opertus mundi

Final OpertusMundi Platform

Report on Deliverable D1.5

Horizon 2020
Innovation Action
2020-2022

PROJECT NUMBER: 870228
START DATE OF PROJECT: 01/01/2020
DURATION: 36 months

OpertusMundi is an Innovation project funded by European Commission's Horizon 2020 programme.

Dissemination Level	Public
Due Date of Deliverable	M36, 31/12/2022
Actual Submission Date	28/12/2022
Work Package	WP1 – Requirements, Architecture, and Integration
Tasks	Task 1.3 Integration
Type	Other
Approval Status	Submitted for approval
Version	1.0
Number of Pages	119
Filename	D1.5_Final_OpertusMundi_Platform_v1.0.pdf

Abstract

This report presents an overview of Deliverable D1.5 "Final OpertusMundi Platform", i.e., the final integrated version of the OpertusMundi platform (**topio.market**) with integrated improvements derived from our pilot. The platform has been initialized, deployed, and is under development and integration testing following the methodologies presented in Deliverable D1.2 (Architecture). This report presents an overview of its current operational status. We first present an update of the platform's architecture (logical, physical) and its deployment over our cloud infrastructure. Next, we present a walkthrough of the services and functionalities provided to its various users, as well as examples of select core workflows, demonstrating the maturity of the platform.

History

version	date	reason	revised by
0.1	01/09/2022	First version (abstract, executive summary, sections)	Spiros Athanasiou
0.2	18/09/2022	Updated Architecture (logical, deployment)	Yannis Kouvaras, Angelos Tzotsos, Michail Alexakis
0.3	20/09/2022	Multiple updates across all sections	Konstantinos Triantos, Alexandra Alexandridou, Kyriakos Psarakis, Thodoris Vakkas
0.4	02/10/2022	Multiple revisions in descriptions (text, screenshots)	Kostas Patroumpas, Leonidas Oikonomou, Alexandra Alexandridou
0.5	08/10/2022	Revised and extended descriptions and workflows	Kostas Patroumpas, Giorgos Chatzigeorgakidis
0.6	18/10/2022	Updated deployment documentation, minor edits in Annexes	Michail Alexakis, Konstantinos Triantos
0.7	22/10/2022	Updated descriptions and workflows	Giorgos Chatzigeorgakidis, Yannis Kouvaras
0.8	25/11/2022	Minor edits and revisions across the entire document	Yannis Kouvaras, Michail Alexakis, Alexandra Alexandridou
0.9	20/12/2022	Internal Review	Dimitris Skoutas, Kostas Patroumpas
1.0	28/12/2022	Final version	Spiros Athanasiou

Author list

organization	name	contact information
ATHENA RC	Spiros Athanasiou	spathan@athenarc.gr
ATHENA RC	Yannis Kouvaras	jkouvar@athenarc.gr
ATHENA RC	Kostas Patroumpas	kpatro@athenarc.gr
ATHENA RC	Dimitris Skoutas	dskoutas@athenarc.gr
ATHENA RC	Angelos Tzotsos	tzotsos@athenarc.gr
ATHENA RC	Giorgos Chatzigeorgakidis	gchatzi@athenarc.gr
ATHENA RC	Michail Alexakis	alexakis@athenarc.gr
ATHENA RC	Nikiforos Leonidakis	nikleonidakis@athenarc.gr
ATHENA RC	Vasilios Georgopoulos	vgeorgopoulos@athenarc.gr
TU Delft	Asterios Katsifodimos	a.katsifodimos@tudelft.nl
TU Delft	Andra Ionescu	a.ionescu-3@tudelft.nl
TU Delft	Kyriakos Psarakis	k.psarakis@student.tudelft.nl
GET	Thodoris Vakkas	tvakkas@getmap.gr
GET	Konstantinos Triantos	ktriantos@getmap.gr
ROLEPLAY	Leonidas Oikonomou	leonidas@roleplay.gr
ROLEPLAY	Alexandra Alexandridou	alexandra@roleplay.gr
ROLEPLAY	Maria Konomi	maria@roleplay.gr

Executive Summary

This report presents an overview of Deliverable D1.5 "Final OpertusMundi marketplace", i.e., the final integrated version of the OpertusMundi platform (**topio.market**) with integrated improvements derived from our pilot. The platform has been initialized, deployed, and is under development and integration testing following the methodologies presented in Deliverable D1.2 (Architecture). This report presents an overview of its current operational status. In this report, the OpertusMundi marketplace shall also be referenced *interchangeably* under its commercial name **'Topio'** (*pronunciation: /ˈto.pi.o/*).

In Section 1, we present the final OpertusMundi marketplace. First, we provide an update of its logical architecture and detail its major components. Next, we present the production system applied to deploy the marketplace, briefly presenting its capabilities and characteristics, as well as its initialization. Finally, we present how documentation is produced and provide links to the actual documentation of the system.

In Section 2, we present a walkthrough of the services and functionalities provided by the platform to its various users, as well as examples of select core workflows, demonstrating the maturity of the platform at this stage of its development.

Are you interested in learning more and joining Topio? Please read this!

This report has been prepared in the context of the EC-funded project 'OpertusMundi', and as such, its structure, content, and presentation format primarily serves the evaluation of our work from the EC.

If you are interested to **learn more** about the Topio marketplace (see topio.market), want to **trade, buy or use geospatial data**, please get in touch with us directly. We will be happy to provide you with guided **walkthroughs** of Topio, understand your **data** and services, **prepare** all your assets and publish them in Topio.

- **Reach out**. You can get in touch with us via email at hello@topio.market for any question you may have, to introduce yourselves, or to arrange a demonstration.
- **Topio Pitch**. A short and jargon-free presentation of Topio is available here[1]. You can find more information about the challenge of fair geospatial data trading, the importance of geospatial data in EU's Data Economy, the business model and services of Topio, as well details on how you can (a) publish and trade assets, (b) discover and purchase assets, (c) harness additional value from your data, and (d) build new applications and services with Topio's services.

[1] Direct access link: https://www.opertusmundi.eu/wp-content/uploads/2021/03/Topio-pitch-2021-v1.1.pdf

- **Topio Video**. Check out this shinny new video we prepared giving you an overview of what Topio is and how you can trade your data here.

Abbreviations and Acronyms

API	Application Programming Interface
AWS	Amazon Web Services
BPM	Business Process Management
BPMN	Business Process Model and Notation
CRUD	Create, Read, Update, and Delete
CSV	Comma Separated Values
DAG	Directed Acyclic Graph
EO	Earth Observation
GIS	Geospatial Information System
IDP	IDentity Provider
ISO	International Organization for Standardization
JSON	JavaScript Object Notation
MVC	Model View Control
OAS	OpenAPI Specification
OGC	Open Geospatial Consortium
PKI	Public Key Infrastructure
REST	REpresentational State Transfer
UI/UX	User Interface/User eXperience
URL	Uniform Resource Locator
UUID	Universally Unique Identifier
VAS	Value-Added Service
VM	Virtual Machine
WMS	Web Map Service
XML	eXtensible Markup Language

Table of Contents

1. System Architecture

In this section, we present the final OpertusMundi platform. First, we present its logical architecture and detail its major components. Next, we present the private cloud infrastructure applied to deploy the platform, briefly present its capabilities, characteristics, and initialization, and elaborate on its physical deployment. Finally, we present how documentation is produced and provide links to the actual documentation of the platform.

1.1. Architecture

In section we present the logical architecture of the final OpertusMundi platform. First, we present a high-level overview of the architecture where the main building blocks of the platform are organized and displayed in layers (Figure 1). Then, we expand our presentation, displaying how the components identified in Section 1.1.1, are instantiated as applications and microservices in our logical architecture (Figure 2). Finally, we present the physical architecture of the platform where more details are provided for both the implementation and the deployment of the OpertusMundi software components (Figure 3).

1.1.1. Building blocks

Overall, the OpertusMundi architecture is based on the concept of **microservices**. Although implementing a system using microservices introduces additional overhead for common features such as transaction management, deployment, fault tolerance and scalability, it also handles complexity more efficiently by decomposing the application into more manageable services. Moreover, it allows different development teams to focus on specific services and work independently, using the best tools suited for the service under development.

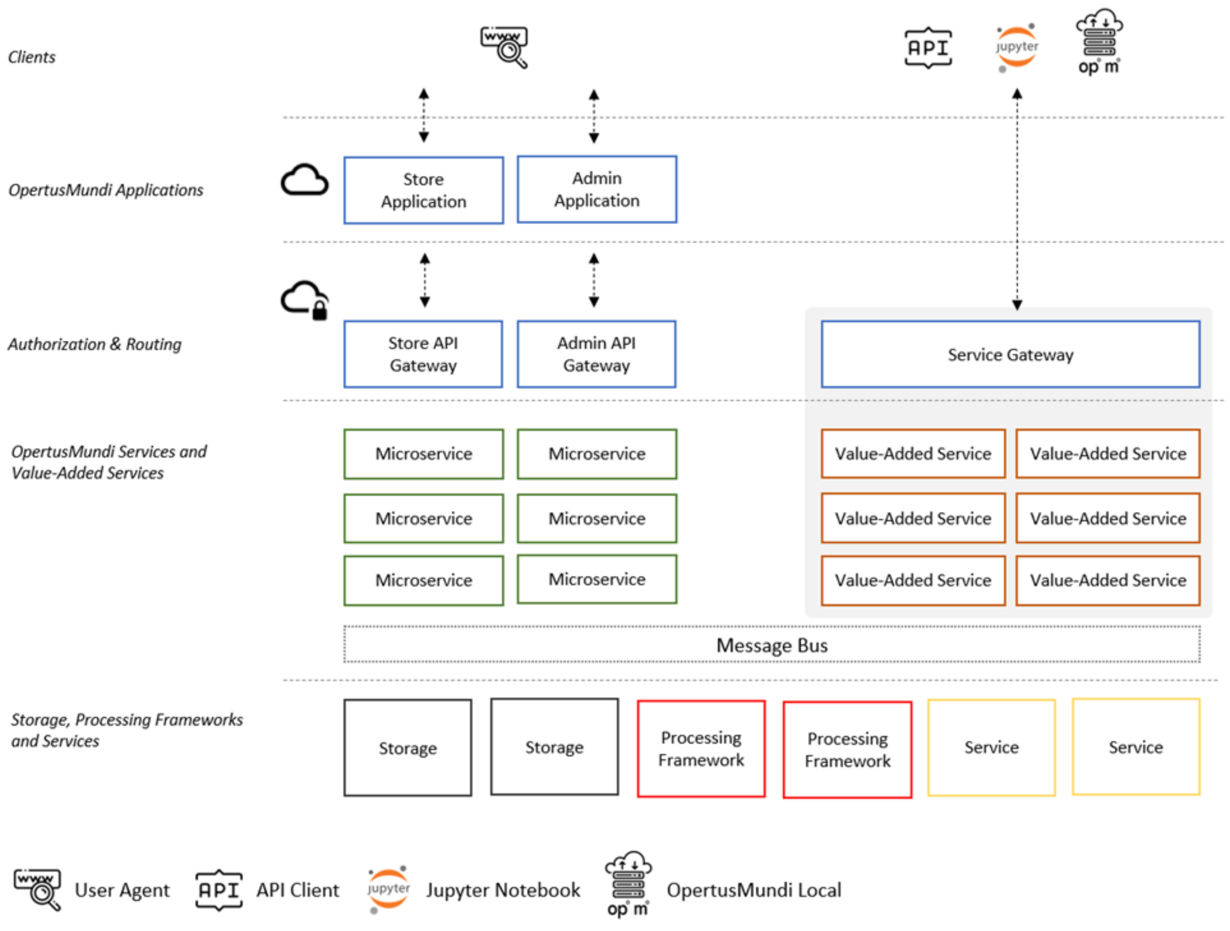


Figure 1: Logical artchitecture overview

Next, we give a short description of each layer:

- **Clients**: OpertusMundi supports four types of clients: (a) user agents (browsers) for accessing the marketplace and admin applications, (b) API clients for consuming platform services, (c) Jupyter Notebooks that interact with platform services programmatically through libraries that wrap OpertusMundi APIs and (d) the OpertusMundi Local application.
- **Applications**: Two separate web applications are being developed for OpertusMundi, namely, the Store and Admin applications. The former implements the User Interface (UI) for the store functionality required for asset providers and consumers while the latter implements the UI for administrative tasks, application monitoring, customer management and billing. Both web applications are exchanging data with the OpertusMundi platform using two separate API gateways.
- **Authorization & Routing**: This layer contains the gateways for the APIs consumed by the web applications and the OpertusMundi value-added services (VAS). For a web application, the API gateway composes responses by invoking several microservices, enforce security rules and shape data according to the requirements of the web applications UI. For a VAS, the service gateway validates application keys, routes traffic to the appropriate services and computes statistics for auditing, accounting, and billing purposes.

