slide-in animation) and ensure clicking the logo or Dashboard closes it or returns home.
- Search term mismatch. Typing “RES” doesn’t bring up “Renewable Energy Sources” despite it being in descriptions; instead, it shows unrelated results.
- External navigation clarity. “Go to” links leave the dashboard for the project site without indication—suggest adding an external-link icon to clarify.

All the above issues have been addressed in the current version of the dashboard.

Summary of the most important remarks from the other opinions:

- Overall usability: Dashboard is convenient, visually appealing, and responsive.
- Service vs. tag confusion: It’s unclear how “support” vs. “complimentary” services differs, and tags mix partner organizations with service names—better to use tags as functional keywords (e.g., “Data Visualization,” “Training”).
- Broken/misdirected links: The QCG link is broken, and the UAP pilot link lands on MathSO’s homepage without showing pilot content.
- HPC configuration clarity & security: Fields (SSH key vs. password vs. passphrase) lack explanations; the “already exists” naming bug blocks new entries; storing private keys in plain text raises data-safety concerns—recommend on-portal key generation/upload and a clear data-storage disclaimer.
- Search & filtering: Both are fast and useful; could be further enhanced by indexing content behind pilot links for more comprehensive results.

The issues above have been partially addressed. Some feedback is regarding the MathSO portal, so it will be considered at the appropriate place.

Usability and responsiveness

The last topic is usability and responsiveness. Results for numerical valuation are shown on Figure 16. Almost all valuations (12) are 4 – good or 5 – perfect.

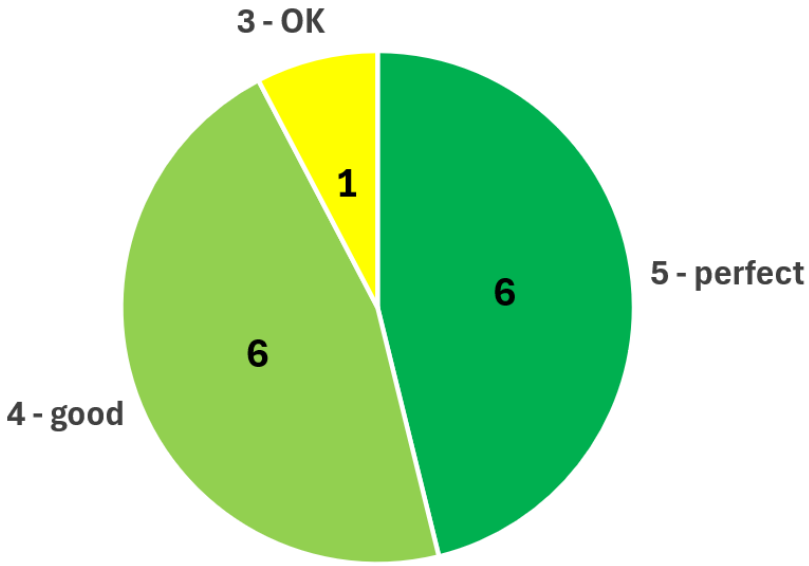

Figure 16. Questionnaire results from internal partners evaluating usability and responsiveness

Summary of the most important remarks from opinions:

- The dashboard is consistently fast, responsive, and easy to use,
- Initial usability and performance of available elements are very good,
- HiDALGO2 services are well collected; future iterations could link service summaries or data back into the dashboard,
- The dynamics of the search panel and the mosaic layout are smoothly rendered with minimal delay,
- Minor lag noted when re-rendering service icons after filtering or returning from a details view,
- Modelling capabilities' responsiveness remains to be evaluated as more features roll out,
- Potential future enhancements:
    - Connect use-case content with underlying code,
    - Integrate AI-driven suggestions to aid solution selection.