- **Services and Value-Added Services**: As mentioned above, OpertusMundi is implemented using a microservice architecture. Services are separated in two distinct categories, namely, application-specific microservices and value-added services. The former implements the core business logic of the OpertusMundi platform and are invoked by the web application API gateways. The latter implement any tasks required for asset management and processing, either it is a long running one (e.g., publishing a WMS endpoint) or executing a data processing operation (e.g., transforming a dataset). Value-added services are configured and managed by one or more microservices as explained later. Services communicate by exchanging messages either through API calls or (optionally) a message bus service.
- **Storage, Processing Frameworks and Services**: The last layer of the architecture contains any existing software systems we are going to deploy and configure for supporting the OpertusMundi platform. These include storage systems (i.e., a relational database or a distributed file system), processing frameworks (e.g., a distributed computing framework) and task-focused services, such as a Public Key Infrastructure (PKI) or a WMS service.

1.1.2. Applications and microservices

After having identified the main building blocks of the OpertusMundi platform, we expand several blocks to display how components, are instantiated as applications and microservices in our logical architecture (Figure 2).

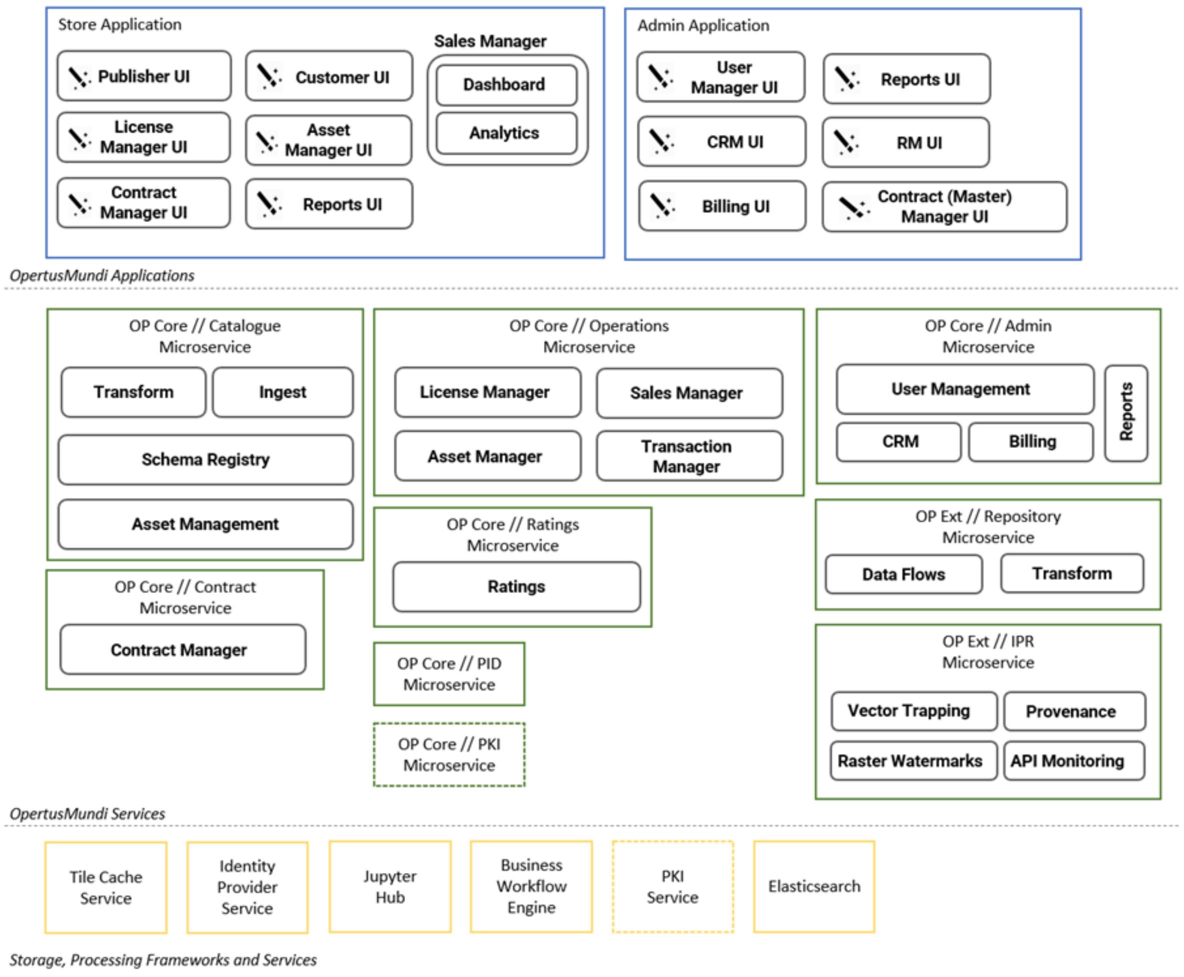


Figure 2: Logical architecture building blocks and supported features

Next, we give a short description of each component:

- **Store Application**: The store application implements all the UI components required for managing asset trading for both consumers and providers. Depending on the authenticated user roles, the appropriate components are rendered. Complex tasks are implemented as wizards that collect user input and result in API method calls. These calls either update one or more database instances at the server-side or interact with more complex business workflows executing at the server-side.
- **Admin Application**: All administrative tasks are handled by the admin application. This application implements the UI required for user management, accounting and billing, customer issue management, master contract editing, and service monitoring and configuration. As in the store application, complex tasks, such as provider reviewing or master contract editing, are implemented using wizards that drive the execution of business workflows at the server-side by invoking appropriate API methods.
- **OP Core // Catalogue Microservice**: The catalogue service implements the core functionality of the OpertusMundi platform, and it is the entry point for assets to the platform. All other services depend on this service for accessing asset metadata and as such, the service participates in almost all business workflows implemented by the system.

- **OP Core // Ratings Microservice**: In the OpertusMundi store application, consumers are able to rate providers and assets they have purchased. This service could be integrated in the catalogue, but we opted to design it as a standalone service to isolate workflows for rating moderation and message exchange between consumers and providers from the catalogue service.
- **OP Core // Operations Microservice**: All interactions between external users and the OpertusMundi platform are handled by the operations microservice. This service is responsible for tracking all actions related to assets' information lifecycle, including asset licensing, pricing, sales, and provisioning. The Operations microservice is the entry point for any business workflow involving consumers or providers, implemented by the Business Workflow Engine described later.
- **OP Core // Admin Microservice**: The admin service implements administrative tasks such as user management, resource management and reporting. In contrast to other services, it is aware of the security context of requests and is accessible *only* by accredited OpertusMundi personnel (i.e., platform administrators).
- **OP Core // Contract Microservice**: The contract service is responsible for managing all types of contracts supported by the OpertusMundi platform. This includes the master contract(s) used by all providers, template contracts created by providers and assigned to their assets, as well as purchase contracts instantiated whenever a transaction in the marketplace takes place. The service maintains the complete lineage of master and template contracts and their various states, supports electronic contract signing and exchange, as well as the rendering of contracts (view in forms and pdf format for export).
- **OP Core // Recommender Microservice**: The recommender service provides asset recommendations for users using data from several sources including, the catalogue, operations, and admin services. Each of these services provides appropriate query endpoints for fetching any required data. The recommender service stores and aggregates this data in a separate database.
- **OP Core // PID Microservice**: To support the lineage tracking of assets, the catalogue service assigns unique identifiers to every asset version. Although the catalogue could generate Universally Unique Identifiers (UUID), we have created a separate service for generating and maintaining persisted unique identifiers. This service will allow linking and dereferencing of OpertusMundi identifiers with custom identifier generated by asset owners.
- **OP Ext // Repository Microservice**: The repository service is responsible for storing asset data available directly to the OpertusMundi platform. Asset metadata are available from the catalogue service while ingestion is invoked by the asset publishing workflow.
- **OP Ext // IPR Microservice**: The IPR microservice consists of a set of services for protecting the intellectual property of data asset owners. Each service handles a specific asset type (e.g., for raster data, the service injects a digital watermark to images). The service is invoked either by the repository service during the ingestion of data assets or by VAS instances during the provisioning of assets.

- **Jupyter Hub**: The OpertusMundi platform enables users to perform complex data processing, analysis, and visualization on geospatial assets they own using Jupyter Notebooks. This value-added service is implemented by deploying and configuring the JupyterHub Service[2].
- **Identity Provider Service**: The identity provider (IDP) service implements single sign-on (SSO) for all OpertusMundi components that require authentication using OAuth2 or OpenID Connect protocols. In OpertusMundi, the IDP is used only for authentication. The authorization is controlled by the Admin service. Moreover, the IDP supports multitenancy for separating consumer and provider users from OpertusMundi personnel (e.g., Helpdesk users, legal experts, admins).
- **Business Workflow Engine**: Several features of the OpertusMundi system involve the execution of complex, optionally long-running business workflows that require user interaction such as provider verification or order fulfillment and payment transfer. These workflows are implemented by a Business Process Management (BPM) service. This service orchestrates several microservices to implement the task at hand and provides detailed audit information for all the steps of every workflow execution instance.
- **PKI Service**: A Public Key Infrastructure (PKI) service for generating public keys and digital certificates for consumers and providers, used for signing contracts and securely accessing various platform services.

1.1.3. Physical architecture

Finally, Figure 3 illustrates the initial physical architecture of the OpertusMundi platform where more details are provided for both the implementation and the deployment of the OpertusMundi software components.