Services to be added

The respondents suggested adding the following services to the dashboard (see Annex I. Internal feedback on the Dashboard for full responses):

- Plan to add the Ktirio GUI (WASM) component, which serves as the input data generator for the Ktirio Urban Building pilot,
- Current services (e.g., hdoc.hidalgo2.eu, git.hidalgo2.eu) meet needs, but modelling capabilities still require evaluation,
- Data management considerations will be addressed in deliverable D2.9,

- User values easy access to source-code repositories and suggests listing available code projects with direct links, rather than only Bitbucket navigation.

Services to be removed

The respondents noted the following regarding removing services (see Annex I. Internal feedback on the Dashboard for full responses):

- Renamed the pilot from UBM to Ktirio Urban Building (KUB); remove all references to “UBM”,
- No other services need to be added at this time,
- Replace Bitbucket with git.hidalgo2.eu GitLab (supports Keycloak SSO),
- Exclude Zabbix/Monitoring from the dashboard—it's an internal PSNC tool,
- OpenProject: not widely used (only by USTUTT); if included, restrict it to internal HiDALGO2 users (same as Zabbix),
- Dashboard works well as a centralized HiDALGO services hub,
- Ktirio Urban Building isn’t really a “pilot”—it should be categorized separately,
- More detailed service info is welcome, since filtering and searching make navigation easy,
- Question raised: Should a Wiki or documentation be listed as a service on the dashboard?

5.3 Results analysis

A summary of the numerical evaluations is shown in Table 4. Regarding averages, overall satisfaction is high, as all averages are above 3.8 on a 5-point scale. Design had two acceptable, and layout one unacceptable score. Functionality and usability had no rating below OK. Usability & responsiveness got the highest average score of 4.25.

Table 4. Summary of the numerical evaluations in the questionnaire

Aspect	1	2	3	4	5	Average
Design	0	2	0	8	2	3.83
Layout	1	0	1	5	5	4.08
Functionality	0	0	3	7	2	3.92
Usability & Responsiveness	0	0	2	5	5	4.25

Key topics and suggestions

Respondents converged on four headline areas for improvement. First, design refinements such as offering smaller, resizable tiles, polishing colour accents and frames, and replacing generic icons with custom logos or real-simulation screenshots. Second, layout adjustments including better use of whitespace, consistent alignment of the filter pane with the service grid, responsive behaviour across device sizes, tabbed or grouped views to reduce scrolling, and clearer separation of category blocks. Third, functional enhancements—most urgently a more intuitive Details panel (e.g. enabling browser-back, logo/Dashboard clicks or a slide-in animation to close it), smarter search that indexes descriptions and behind-link content, true keyword-based tagging instead of mixing organization names, and fixing the item-renaming bug. Finally, usability and navigation tweaks, such as adding an external-link icon to “Go to” buttons, monitoring minor icon-rendering lags after filtering, and ensuring consistent off-dashboard navigation, along with planning to integrate modelling workflows and services like the Ktirio GUI and GitLab while pruning deprecated entries (UBM pilot, Bitbucket, internal-only tools).

Service Additions & Removals

The following services are requested to be added:

- Ktirio GUI (WASM) as the Urban Building input-data generator,
- git.hidalgo2.eu (GitLab) instead of Bitbucket, leveraging Keycloak SSO.

Both services are integrated at the time of the delivery submission.

The following services are requested to be removed:

- UBM pilot (renamed to Ktirio Urban Building),
- Bitbucket (replace with GitLab),
- Zabbix/Monitoring and OpenProject from the public dashboard – restrict to internal usage only.

While UBM pilot and Bitbucket are planned to be removed at the next iteration, Monitoring and OpenProject will be set to restricted access to internal users only.

6 Conclusions and roadmap

6.1 Conclusion

The deliverable D2.8 represents a major advance in unifying the HiDALGO2 services under a coherent, user-centric dashboard that satisfies both internal and external stakeholder requirements. Over the course of its development, the dashboard has matured from an initial proof-of-concept into a polished platform where branding, typography, and responsive design come together to guide users effortlessly to the tools they need. Integration with Keycloak single-sign-on ensures secure, streamlined access, and the two-stage Dockerized deployment on the PSNC Cloud has laid the groundwork for a robust, reproducible release process.

Feedback gathered through the internal stakeholder questionnaire has been instrumental in refining every aspect of the user experience. Spacing and alignment were tightened to improve readability, search and filter controls were optimized for speed and relevance, and the Details panel was redesigned to feel more intuitive and less disruptive. Custom logos and real-simulation screenshots have replaced placeholder icons, tag semantics have been clarified, and “Go to” links now clearly signal when a user is leaving the dashboard. Each change responds directly to partner input, illustrating HiDALGO2’s commitment to an iterative, feedback-driven development cycle.

Looking ahead, the dashboard will continue to evolve as new services—such as the Ktirio GUI, advanced data-management portals, and integrated visualization subsystems are brought online. Deeper coupling between pilots and dashboard summaries will further enhance usability and productivity. By combining a solid technical foundation with an unwavering focus on user-driven improvements, D2.8 sets the stage for a truly comprehensive, adaptable portal that will serve as the HiDALGO2 ecosystem’s principal point of entry for years to come.

6.2 Roadmap

Planned implementations for the first iteration release (2025.06.01):

- Access to all pilot applications using only dashboard and linking to the proper WFO portal,
- DMS frontends, including Hadoop frontend for users,
- Ktirio-GUI WASM integration,
- Ktirio Urban Building documentation platform integration,
- Enable visualization through the dashboard,
- Monitoring pending jobs, HiDALGO2 tools and EuroHPC site status,
- Clarification of simulation environment definition configuring namelist and submit script,

- Further analysis of the feedback in the questionnaire, identifying additional features to implement,
- First implementation of user rights management and access levels.

Planned implementations for the final version release (2026.01.01):

- Final implementation of user rights management and access levels,
- Implementation of identified features from the first questionnaire,
- Implementation of simulation environment definition and boundary conditions download into dashboard or WFO.

References

[1] The official KeyCloak website https://www.keycloak.org
[2] A short description of GitLab Runners https://docs.gitlab.com/runner/
[3] Docker two-stage build documentation https://docs.docker.com/build/building/multi-stage/
[4] Nginx https://docs.nginx.com/
[5] JupyterHub official site https://jupyter.org/hub and documentation https://jupyterhub.readthedocs.io/
[6] CKAN official site https://ckan.org/
[7] Zammad https://zammad.org/
[8] Wiki https://js.wiki
[9] Moodle https://moodle.org
[10] HedgeDoc: https://hedgedoc.org/
[11] Keycloak policy enforcer https://www.keycloak.org/securing-apps/policy-enforcer and Keycloak Authorization Services Guide https://www.keycloak.org/docs/latest/authorization_services/index.html
[12] Keycloak SSO: https://www.keycloak.org/docs/latest/server_admin/index.html#_features
[13] Proof Key for Code Exchange https://www.rfc-editor.org/info/rfc7636 and https://oauth.net/2/pkce/
[14] Hadoop HDFS https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-hdfs/HdfsUserGuide.html
[15] Yarn: https://hadoop.apache.org/docs/stable/hadoop-yarn/hadoop-yarn-site/YARN.html
[16] Apache Spark https://spark.apache.org/
[17] Antora https://antora.org/
[18] ECMWF https://www.ecmwf.int/
[19] Open Street Map https://www.openstreetmap.org/

Annexes

Annex I. Internal feedback on the Dashboard

Questionnaire

Guidelines

Internal feedback of Hidalgo2 partners on the current implementation of the Hidalgo2 dashboard. Please, take your time to read carefully. The questionnaire is short.

We would like to have a short feedback on the topic. Please rate design, layout, functionalities and responsiveness. Please give feedback for every point, sharing your suggestions, opinions, if you have found any issues, or if you find something particularly positive. Also, please indicate, if any additional services need to be integrated into of removed from the dashboard.