[2] https://jupyterhub.readthedocs.io/en/stable/

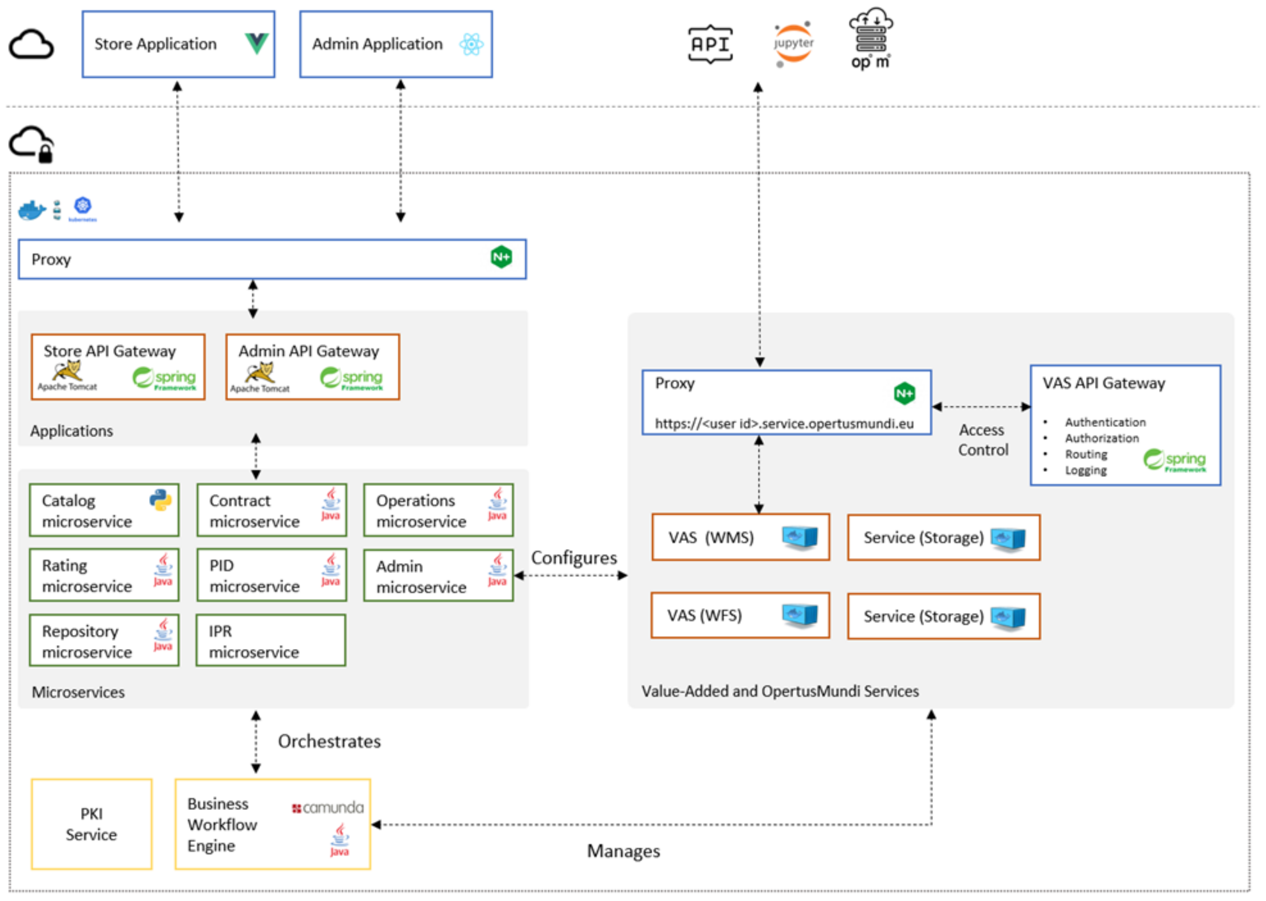


Figure 3: OpertusMundi physical architecture

Next, we provide a short description of each component:

- **Store & Admin applications**: The Store and Admin web applications are implemented using the Vue.js[3] and React[4] JavaScript frameworks, respectively. The applications can be hosted either separately using the Node.js[5] JavaScript runtime or packaged as part of the Store and Admin API Gateway services. Since the web applications are tightly coupled to the API Gateway implementation due to the API versioning, we opted for the second solution. Packaging is handled automatically during the build phase. The source of the web applications is available at the project GitHub repository[6,7].
- **Proxy Server**: API Gateway services are published behind a proxy service implemented using the NGINX Plus[8] proxy. Using a well-established proxy server, allows for simpler configuration, offers increased security, and provides a richer set of configuration options e.g., implementing routing rules through configuration.

[3] https://vuejs.org/
[4] https://reactjs.org/
[5] https://nodejs.org/en/
[6] https://github.com/OpertusMundi/frontend-marketplace
[7] https://github.com/OpertusMundi/frontend-admin
[8] https://docs.nginx.com/nginx/

- **Application Server & API Gateways**: The API Gateway services are implemented using the Spring Boot[9] framework and can be hosted either as standalone applications with an embedded Apache Tomcat[10] server or inside a separate Tomcat installation. During the first stages of development, we use the first option. The applications are being developed using the Spring Web MVC[11] web stack. Since Spring MVC does not provide a non-blocking implementation and API Gateways are the main entry point for all web application requests, we may later switch to Spring WebFlux[12] to further increase scaling and robustness. The source of the API Gateway services is available at the project GitHub repository[13,14].
- **Microservices**: OpertusMundi services are being implemented using either the Java or Python programming languages. All services expose their functionality using HTTP APIs and exchange messages in JSON[15] format. The declared APIs are publishing their documentation as OpenAPI documents. In addition to the OpenAPI documents, most services provide the appropriate UI to visualize and interact with the provided API.

  Due to the Microservices Architecture, the platform does not support transactions across multiple databases. Therefore, during the execution of business workflows that involve the invocation of several service methods, error handling is implemented using compensation and retry operations. Hence, services implement idempotent operations and provide compensation methods wherever required. Idempotent methods guarantee predictable results for retry operations. A compensation operation may be as simple as a CRUD operation or involve more complex processing. In either case, in the context of the service, a compensation operation will be transactional.
- **Business Workflow Engine**: Many OpertusMundi features require the interaction of several users and services, forming complex workflows. For instance, the order, payment, and physical delivery of an asset is a long running process that requires input from the consumer, provider, and OpertusMundi personnel, as well as the invocation of multiple service methods. Moreover, depending on the business logic, time-based events may cause additional method invocations (e.g., sending a notification if asset delivery is not fulfilled within a given timeframe).

  We could have implemented such workflows manually and control method invocation flow using a publish/subscribe event bus. Nevertheless, such a solution is hard to maintain as workflow complexity increases and we would also have to implement tracking manually by aggregating data from multiple sources. Instead, we are using a Business Process Management (BPM) service for handling workflows, and specifically the Camunda[16] BPM platform. In OpertusMundi, Camunda is used as a microservice orchestration service that provides a single point of reference for querying the state of any workflow execution

[9] https://spring.io/projects/spring-boot
[10] http://tomcat.apache.org/
[11] https://docs.spring.io/spring/docs/current/spring-framework-reference/web.html#spring-web
[12] https://docs.spring.io/spring/docs/current/spring-framework-reference/web-reactive.html#spring-webflux
[13] https://github.com/OpertusMundi/api-gateway-service
[14] https://github.com/OpertusMundi/admin-gateway-service
[15] https://www.json.org/json-en.html
[16] https://camunda.com/

instance and communicates with other services by exchanging JSON formatted messages over HTTP. Using Camunda, we can easily view the lineage of a workflow execution and arguments, results, and errors of service method calls.

Camunda can be deployed in several ways with the most common ones being (a) deployed as a web application inside a Java Servlet container, (b) packaged as a Docker Image or (c) integrated in a Spring Boot application. We have selected the latter option. Moreover, to decouple the BPM engine from other OpertusMundi services, a separate service, namely the BPM worker, is used for executing external tasks, i.e., any task that executes custom business logic or invokes other services. Multiple worker instances are deployed and Camunda propagates any external tasks to them. Hence, Camunda is focused only on workflow orchestration and error handling.

- **Value-Added Services**: Value-added services comprise a series of applications or micro-services externalized towards platform users (i.e., data consumers, data owners, users) as commercial services accessible via a trading agreement with the marketplace. A VAS instance may be as simple as configuring a custom WMS service endpoint based on an asset uploaded by a user, to performing complex automated geospatial data integration jobs. VAS instances are initialized by BPM workflows and are invoked: (a) directly by platform users e.g., a WMS service, (b) using programmatic APIs e.g., importing a library in a Jupyter Notebook, (c) as part of a business workflow, or (d) by time events generated by a scheduler.
- **VAS API Gateway**: VAS instances have several functional requirements such as authentication, authorization, logging, and routing. We implemented these features inside a separate service, namely VAS API Gateway, which is integrated with NGINX Plus[17] proxy. Each time an API request reaches the proxy server, a second request is sent to the VAS API Gateway and depending on the HTTP Status Code[18] of the response, the request is accepted or rejected. Using a separate service allows the implementation of various authentication and authorization schemes.

1.2. Deployment

In this section, we present the production system applied to deploy the OpertusMundi platform, briefly presenting its capabilities and characteristics, as well as its initialization. Then, we present the actual production architecture in terms of Kubernetes workload[19] objects, how software is packaged, and how deployment is automated. The current versions for all major software components comprising the OpertusMundi platform are provided in Annex 3.2.

[17] https://docs.nginx.com/nginx/admin-guide/security-controls/configuring-subrequest-authentication/

[18] https://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

[19] https://kubernetes.io/docs/concepts/workloads/

1.2.1. Production Architecture

The OpertusMundi platform is deployed within the Hellenic Data Service[20] (HELIX), the national cloud infrastructure for data intensive research & innovation co-developed by Athena RC, harnessing its highly scalable and compute, network, and storage resources. OpertusMundi is one of the first commercial services hosted by HELIX for Industrial Data Platforms and Data Spaces, ensuring its sustainable operation and implementation of our exploitation plans.

On a system level, OpertusMundi is deployed on top of the Synnefo cloud stack[21], within several virtual machines. Synnefo is a complete open source cloud stack written in Python that provides Compute, Network, Image, Volume and Storage services, like those offered by AWS[22]. On a hardware level, the production system is hosted on Athena RC's private IaaS available from okeanos-knossos.gr, currently allocating *256 CPU cores, 256 GB main memory, 20TB storage.*

On top of the virtual machines provided by our IaaS infrastructure, a Kubernetes cluster is formed. The cluster has a single master node (the so-called control plane) and a dozen of worker nodes (*each one typically holding 16 CPU cores, 32GiB memory, and ~ 120GiB of fast node-local storage*). All cluster nodes (each one is a virtual machine) communicate on a private IP network and are not (directly) exposed to the outer world.

Inside a Kubernetes cluster, the smallest unit of execution which can be scheduled (to run on a node) is the Pod[23]. A Pod may need to communicate with other Pods, so it must be addressable and may participate in a cluster-wide network of Pods. A virtual IP network of Pods is formed on top of the network of nodes (hosting those Pods). Kubernetes does not directly implement the networking stack for Pod-to-Pod communication; instead, it relies on a Container-Network-Interface (CNI[24]) network plugin. In our platform, we have chosen the Calico[25] network plugin as a mature and well-tested open-source solution. Calico may not be the easiest (to set up) networking solution, but offers some significant advantages compared to other CNI plugins: (a) is performant and trace-friendly as it routes[26] IP packets directly (i.e., no packet encapsulation) between nodes, and (b) supports traffic filtering based on a defined Network Policy[27] at the cluster level.

Almost every component of the OpertusMundi platform is represented as a Kubernetes workload resource inside the cluster:

- as a Deployment[28], if the application is characterized as a set of stateless services (stateless in the sense that new members can be added to the set and each member can be safely rescheduled to another node of the cluster).

[20] http://www.hellenicdataservice.gr
[21] https://www.synnefo.org/
[22] https://aws.amazon.com/
[23] https://kubernetes.io/docs/concepts/workloads/pods/
[24] https://github.com/containernetworking/cni#cni---the-container-network-interface
[25] https://www.projectcalico.org/calico-networking-for-kubernetes/
[26] Calico routes using BGP routing protocol (https://en.wikipedia.org/wiki/Border_Gateway_Protocol)
[27] https://kubernetes.io/docs/concepts/services-networking/network-policies/
[28] https://kubernetes.io/docs/concepts/workloads/controllers/deployment/

- as a StatefulSet[29], if the application is characterized as a set of stateful services (stateful in the sense that each member of the set must stick to certain storage resources for the lifetime of the application) with each member holding a stable network cluster-local address.
- as a Job[30], if the application is characterized as a one-off batch processing task.
- as a CronJob[31], if the application is characterized as (possibly retry-able) periodic task running on a given time schedule. In fact, a CronJob just instantiates Jobs at certain time points.