Rating will go from 1 to 5, with the numbers having the following meaning:

- 5: It is perfect as it is, it should not be changed. – Please give feedback on what you liked best.
- 4: It is good, however I think that it could be improved. – Please give feedback on what you feel that could be improved.
- 3: It is OK, however I do have the following suggestions, which could significantly improve this.
- 2: It is acceptable for now, but the following issues need to be addressed in a timely manner.
- 1: It is not acceptable this way, and the following issues need to be addressed as soon as possible.

Independent from the rating, please share your insights, suggestions, and any issues you have found. Please, also provide your e-mail, so we can contact you later, if any clarification is needed.

Questions

- Please rate the design of the dashboard. Design consists of graphical elements, like icons, fonts, pictograms, style of visual elements, like boxes.
- Please share your thoughts, opinions and suggestions for design.
- Please rate the layout of the dashboard. Layout consists of the placement and sizing of the various elements, including management of empty space.
- Please share your thoughts, opinions and suggestions for layout.
- Please rate the functionalities of the dashboard. Functionalities include the ability to search and filter tiles, the support for a detail page, tags and sorting.

Later functionalities will include user roles and editing. Who have access to editing, please also include its rating here.

- Please share your thoughts, opinions and suggestions for functionalities.
- Please rate the usability and responsiveness of the dashboard. Usability means, that its functionalities efficiently support daily work and activities. Responsiveness refers to the speed of the portal.
- Please share your thoughts, opinions and suggestions for usability and responsiveness.
- Please indicate, if there are any additional services to be added to the dashboard. Please refer to the TOC of D2.8 for the services that are planned, but not yet added. These will be added, so you do not need to mention them here.
- Please indicate, if there are any additional services to be removed from the dashboard.

Answers

Please rate the **design** of the dashboard. Design consists of graphical elements, like icons, fonts, pictograms, style of visual elements, like boxes.
4 - good
4 - good
4 - good
4 - good
5 - perfect
4 - good
4 - good
4 - good
2 - acceptable
2 - acceptable
4 - good
5 - perfect

4 – good

Please share your thoughts, opinions and suggestions for **design**.
items are a bit big and good be smaller. We cannot edit an item name, if we do it creates a new item(bug). We should be able to change the logos, they are necessarily properly representing the pilots,
Details sections with more easy and structured text. Some diagrams and images should help to understand it more quickly.
I miss a brief explanation about what you will find in the dashboard
The design has improved a lot. Also, the icons/design is more responsive to changing window dimensions. However, I think can still make it overall look more modern/nicer, but this is not critical, maybe for the next iteration.
It is very intuitive and easy to use, I personally like it very much.
1- The rounded box surrounding the services on each category, clearly marked on top-left corner in orange stroke, but that vanishes on the bottom-right is weird to me, as it gives the impression the block gets broken on that side. Not a strong point, just a look and feel comment. 2- Another aspect comment is about the black down arrow head icons for expanding and collapsing the tags and organization categories in the leftmost panel. They are rather simple, with no high resolution. 3- I would change the dashboard introduction service: "Welcome to the HiDALGO2 Dashboard!" with some brief description of its purpose and functionality.
Great and elegant approach for grouping. The general look is modern and enterprise. However, the search panel background colour looks a little bit raw to me. But, maybe it's only due to my display.
I do not understand the functionality of the button "details" on each use case.
I think that there is a few thing for optimisation in regards of look&feel. The overall design idea is reasonable, so I like the focus on key information. However I would opt for a bit more fancy overall look. Now the sections devoted to different categories (groups) seem to collide each other. I think it would be great to distinguish them somehow. For example, they may have some descriptions, not only boxes with elements.

I think we should change the icons for the pilots. Much better to have a screenshot of the actual simulations/visualization than this generic ones. When clicking on details button, the description font is too big
The yellow frames (grouping) are too distracting and not helpful, the titles are enough. The home page of `portal.hidalgo2.eu` is totally necessary. Why only a logo and not an explanation of how to use the portal or an overview of the functionalities? I think it is super confusing that if you click on `Go to` you are always directed to some contents of the HiDALGO2 web presentation. BUT, when you log in, you get directed to the portal (MathSO-tool?), when you click go to at Urban Air Pilot but for the others you still get directed to the web presentation. I don't like having a different outcome from the same button depending on when I'm logged in or not. Rather, add another button that directs to the page if you are logged, similar to when some fields become available only after logging in. I think the User Portal (MathSO?) could also be added to the register on top: `Dashboard`, `HPC Portal`-(or something similar), `Project Website`, and `Contact Us`. I'm not a big fan of the design of the utility field with `Search`, `Grouping`, `Tag-filter`, etc. I think the gray box is not matching the rest of the design. It really distracts from the rest of the page. Otherwise, it's nice and clean. I like it.
Clear design, easy to use. I especially like the search function. It would be great if the search also could search in the contents *behind* the pilot links (not only within dashboard.hidalgo2.eu). E.g., within the contents of the site https://www.hidalgo2.eu/urban-air-project/.
pretty slim

Please rate the **layout** of the dashboard. Layout consists of the placement and sizing of the various elements, including management of empty space.
4 - good
4 - good
5 - perfect
3 - OK
4 - good
4 - good

3 - OK
5 - perfect
1 - not acceptable
4 - good
5 - perfect
5 - perfect
5 - perfect

Please share your thoughts, opinions and suggestions for **layout**.
a grid layout would be nice organized by category (pilot, orchestration) or by name or by recent use. Allow for smaller items/views of the services (they are a bit big by default) Adapt for different screen sizes (from cell phones, tablets to very large screen) and use nicely the whole screen (not too crowded though)
Simple and good looking, it is very effective
I like the simplicity of the layout avoiding overloading information and elements that may delay accessing the page
I am not so sure about the Panel where we can change the grouping, categories etc., if we should move it or use a more compact form for it. Maybe in the next iteration, the overall design could use the space differently, but I am not sure how.
I like the layout very much. Personally I would also like to reduce the size of each project so that I can have an overview. It would be great if it is possible to provide resizing option.
1- I would align the top of the leftmost panel that manages the layout organization with the top of the service block delimiter on the right 2- To avoid long scroll down, I would organize the selected categories in tabs, so accessing the services on other categories does not require to scroll down and they get visible on the top page, in case the grouping checkbox is marked.
I've got 31.5" display with resolution 2560x1440, Ubuntu Gnome, Chrome and scaling set to 100%. With that size, most pages will leave some space on the sides.

Here, the content looks very wide, even too wide and it is a little hard to read that wide content (what if somebody has even wider aspect ratio display). Also the font is a bit bigger than I used to see in other places. Hence, I think it's unnecessary to add blank space on the left and right side of the case picture. And maybe it's a good idea to limit the maximal width of the main panel. On the details page the images are big but low resolution, which doesn't look professional.
-
I would opt for improvements here. Particularly, spacing needs to be improved: the paddings/margins are very often too small.
increase the icon for the pilots if possible
Clean and good-looking. I like the simple, bright and nice design a lot. In the future, I think it would be cool to have a dark mode as well, and if you like to be extra nice, having accessibility features like high contrast mode, colours for colourblind people and similar features that allow people with disabilities to use the dashboard more easily.
I really like the idea of being able to switch between matrix vs list layouts.
Cosmetics: maybe we need to change the block orange shadow into blue/dark blue?