An overview of the OpertusMundi deployment scheme is presented in Figure 4 and comprises:

- A single **administration server** (outside of the Kubernetes cluster) that acts:
    - as a gateway for accessing the deployment's private network and administrating VM hosts remotely using SSH;
    - as a control machine for running maintenance/configuration tasks against the entire Kubernetes cluster (e.g., running an Ansible playbook that prepares a new node for joining the cluster);
    - as a gateway for accessing the Kubernetes cluster via command-line interface (kubectl, kubeadm), or via HTTP REST interface (Kubernetes API);
    - as a proxy for Kubernetes dashboard web UI;
    - as a proxy to several administrative/monitoring web UIs exposed from internal services.
- A **group of NFS servers** that provide distributed filesystem storage to applications running in the Kubernetes cluster. Because NFS service is closely coupled with the Linux kernel (NFS is not a FUSE[32] filesystem), NFS servers are not containerized and run outside of the cluster on dedicated virtual machines. Each NFS server of this group participates in a primitive form of load balancing and serves a specific non-overlapping part (a shard) of exported filesystems. In Kubernetes, each exported NFS mount point is represented as a Persistent Volume (PV[33]) and is mapped one-to-one to a Persistent Volume Claim (PVC) which in turn is the abstract storage resource requested by (many) Pods from (possibly different) nodes of the cluster.
- A **Continuous-Integration (CI) server** for managing integration tasks such as running tests, building software artifacts (executables, libraries, Docker images), and uploading Docker images to the site-local private registry. Tasks are triggered from GitHub webhooks when commits are pushed, or tags are created in code repositories that we "watch". The final build status also becomes available to the README page of the origin code repository so

[29] https://kubernetes.io/docs/concepts/workloads/controllers/statefulset/
[30] https://kubernetes.io/docs/concepts/workloads/controllers/job/
[31] https://kubernetes.io/docs/concepts/workloads/controllers/cron-jobs/
[32] https://en.wikipedia.org/wiki/Filesystem_in_Userspace
[33] https://kubernetes.io/docs/concepts/storage/persistent-volumes/

that developer teams can instantly access valuable information on possible errors (e.g., the error log of a failed integration test). Tags of resulting images follow exactly the tag names of the upstream code (which in turn follow semantic-versioning guidelines). In the specific (but very common) case of an update on the patch (1.2.X) version of an application, the build server can "promote" a successful build to a deployment, i.e., it can rollout a redeployment of the application to the staging environment. Currently, the CI server lives outside the Kubernetes cluster (mainly because of the slightly different configuration of the underlying container engine; but this is not a hard requirement and may change in the future). The CI server is deployed as a Drone-CI[34] cluster of strongly-coupled containers (single master, multiple runners) in a dedicated virtual machine.

- A **public-facing LVS[35] virtual server**: this is a master/standby pair of KeepAlived[36] servers that act as transport-layer (OSI Layer 4) proxy to all the HTTP applications exposed by the OpertusMundi system (see Figure 4). The LVS server has a public IP address and is handling all traffic from/to the outer world acting as a NAT router. In this LVS-NAT[37] setup, incoming requests are DNAT-ed and distributed to all available Kubernetes ingress HTTP endpoints (see below). The replies follow the exact reverse path (due to NAT setup) and are finally served by the LVS server[38] back to the requesting client. The LVS server has a crucial position as a single point of failure and so is always accompanied by a *standby clone* ready to take over in case of failure. A failure is detected as a loss of heartbeat during the VRRP[39] communication between master and standby; in such a case, the standby promotes itself to master, assumes ownership of the public IP address, and notifies the administrators on this transition (as the master/standby pair will now operate in a degraded state).
- An **ingress controller** running as a Kubernetes DaemonSet[40]. The ingress[41] controller is the application-layer (OSI Layer 7) proxy that is responsible to terminate SSL connections and distribute external HTTP requests to matching (according to hostname/path rulesets) cluster HTTP services. Currently, we use the Nginx ingress controller due to its high degree of integration with the Kubernetes project. Apart from distributing requests, an ingress controller may also need to preprocess requests and perform authorization-based filtering for certain request patterns. It is important that Nginx-based ingress controller enables us to perform this kind of preprocessing in a language-neutral manner as an authorization *subrequest*[42] i.e., as a nested synchronous request to an external (to the ingress controller) HTTP service responding with a 2xx/4xx HTTP status on success/failure.

[34] https://docs.drone.io/

[35] https://en.wikipedia.org/wiki/Linux_Virtual_Server

[36] https://keepalived.readthedocs.io/en/latest/introduction.html

[37] http://www.linuxvirtualserver.org/VS-NAT.html

[38] The LVS-NAT is simple and secure because it hides all the internal communication to the ingress controllers. But, for huge HTTP workloads (in terms of output throughput and in number of connections) it can become a I/O bottleneck. In such a case we can consider moving to a more complex LVS-DR (Direct Routing) architecture that would require ingress controllers to be partially exposed to the public IP network (to support the reply route).

[39] https://en.wikipedia.org/wiki/Virtual_Router_Redundancy_Protocol

[40] https://kubernetes.io/docs/concepts/workloads/controllers/daemonset/

[41] https://kubernetes.io/docs/concepts/services-networking/ingress/

[42] https://docs.nginx.com/nginx/admin-guide/security-controls/configuring-subrequest-authentication/

- A **private Docker registry** running as a Kubernetes Deployment. All image artifacts (built from the CI server) are tagged and pushed to this registry so they can be available to various Kubernetes workloads. A job runs periodically (inside the cluster, as a Kubernetes CronJob) to perform garbage-collecting tasks on the registry; mainly to remove unused image layers (i.e., not referenced by containers), which on the long run would pile up and waste a significant amount of node-local storage space.
- A **log collector** running a pair of rsyslog[43]/logrotate[44] binaries as a Kubernetes StatefulSet. This log collector can receive log records in the Syslog protocol (RFC-5424[45]) over a TCP or UDP transport layer. The Syslog protocol is widely used and can be considered as the least-common-denominator for remote logging (as almost every programming framework has library support for it). Upon reception, every log record is categorized (based on the attributes it carries) and is either piped as input to another log-analyzing process or stored in a centralized manner. Before finally storing a record, it can possibly transform this record to a shape suitable for later processing. In a microservice-oriented architecture, such as ours, centralization of logs is crucial as it enables us to quickly spot problematic situations and correlate problems with other tasks running on the same time window. Additionally, it prepares a common ground for both offline batch-style analysis (e.g., web analytics for understanding the behavior of customers), and online analysis of logs (e.g., detecting unusually high traffic at real time).
- The **main relational database server** running PostgreSQL/PostGIS as a Kubernetes StatefulSet. This is deployed as a database cluster with a single master and a couple of hot-standby nodes (see also Figure 5). The cluster is operating in asynchronous streaming-replication mode, i.e., standbys are constantly replaying changes directly from the transaction log (Write-Ahead Log, WAL) of the master server. In that way, standbys are effectively replicas of the master (possibly lagging for some tens of milliseconds). To take advantage of this fact, a load-balancing server (PgPool[46]) stands in front of the real database cluster and distributes read-only queries (SELECTs and calls to side-effect-free functions) to all members (i.e., both master and standbys) of the cluster. This reduces load from the master server and improves the overall utilization of the cluster (which is especially important for computationally-intensive spatial queries). Of course, write workload (e.g., INSERT queries) are only handed to the master node and then results (WAL records) are replicated to the entire cluster. We must note that asynchronous standbys are not placed behind PgPool proxy because there is a small (but certainly non-zero, in a high load scenario) possibility of serving inconsistent results. Nevertheless, asynchronous standbys are still useful because they are very close to master's state (so, they can easily be promoted to synchronous in the event of failure of a synchronous standby) and can also serve queries that can tolerate approximate results (e.g. in analytics workloads).

[43] https://www.rsyslog.com/
[44] https://linux.die.net/man/8/logrotate
[45] https://datatracker.ietf.org/doc/html/rfc5424
[46] https://www.pgpool.net/docs/latest/en/html/intro-whatis.html

- A **group of Kubernetes deployments for Core/VAS services of the OpertusMundi platform** (e.g., the Store web application, the Camunda workflow engine, or a VAS service instantiated for a user). Almost all Core/VAS services are represented as Kubernetes Deployments: either store their state in cluster-wide persistent volumes (e.g., PVs backed with a NFS filesystem) or are "stateless" in the sense that they do not directly manipulate their state but instead rely on a database engine (commonly this is just the main PostgreSQL database, but it can also be a database instance dedicated to a user). This representation is flexible enough as it allows for services to be easily scaled-out and also allows individual Pods to be rescheduled on other nodes of the Kubernetes cluster (in case, for example, of a node failure).