Please rate the **functionalities** of the dashboard. Functionalities include the ability to search and filter tiles, the support for a detail page, tags and sorting. Later functionalities will include user roles and editing. Who have access to editing, please also include its rating here.
5 - perfect
3 - OK
3 - OK
4 - good
5 - perfect
4 - good
2 - acceptable

3 - OK
4 - good
4 - good
4 - good
4 - good
4 - good

Please share your thoughts, opinions and suggestions for **functionalities**.
Ok for me at the moment
Difficult to assess when not all the functionalities are working, particularly the access to the modelling section of every pilots, what makes the dashboard very useful.
There still are so much not available functionality as pilots are yet in the process to integrate their functionality, so it is early to assess.
Support is good through the dashboard editor, and the filtering etc. However, as I mention above, the filtering panel maybe takes up more space than necessary, or is maybe not fitting with the overall design.
Very convenient to use
1- not clear the difference between support services and complimentary services 2- tags are mixing up HiDALGO organizations (i.e. partners) and services (e.g. MathSO), which is weird for tags. Maybe MathSO is not required as SZE is included and we can rename tags as Service Providers. Indeed, tags and organizations contains pretty much the same entries. An option is that tags refers to keywords describing common functionality among services: e.g. Workflow Orchestrators, Data Visualization, Training, Documentation, Code Management, etc 3- The link to QCG is broken 4- The link to the UAP pilot goes to the main page of MathSO, but no UAP pilot content is provided.
However, when I open the "Details", I'm lost because: - I can't go back in my browser, - it's not a new tab, - when I click the logo nothing happens,

- when I click the "Dashboard" button also nothing happens.
I had to reopen the page. After a longer while I realized that it is a panel with "X" for closing.
Hence:
- the panel looks like the whole page content, please do something with it (maybe animation and smaller panel size?),
- HiDALGO2 logo should open the main dashboard page,
- clicking the "Dashboard" maybe should close the details panel, but I'm not sure.

Another issue - when I type "RES" in the search, it doesn't show Renewable Energy Sources, despite the abbreviation is used in the description. It shows Wildfires instead.

When I click "Go to" for the pilot, it takes me out of the dashboard, to the main project site, while I'm still thinking it is the dashboard (realized after a while). Maybe you should add the "external link" icon in the "Go to" button?

My rating connects with the fact that I do not remember exactly the goal that the dashboard represents. I can give you more feedback when I know.

In general the proposition is fine. I am not sure how the filtering based on institution will work and what is the purpose of this filtering.

for this first version is fine

I'm missing some explanations, I guess they will be added to `User Guide` once it is available? E.g.,

* HPC Configuration: Explanation of purpose and usage. E.g. create a new SSH key, don't reuse a private key you use on your other systems. There is a `Password` field, I assume the one that is used for password connection instead of SSH, and then there is a `Passphrase (will not be saved)`, I assume that is the encryption password for the ssh key as it is only queried if `SSH connection check (recommended)` is checked. It makes sense, but it is not clear at first glance. So verbosity of this page and it fields could be improved. Also, a short description what this is for would be ideal, i.e., "HPC system credentials to allow the portal to access your user profile on the host cluster". What is the tick `Save SSH credentials to database` for? Which database? What is HPC configuration? I can write anything there, and it looks like I'm not supposed to write anything in there, but as if the content is provided automatically after submitting my credentials. I suppose it requires the content I would usually put into my ssh-config file? Furthermore, I don't like that the private key needs to be copy-pasted there and is visible in clear text. Ideally, the portal is able to create a private key for me, allows me to export a public key after generation,

and allows me to import a private key by uploading a file. (Allowing to past it there is also fine, as most users would probably prefer this way.) BUG: No matter what I do, I always get the error `HPC configuration already exists with the same name!`, even though I did not register anything yet. * General, I'm missing a data safety disclaimer. I.e., Where is the data stored? Who has access to it? How long is it stored? ETC. To use the functionality of the user portal, I need to store sensitive data like passwords, SSH-keys, accounts for billing, username, etc. I as a user would refrain from using the portal if I can't be sure that my data, and especially project related data, is stored and processed securely. * I could not test the remaining features of the Portal, as I don't have anything to insert there at the moment. But it all seems reasonable. * The dashboard features are reasonable and self-explanatory. Very nice and good-looking. :-)
Filtering is also nice beside search. Filtering is also very responsive. Nice work! If search could be made to work including behind-the-link contents (as described above), filter might also be possible with this extended content. It seems like this would be very useful to make navigation faster.
For the moment everything looks OK. Let see, if there will be some new usage scenarios which should be addressed by the functionalities update/upgrade.

Please rate the **usability and responsiveness** of the dashboard. Usability means, that its functionalities efficiently support daily work and activities. Responsiveness refers to the speed of the portal.
5 - perfect
4 - good
4 - good
5 - perfect
5 - perfect
4 - good
5 - perfect
4 - good

3 - OK
4 - good
5 - perfect
5 - perfect
4 - good

Please share your thoughts, opinions and suggestions for **usability and responsiveness**.
the system seems quite responsive;
Still we need to know how responsive are the modelling capabilities of the dashboard
It is early to assess it. The usability and responsiveness of the elements already available it is quite good.
Dashboard is fast, responsive and has good usability. All the HiDALGO2 services are collected. Maybe in the next iteration, there can be some more features added that link some information, or summaries, from the services back to the dashboard.
Very fast responded.
1- Slightly delay when rendering the service icons after changing the filters or when coming back from a service details page
I like the search panel dynamics, it's very fast. The tiled look is also a good idea. The page runs smoothly.
there's now much to do in the use cases section. Maybe need to connect it with the codes?
I think that basic functionality is in place. Maybe for later future I would consider integration of some AI to help in selection of a proper solution.
for this first version is fine
Really great work. It looks good it does what it is supposed to do, and it is self-explanatory.
Very fast! Excellent job!

So far so good

Please indicate, if there are any additional **services to be added** to the dashboard. Please refer to the TOC of D2.8 for the services that are planned, but not yet added. These will be added, so you do not need to mention them here.
We will add Ktirio GUI (WASM) as discussed in several discussions already. this is an input data generator for Ktirio Urban Building
To me the services provided are OK, but we need to assess the modelling capabilities as well
I don't miss any of them today
hdoc.hidalgo2.eu and git.hidalgo2.eu are two services I can think of right now. Later we would also need to add Ktirio GUI, or we go from there through the UB pilot service.
There are no additional services wished to be added from my side
To be discuss in the context of D2.8, to what concerns data management.
-
-
...
Can't think of any.
I like the easy access to source code repositories. Perhaps a more detailed list of the codes available (with their web site) would be also useful, instead of having to click on the Bitbucket.
I do not know.

Please indicate, if there are any additional **services to be removed from** the dashboard.

See my remark on the pilot, I changed the name of the pilot to reflect what we write in deliverables and communications. We use now Ktirio Urban Building (KUB) for the pilot instead of UBM. UBM should now be removed. Thanks
None of the existing ones
I do not think so
I believe we will not need Bitbucket, but add the git.hidalgo2.eu Gitlab instead (which has Single Sign On with Keycloak). We also don't need Zabbix/Monitoring to be accessible from Dashboard, this is an internal service at PSNC. I am also not sure about OpenProject. Or it can be there but only for the internal HiDALGO2 users. Maybe the same can be for Zabbix, only for internal users.
Looks all good for me
I am afraid OpenProject is not used by most of us, excepting USTUTT. I am fine with the others, indeed the dashboard is an excellent hub for me to get access to the HiDALGO services I am using
-
-
Ktirio Urban Building is not really a pilot! Must be placed in a different category...not sure which one thou
Can't think of any.
More information is better in my opinion - since fast filtering and search makes it easy to navigate already.
Do we want to have a Wiki on the service dashboard? Do we consider documentation as a service?