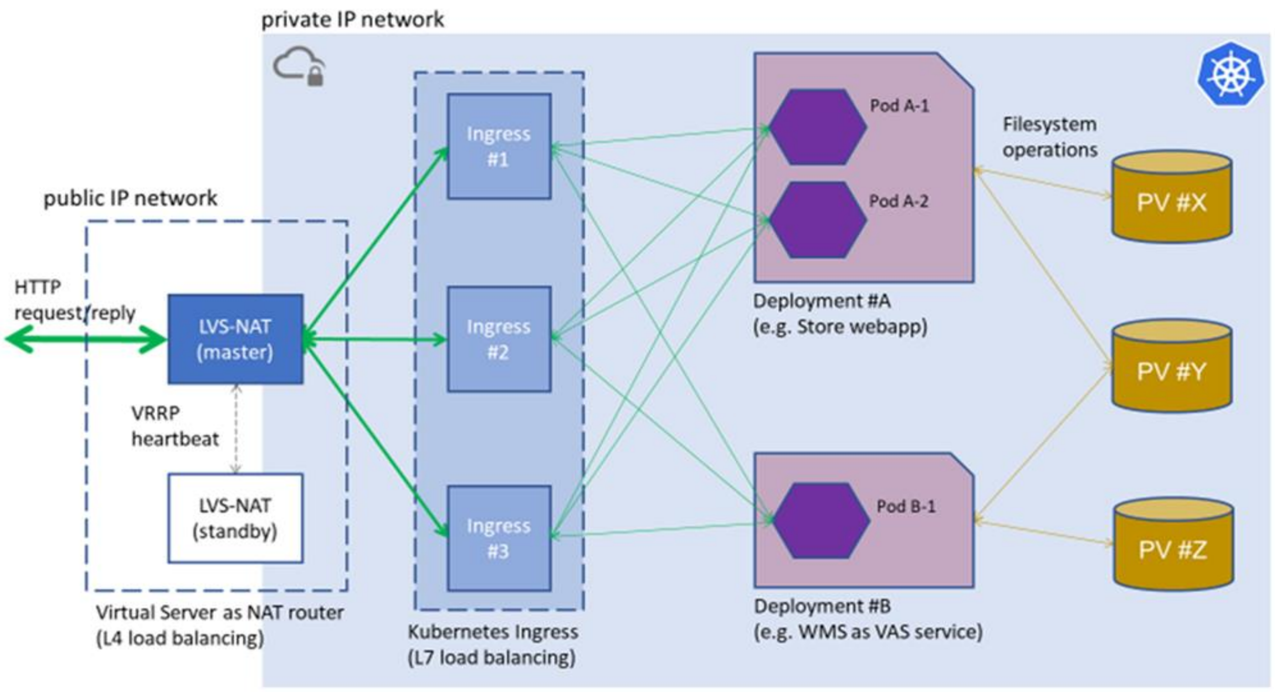


Figure 4: Kubernetes Ingress

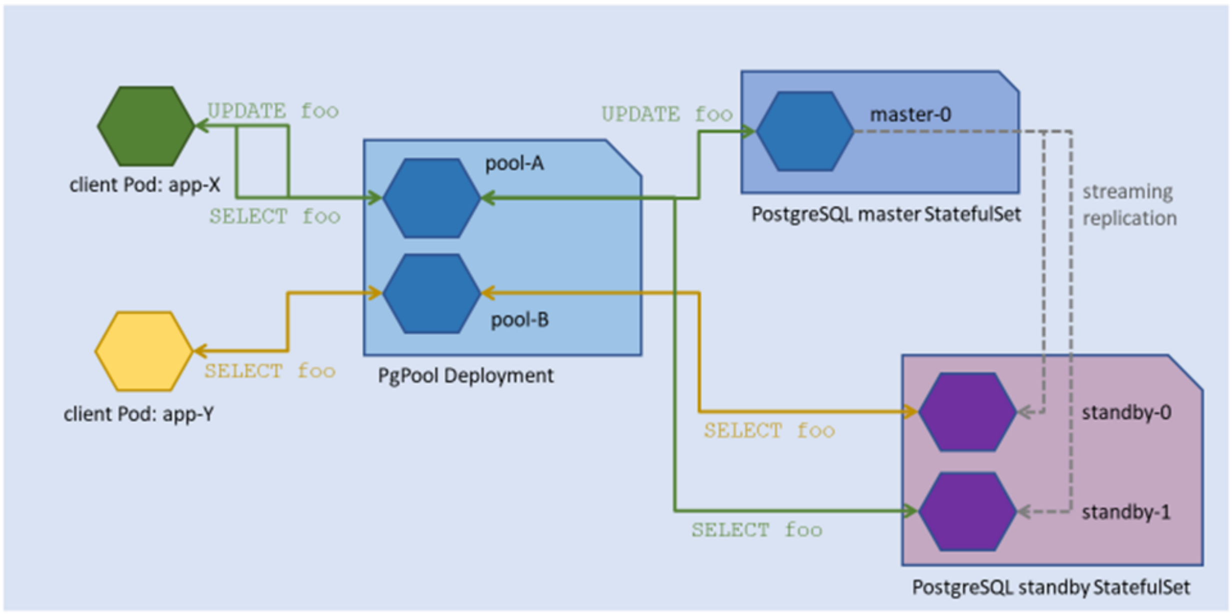


Figure 5: Overview of PostgreSQL database cluster

1.2.2. Packaging

In the OpertusMundi platform, all microservices, libraries and command-line tools are packaged as versioned Docker images and uploaded to an internal Docker registry. On each runtime environment (be it a VM or a bare-metal machine), the platform orchestration tool (Kubernetes) pulls the required images and creates containers from them.

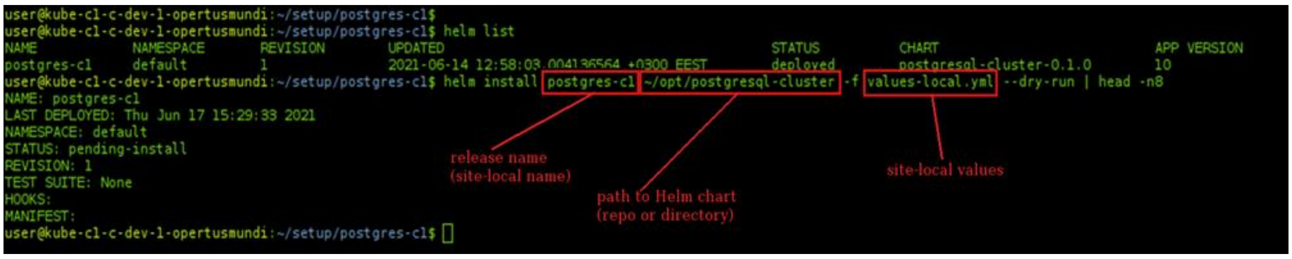
Every software component of the OpertusMundi platform (a tool, a library, or a service) is also accompanied by a build recipe to package it as a Docker image. This recipe, the so-called Dockerfile, is part of the source code repository of the component and is invoked by the build server (triggered by a Git push or a tag event) to produce a container image which is also versioned along with the packaged component. As mentioned before in Section 1.2.1, all upstream projects are expected to adhere to semantic-versioning[47] guidelines and so Git tags are reused to tag Docker images (but keeping only the first 3 levels of X.Y.Z version and discarding trailing metadata). In this way, there is a straightforward relation between a container's image and the code tag/commit it originated from.

The majority of OpertusMundi services are rolled out as Kubernetes deployments (or statefulsets, but this makes no difference here). In contrast to a Dockerfile that builds a container image, a deployment manifest is usually not under source control because it depends on site-specific parameters which are either security-sensitive or are only known at the runtime environment. It is rather common for only a deployment template to be kept under source control so it can be later used as a starting point for configuring and fine-tuning the real deployment into the real environment. This configuration step can be short, tidy and less error-prone (less manual editing)

[47] https://semver.org/

by referencing sections of configuration objects[48] that already exist (and of course, are pre-validated) inside the production Kubernetes cluster.

This packaging step for a deployment is generally handled by creating a Helm[49] chart so that we factor-out the configurable values and compose the final deployment manifest from a template that renders the actual values at runtime (aka deployment time). In a nutshell, a Helm chart describes what is (and what is not) configurable and how a runtime-provided configuration will generate an actual deployment using templating code and validation logic. The term "validation" is quite broad here, as it can vary from a simple check inside the templating code (e.g., is a string value non-empty?), to complex pre-/post-install tests that may also examine the target runtime environment (e.g., can we ping the service we have just deployed?). A Helm chart can be kept under a source code repository because all site-specific values are factored out; usually these values are just replaced by examples and are accompanied by helpful documentation snippets. On a particular runtime environment (e.g., on production), an *installed* Helm chart automatically keeps a version of itself by assigning a serial number to the set of *actual* configuration values it has derived from. So, it is possible to roll-back to a previous version (in case of an unexpected error, or in case of a post-install test failure) or to just compare differences with previously installed versions.



```
user@kube-cl-c-dev-1-opertusmundi:~/setup/postgres-cl$
user@kube-cl-c-dev-1-opertusmundi:~/setup/postgres-cl$ helm list
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART                           APP VERSION
postgres-cl     default         1               2021-06-14 12:58:03.004136564 +0300 EEST deployed       postgresql-cluster-0.1.0        10
user@kube-cl-c-dev-1-opertusmundi:~/setup/postgres-cl$ helm install postgres-cl ~/opt/postgresql-cluster -f values-local.yml --dry-run | head -n8
NAME: postgres-cl
LAST DEPLOYED: Thu Jun 17 15:29:33 2021
NAMESPACE: default
STATUS: pending-install
REVISION: 1
TEST SUITE: None
HOOKS:
MANIFEST:
user@kube-cl-c-dev-1-opertusmundi:~/setup/postgres-cl$
```

Figure 6: Installing a Helm chart

1.3. Documentation

In this sub-section, we describe the documentation requirements of the project, present the tools used for generating documentation, and provide examples and links to the generated documentation for the OpertusMundi system.

In the OpertusMundi project there are four distinct cases where documentation is required. First, project documentation is required to easily deploy and configure the system in a development or production environment. Second, server-side code that manages data, job execution and security, requires documentation to be accessible and extensible. Likewise, client-side code that executes on the browser needs to be documented. Finally, API documentation that allows client-side code and services to interact with each other must be well documented for the developers to access functionality exposed by the OpertusMundi services. Depending on the code being documented, several tools have been used for generating documentation. In all cases, the documentation

48 https://kubernetes.io/docs/concepts/configuration/configmap/
49 https://helm.sh/docs/topics/charts/

generation process has been integrated in the project build process. Hence, the documentation is updated whenever the code changes. The whole documentation is packaged along with the OpertusMundi distributable artifacts and becomes accessible once the OpertusMundi application has been deployed. Moreover, the documentation artifacts are committed to the GitHub repository and published using the GitHub Pages[50] feature.

During development, we use several tools for generating documentation for server-side code, client-side code, and API methods signatures. For server-side code, we produce code primarily using the Java and Python programming languages, and thus generate documentation with Javadoc[51] and Sphinx[52] respectively. These tools generate documentation files from comments inside the source code, making code self-documented and easier to understand.

For the API documentation, we require every service implemented for the OpertusMundi platform to publish its documentation as an OpenAPI document, i.e., a document that conforms to the Open API Specification[53] (OAS). OAS is a standard, language-agnostic interface for APIs that allows both humans and computers to discover and understand the capabilities of a service. There are several libraries for generating an OpenAPI document from source code, such as SpringDoc[54] and OpenAPI Core[55] for Java and Python, respectively. The generated document can be viewed using tools such as Swagger UI[56] or ReDoc[57].

For the project deployment documentation, we use simple pages written with Markdown[58] syntax and commit the documentation as part of the source code.

Finally, all tools described above have full support for templating, thus allowing the configuration of the layout and styling of the generated documentation.

A reference to documentation artifacts for all OpertusMundi software components can be found at:

- https://github.com/OpertusMundi/website/wiki

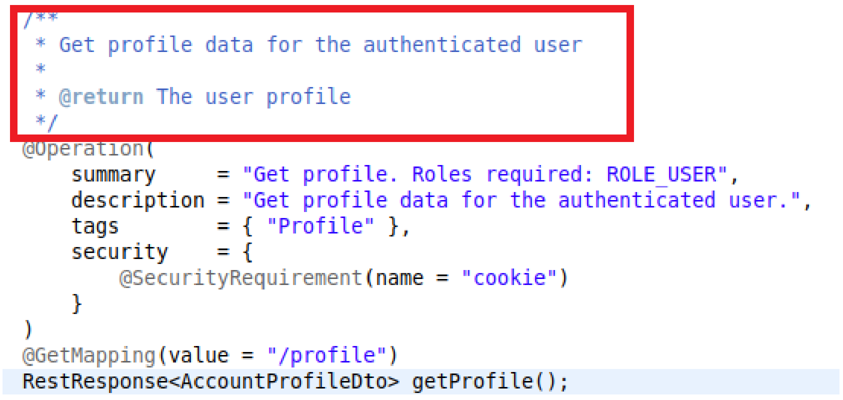
Next, we give a brief description of every tool, followed by a simple example.

1.3.1.1. Javadoc

The Javadoc tool is used for generating documentation for Java source code. It does so by parsing the documentation comments from all java source files and produces HTML pages for every package, class, and interface. For each class and interface, documentation is generated for public

[50] https://pages.github.com/
[51] https://docs.oracle.com/javase/8/docs/technotes/tools/windows/javadoc.html
[52] https://www.sphinx-doc.org/en/master/
[53] https://swagger.io/specification/
[54] https://springdoc.org/
[55] https://github.com/p1c2u/openapi-core
[56] https://swagger.io/tools/swagger-ui/
[57] https://github.com/Redocly/redoc
[58] https://en.wikipedia.org/wiki/Markdown

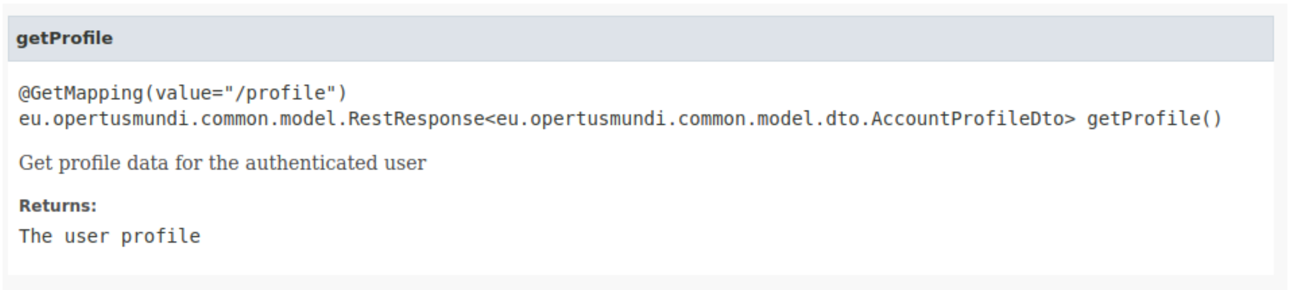
and protected constructors, methods, and fields. An example of Javadoc-compatible documentation comments is depicted in Figure 7.



```
/**
 * Get profile data for the authenticated user
 *
 * @return The user profile
 */
@Operation(
    summary     = "Get profile. Roles required: ROLE_USER",
    description = "Get profile data for the authenticated user.",
    tags        = { "Profile" },
    security    = {
        @SecurityRequirement(name = "cookie")
    }
)
@GetMapping(value = "/profile")
RestResponse<AccountProfileDto> getProfile();
```