Annex II. List of requirements

#ID	Title	Description	Required by	Release date
DASH-REQ-001	SZE	Access all pilot	Access All pilot using only dashboard, and auto-link to the proper WFO portal	Delayed for first Iteration release (2025.06.01)
DASH-REQ-002	SZE	Access CKAN directly from dashboard	After login, users will be able to create datasets and upload data to CKAN	Done in Initial release (2024.11.30)
DASH-REQ-003	SZE	SSO auth	Using dashboard as a single entry point for HiDALGO2 infrastructure	Done in First Iteration release (2025.06.01)
DASH-REQ-004	SZE	user levels in Dashboard infra	Different users with different roles can access different services (example: admin users can access to Keycloak to create new user)	Planned for Final release (2026.01.01)

#ID	Title	Description	Required by	Release date
DASH-REQ-005	ATOS	Access from dashboard to HIDALGO2 DMS frontends	Multiple DMS frontends will be provided, each on its own frontend service (different URLs), This list of DMS services includes services for end-users (not exhaustive lists, could be increased in future): - CKAN (already posted in above requirement) - HUE frontend (Hadoop frontend for users) - Atlas (Catalog) - to be confirmed. - Apache Knox (Gateway) - to be confirmed Other DMS services, intended for administration are not required to be integrated within the frontend, they can be accessed by administrators in their own URL: - Hadoop UI - NIFI	Partially delayed for First Iteration release (2025.06.01)
DASH-REQ-006	UNISTRA	access to CKAN	-interaction with data management platform ;	Done in Initial release (2024.11.30)
DASH-REQ-007	UNISTRA	data preparation frontend	run Ktirio-GUI in wasm mode in the dashboard, to prepare the dataset before upload to CKAN	Planned for First Iteration release (2025.06.01)
DASH-REQ-008	UNISTRA	access documentation website of Ktirio urban building (KUB)	the KUB pilot has a documentation explaining how to configure and run it, the documentation is hosted on a website than should be accessible	Planned for First Iteration release (2025.06.01)
DASH-REQ-009	UNISTRA	Monitor pilot health	use reframe json files to produce a dashboard of the health of the pilot over time and over the different machines used; this should be generated automatically e.g. through CI/CD or portal	Planned for First Iteration release (2025.06.01)

#ID	Title	Description	Required by	Release date
DASH-REQ-010	UNISTRA	Visualisation	Enable visualisation through the dashboard either 3D interactive but also through Jupiter files and automated report generation	Delayed for first Iteration release (2025.06.01)
DASH-REQ-011	UNISTRA	Embed seamlessly webpages associated to KUB	KUB provides webpages using asciidoc that are rendered into websites. it would be super to be able to integrate seamlessly these pages into the dashboard	Delayed for first Iteration release (2025.06.01)
DASH-REQ-012	PSNC (RES)	access to CKAN	interaction with data management platform, including historic data provided by other pilots for reanalysis and/or using this as initial input data for RES. Browse historic data of RES runs.	Done in Initial release (2024.11.30)
DASH-REQ-013	PSNC (RES)	QCG	interaction with QCG for ensemble runs and uncertainty quantification	Done in Initial release (2024.11.30)
DASH-REQ-014	PSNC (RES)	ticketing system	access to ticketing system to solve issues users encounter	Done in Initial release (2024.11.30)
DASH-REQ-015	PSNC (RES)	visualisation	visualisation of RES results using in-house scripts and/or the Hidalgo2 vis tools	Delayed for first Iteration release (2025.06.01)
DASH-REQ-016	PSNC (RES)	monitoring	monitoring of pending jobs, status of hidalgo2 tools and status of EuroHPC sites	Planned for First Iteration release (2025.06.01)
DASH-REQ-017	MTG (WF) Landscape	Boundary conditions download	Select model for boundary conditions / Select date / Select regional area -> DMS (Hadoop?) Combustibles y topo	Planned for Final release (2026.01.01)
DASH-REQ-018	MTG (WF) Landscape	Simulation environment definition	Namelist configuration / submit script	Planned for First Iteration release (2025.06.01)