Figure 7: Javadoc example

The produced documentation for the code in Figure 7 is illustrated in Figure 8.



getProfile

```
@GetMapping(value="/profile")
eu.opertusmundi.common.model.RestResponse<eu.opertusmundi.common.model.dto.AccountProfileDto> getProfile()
```

Get profile data for the authenticated user

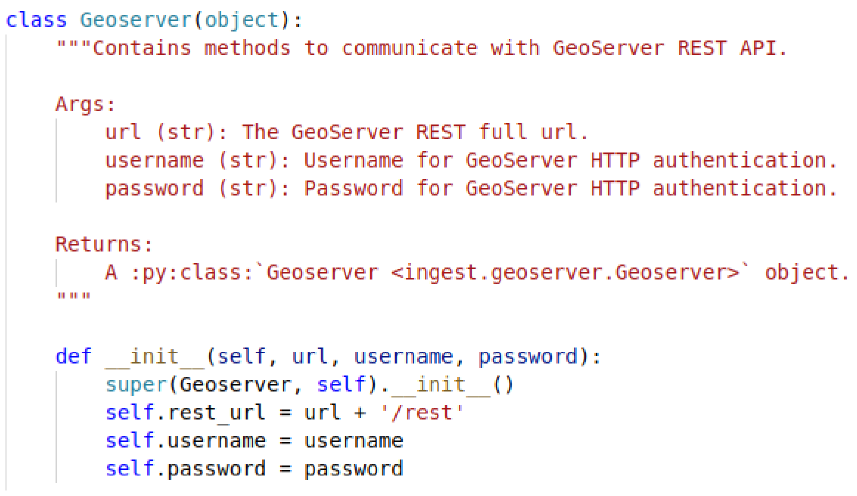
Returns:

`The user profile`

Figure 8: Javadoc documentation example

1.3.1.2. Sphinx

Sphinx is a documentation generator for the Python programming language. It creates documentation by parsing comments in the source code and produces HTML pages (or files in other supported formats such as plain text) for every module, class, and method. Examples of documentation comments and the actual rendered result are depicted in Figure 9 and Figure 10 respectively.



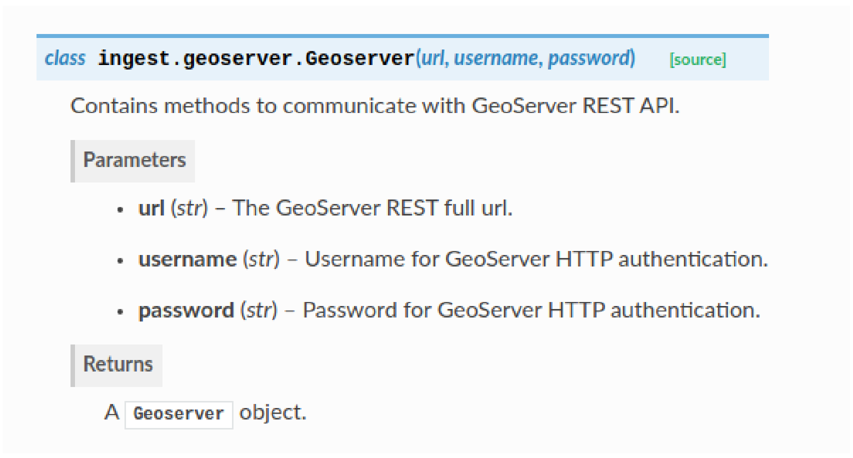
```
class Geoserver(object):
    """Contains methods to communicate with GeoServer REST API.

    Args:
        url (str): The GeoServer REST full url.
        username (str): Username for GeoServer HTTP authentication.
        password (str): Password for GeoServer HTTP authentication.

    Returns:
        A :py:class:`Geoserver <ingest.geoserver.Geoserver>` object.
    """

    def __init__(self, url, username, password):
        super(Geoserver, self).__init__()
        self.rest_url = url + '/rest'
        self.username = username
        self.password = password
```

Figure 9: Python comments


class **ingest.geoserver.Geoserver**(*url, username, password*) [source]

Contains methods to communicate with GeoServer REST API.

Parameters

- **url** (*str*) – The GeoServer REST full url.
- **username** (*str*) – Username for GeoServer HTTP authentication.
- **password** (*str*) – Password for GeoServer HTTP authentication.

Returns

A `Geoserver` object.

Figure 10: Sphinx generated documentation

1.3.1.3. SpringDoc

In the OpertusMundi project, we use the Spring Boot framework for developing services. SpringDoc is a library for generating OpenAPI documentation for Spring Boot projects. The documentation is generated at runtime by inspecting the application configuration and annotations on classes and methods. The library supports annotations for Spring MVC such as mappings, validation annotations and custom OpenAPI annotations. Examples of annotations for a method and a class are shown in Figure 11 and Figure 12 respectively.

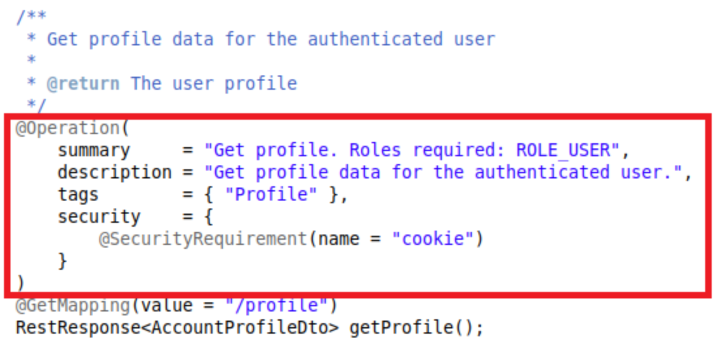
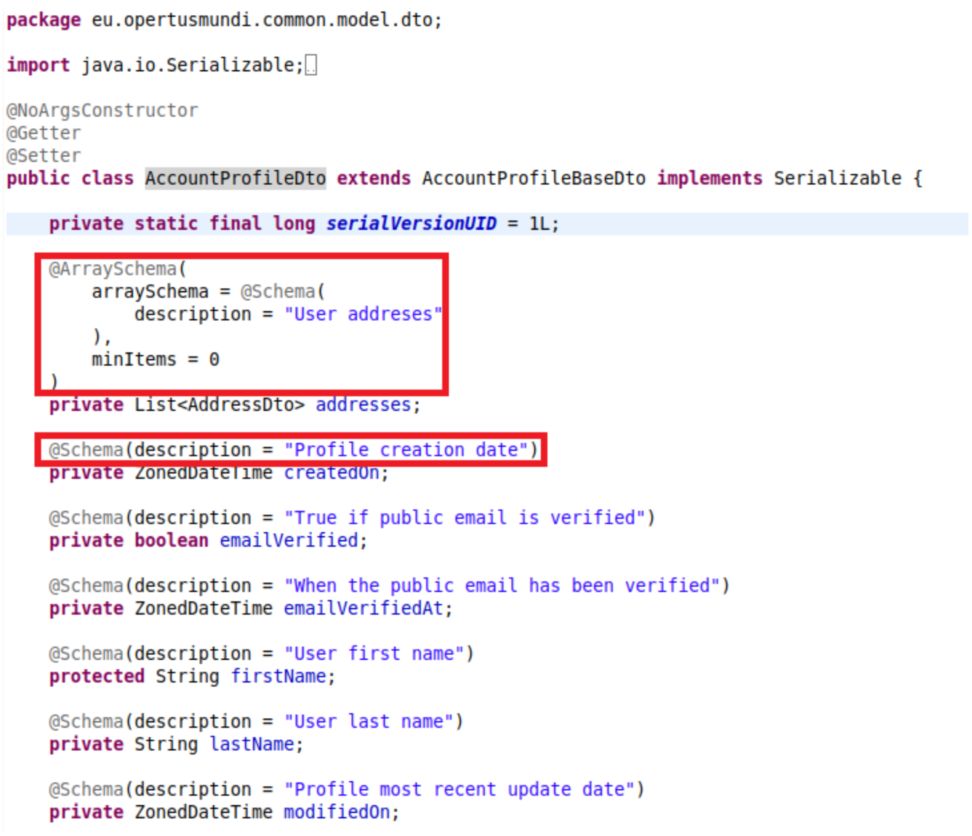

```
/**
 * Get profile data for the authenticated user
 *
 * @return The user profile
 */
@Operation(
    summary     = "Get profile. Roles required: ROLE_USER",
    description = "Get profile data for the authenticated user.",
    tags        = { "Profile" },
    security    = {
        @SecurityRequirement(name = "cookie")
    }
)
@GetMapping(value = "/profile")
RestResponse<AccountProfileDto> getProfile();
```

Figure 11: SpringDoc annotations for a method



```java
package eu.opertusmundi.common.model.dto;

import java.io.Serializable;

@NoArgsConstructor
@Getter
@Setter
public class AccountProfileDto extends AccountProfileBaseDto implements Serializable {

    private static final long serialVersionUID = 1L;

    @ArraySchema(
        arraySchema = @Schema(
            description = "User addreses"
        ),
        minItems = 0
    )
    private List<AddressDto> addresses;

    @Schema(description = "Profile creation date")
    private ZonedDateTime createdOn;

    @Schema(description = "True if public email is verified")
    private boolean emailVerified;

    @Schema(description = "When the public email has been verified")
    private ZonedDateTime emailVerifiedAt;

    @Schema(description = "User first name")
    protected String firstName;

    @Schema(description = "User last name")
    private String lastName;

    @Schema(description = "Profile most recent update date")
    private ZonedDateTime modifiedOn;
```

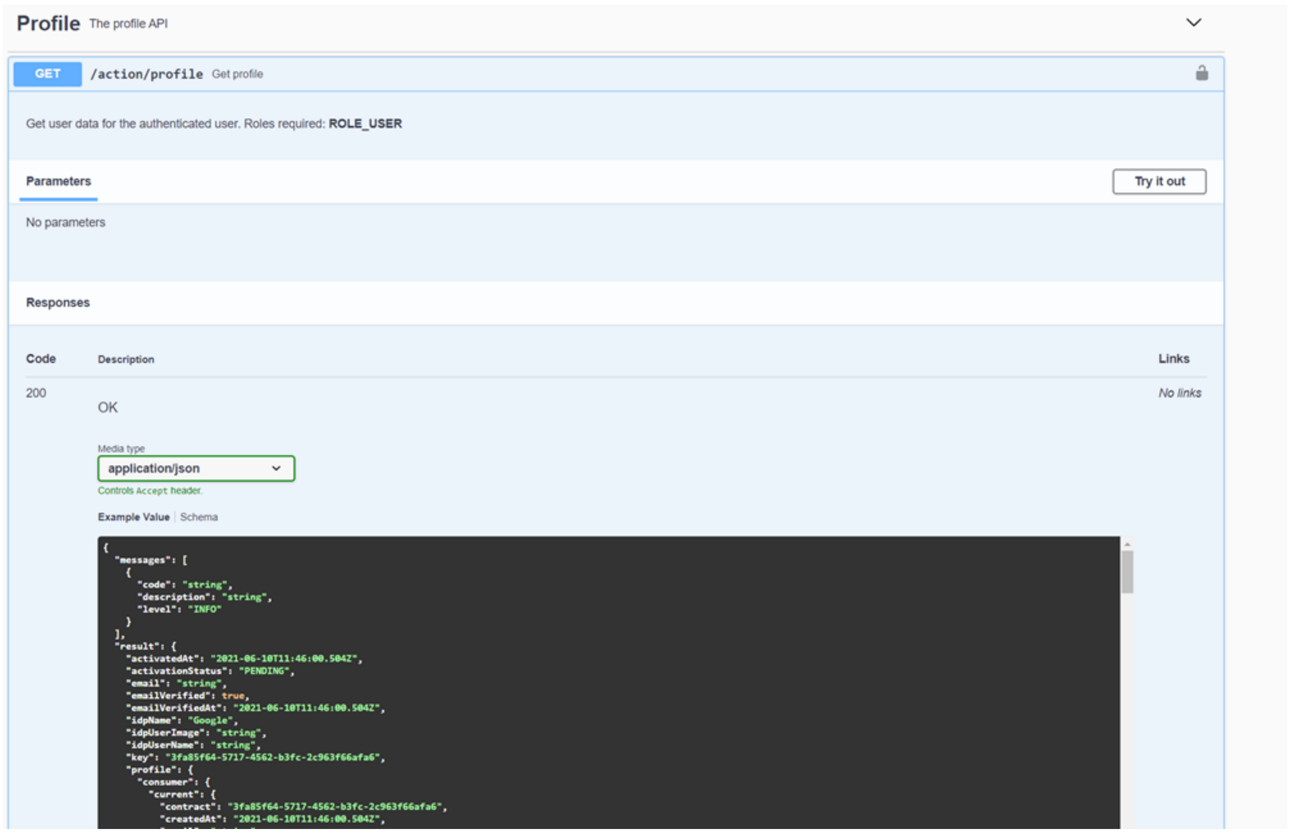
Figure 12: SpringDoc annotations for a class

The generated OpenAPI document for the OpertusMundi API gateway is available at:

- https://api.dev.opertusmundi.eu/api-docs

The OpenAPI document can be used to visualize the documentation using a tool like Swagger UI or ReDoc. OpertusMundi API gateway supports both tools. Examples for both tools are depicted in Figure 13 and Figure 14. Swagger UI allows developers to test the API by invoking methods using the UI. ReDoc does not support API invocation but offers better readability. The documentation rendered by both tools is available at:

- https://api.dev.opertusmundi.eu/swagger-ui/index.html?configUrl=/api-docs/swagger-config
- https://api.dev.opertusmundi.eu/docs



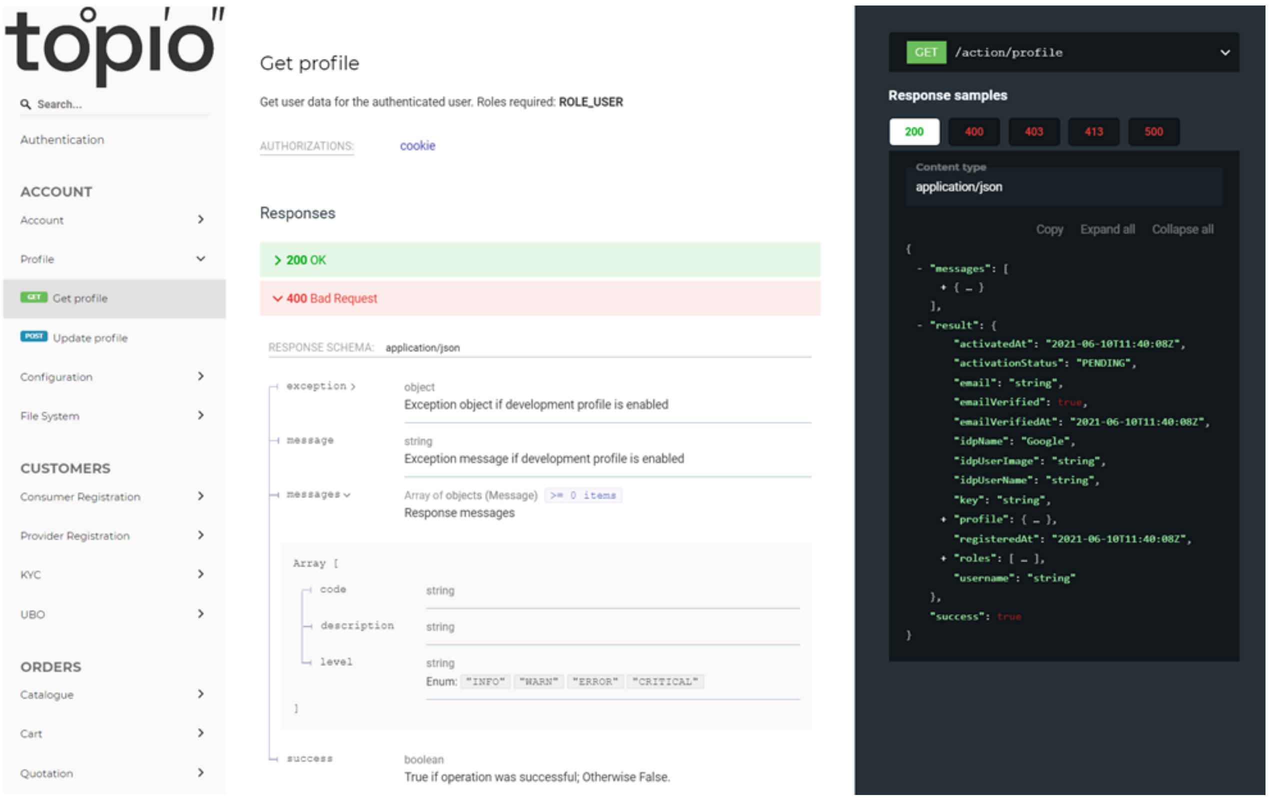
Get user data for the authenticated user. Roles required: **ROLE_USER**

No parameters

Code	Description	Links
200	OK	*No links*

```
{
  "messages": [
    {
      "code": "string",
      "description": "string",
      "level": "INFO"
    }
  ],
  "result": {
    "activatedAt": "2021-06-10T11:46:00.504Z",
    "activationStatus": "PENDING",
    "email": "string",
    "emailVerified": true,
    "emailVerifiedAt": "2021-06-10T11:46:00.504Z",
    "idpName": "Google",
    "idpUserImage": "string",
    "idpUserName": "string",
    "key": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
    "profile": {
      "consumer": {
        "current": {
          "contract": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
          "createdAt": "2021-06-10T11:46:00.504Z",
```

Figure 13: Open API documentation with Swagger UI



Get profile

Get user data for the authenticated user. Roles required: **ROLE_USER**

AUTHORIZATIONS: cookie

Responses

> 200 OK

⌄ 400 Bad Request

RESPONSE SCHEMA: application/json

`exception >`	object Exception object if development profile is enabled
`message`	string Exception message if development profile is enabled
`messages ⌄`	Array of objects (Message) `>= 0 items` Response messages

Array [

`code`	string
`description`	string
`level`	string Enum: `"INFO"` `"WARN"` `"ERROR"` `"CRITICAL"`

]

`success`	boolean True if operation was successful; Otherwise False.

GET /action/profile

Response samples

200 400 403 413 500

Content type
application/json

Copy Expand all Collapse all

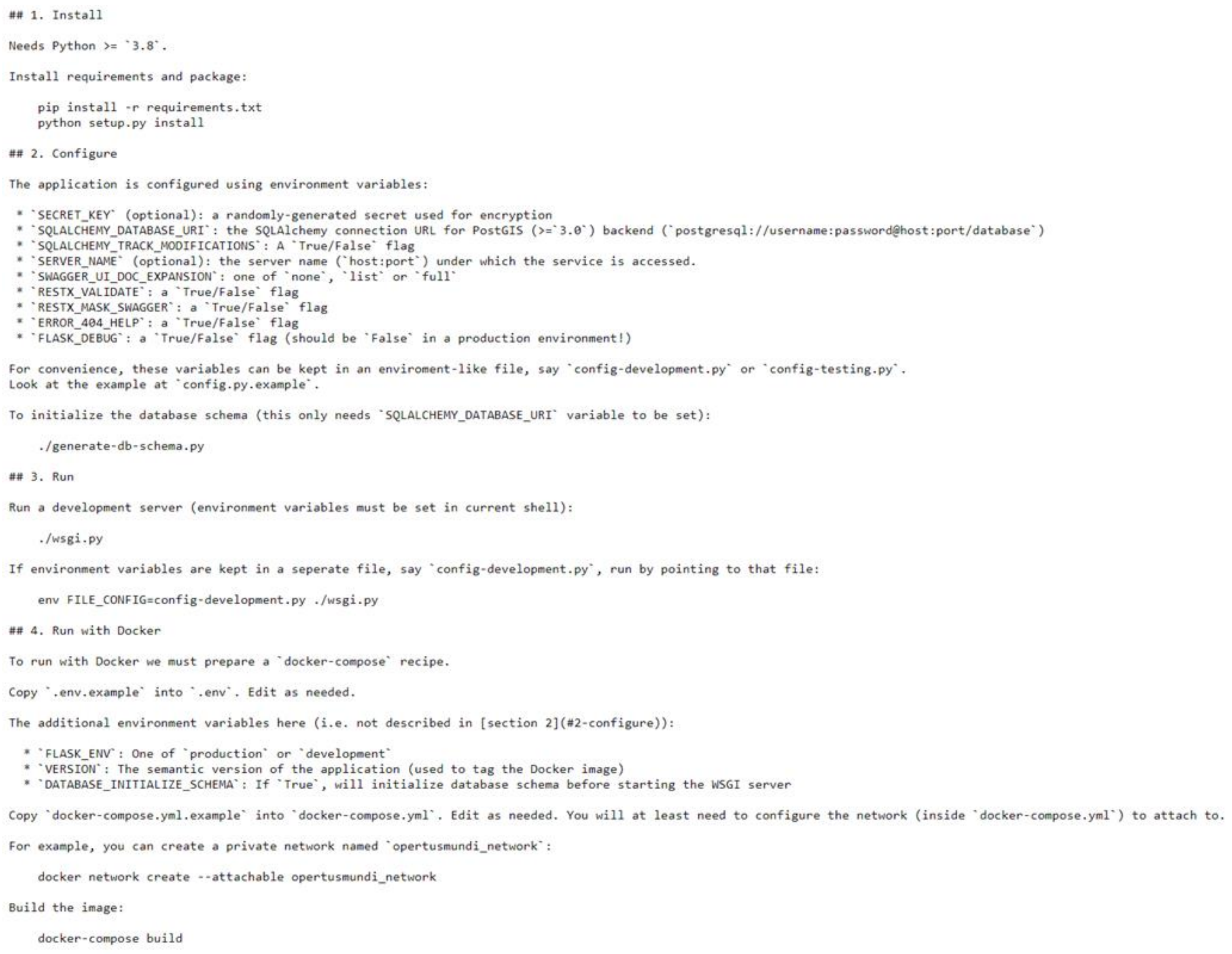
```
{
  - "messages": [
      + { … }
    ],
  - "result": {
        "activatedAt": "2021-06-10T11:40:08Z",
        "activationStatus": "PENDING",
        "email": "string",
        "emailVerified": true,
        "emailVerifiedAt": "2021-06-10T11:40:08Z",
        "idpName": "Google",
        "idpUserImage": "string",
        "idpUserName": "string",
        "key": "string",
      + "profile": { … },
        "registeredAt": "2021-06-10T11:40:08Z",
      + "roles": [ … ],
        "username": "string"
    },
    "success": true
}
```

Figure 14: Open API documentation with ReDoc

1.3.1.4. Markdown

Markdown is a lightweight markup language with plain text format that is commonly used for generating HTML documentation. Repository hosting services like GitHub, support editing of markdown documents and automatic rendering to HTML. An example of a markdown document from the OpertusMundi catalogue repository is shown in Figure 15. The generated HTML pages are available at:

- https://github.com/OpertusMundi/catalogue-service/blob/master/README.md



```
## 1. Install

Needs Python >= `3.8`.

Install requirements and package:

    pip install -r requirements.txt
    python setup.py install

## 2. Configure

The application is configured using environment variables:

 * `SECRET_KEY` (optional): a randomly-generated secret used for encryption
 * `SQLALCHEMY_DATABASE_URI`: the SQLAlchemy connection URL for PostGIS (>=`3.0`) backend (`postgresql://username:password@host:port/database`)
 * `SQLALCHEMY_TRACK_MODIFICATIONS`: A `True/False` flag
 * `SERVER_NAME` (optional): the server name (`host:port`) under which the service is accessed.
 * `SWAGGER_UI_DOC_EXPANSION`: one of `none`, `list` or `full`
 * `RESTX_VALIDATE`: a `True/False` flag
 * `RESTX_MASK_SWAGGER`: a `True/False` flag
 * `ERROR_404_HELP`: a `True/False` flag
 * `FLASK_DEBUG`: a `True/False` flag (should be `False` in a production environment!)

For convenience, these variables can be kept in an enviroment-like file, say `config-development.py` or `config-testing.py`.
Look at the example at `config.py.example`.

To initialize the database schema (this only needs `SQLALCHEMY_DATABASE_URI` variable to be set):

    ./generate-db-schema.py

## 3. Run

Run a development server (environment variables must be set in current shell):

    ./wsgi.py

If environment variables are kept in a seperate file, say `config-development.py`, run by pointing to that file:

    env FILE_CONFIG=config-development.py ./wsgi.py

## 4. Run with Docker

To run with Docker we must prepare a `docker-compose` recipe.

Copy `.env.example` into `.env`. Edit as needed.

The additional environment variables here (i.e. not described in [section 2](#2-configure)):

  * `FLASK_ENV`: One of `production` or `development`
  * `VERSION`: The semantic version of the application (used to tag the Docker image)
  * `DATABASE_INITIALIZE_SCHEMA`: If `True`, will initialize database schema before starting the WSGI server

Copy `docker-compose.yml.example` into `docker-compose.yml`. Edit as needed. You will at least need to configure the network (inside `docker-compose.yml`) to attach to.

For example, you can create a private network named `opertusmundi_network`:

    docker network create --attachable opertusmundi_network

Build the image:

    docker-compose build
```

Figure 15: Markdown example

2. The Topio Marketplace

In this section, we present a walkthrough of the services and functionalities provided by the platform to its various users, as well as examples of select core workflows, demonstrating the maturity of the platform. The reader is reminded the following:

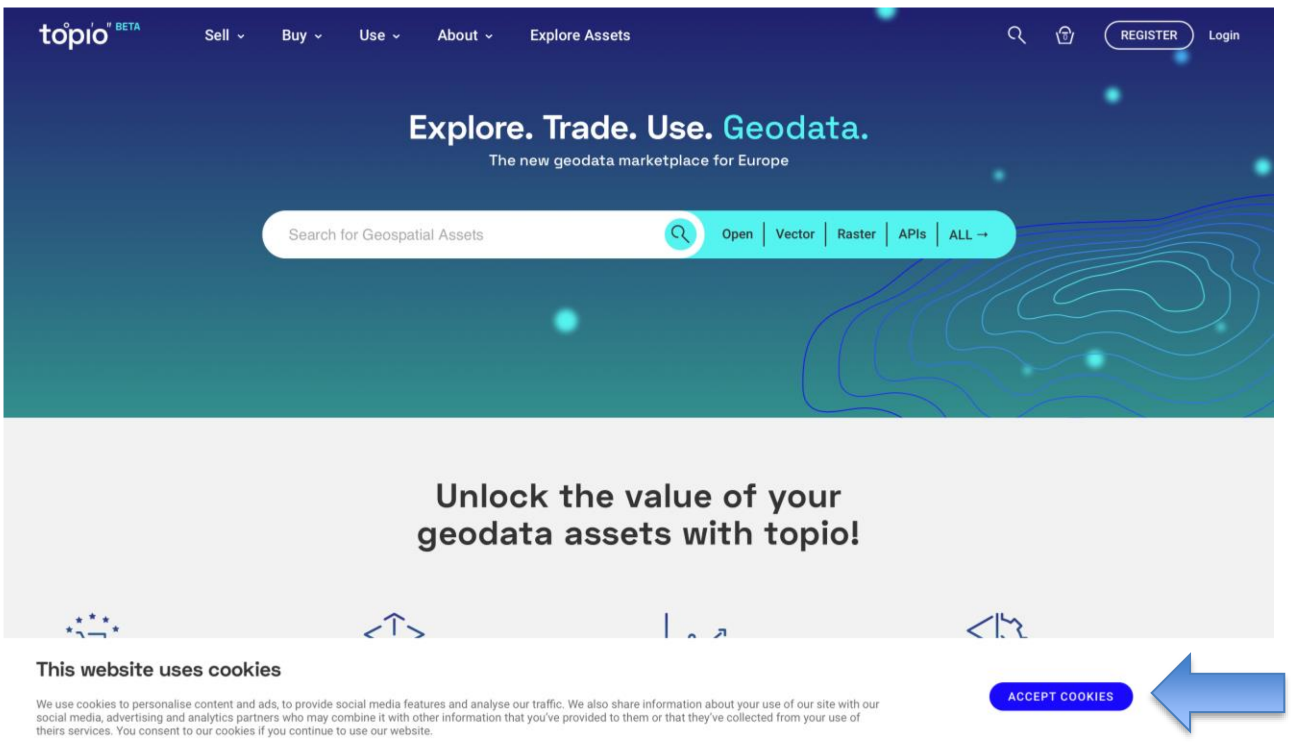
- The presentation that follows is not intended to be exhaustive in nature and should not be considered as a comprehensive documentation for the platform's various offerings; these areas are addressed by the corresponding deliverables presenting in more detail the individual platform services and their operation.
- Interested data suppliers are encouraged to contact us for private demonstrations, business inquiries, and asset onboarding (please see Executive Summary).
- All screenshots and descriptions are correct as of the time of this writing and are expected to be modified in the future, as the platform continues to be in development, even after the end of the OpertusMundi project. In addition, the platform's messages, labels, and assets (data, services) presented in the following screenshots are integrated for *testing and illustration purposes alone* and hence *may not always correspond* to actual commercial or proprietary geospatial assets. As such: (a) the same assets may appear multiple times, (b) the content of assets may be derived from open licensed assets, (c) the provided metadata (e.g., pricing, terms, profiling) may not be accurate.

2.1. Landing page

2.1.1. Overview

The landing page of Topio welcomes a user when she first visits the platform, communicating the core functionalities to geospatial assets users, prospective buyers, and data suppliers[59]. The landing page was designed to be as intuitive as possible, provide a streamlined experience to visitors, pique their interest, and enable them to easily discover the marketplace's offerings. Figure 16 illustrates Topio's landing page. At the center of the page lies the "Search Assets" bar, which the user can click to initiate a search for geospatial assets. Of course, the user must first accept the cookies used by Topio, by clicking on the corresponding button at the bottom of the page (blue arrow in Figure 16).

[59] Please note that since M26 throughout the Topio marketplace and our deliverables we reference geospatial data providers, sellers, vendors, or owners, simply as 'suppliers' to simplify the messaging and terminology of Topio. According to our discussions with geospatial data suppliers and our communications team, this term encompasses all types of organizations and companies offering the geospatial assets through the market and is much preferable for a consistent messaging.



Explore. Trade. Use. Geodata.

The new geodata marketplace for Europe

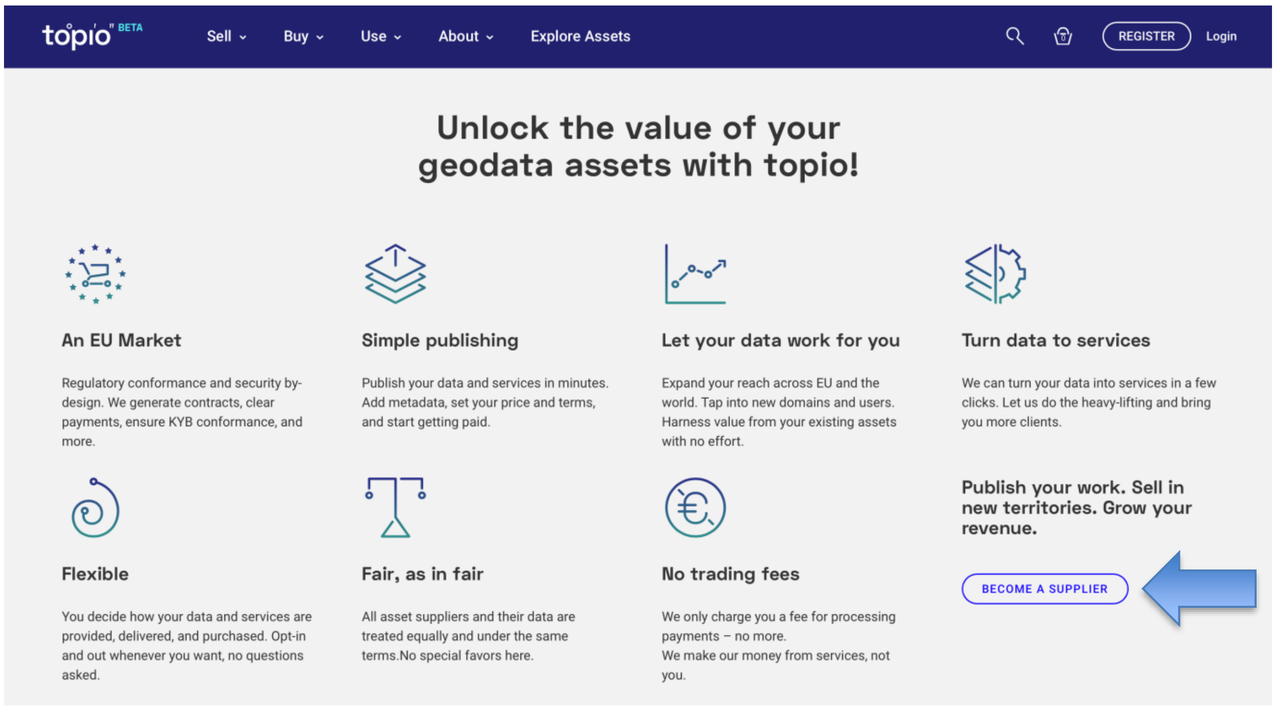
Unlock the value of your geodata assets with topio!

This website uses cookies

We use cookies to personalise content and ads, to provide social media features and analyse our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of theirs services. You consent to our cookies if you continue to use our website.

Figure 16: Topio landing page

Scrolling further down the landing page, the user is presented with more information regarding the benefits of becoming a supplier in Topio (Figure 17), along with a button (indicated with a blue arrow), that redirects to the "Register as an Asset Supplier" page (see next section).



Unlock the value of your geodata assets with topio!

An EU Market

Regulatory conformance and security by-design. We generate contracts, clear payments, ensure KYB conformance, and more.

Simple publishing

Publish your data and services in minutes. Add metadata, set your price and terms, and start getting paid.

Let your data work for you

Expand your reach across EU and the world. Tap into new domains and users. Harness value from your existing assets with no effort.

Turn data to services

We can turn your data into services in a few clicks. Let us do the heavy-lifting and bring you more clients.

Flexible

You decide how your data and services are provided, delivered, and purchased. Opt-in and out whenever you want, no questions asked.

Fair, as in fair

All asset suppliers and their data are treated equally and under the same terms.No special favors here.

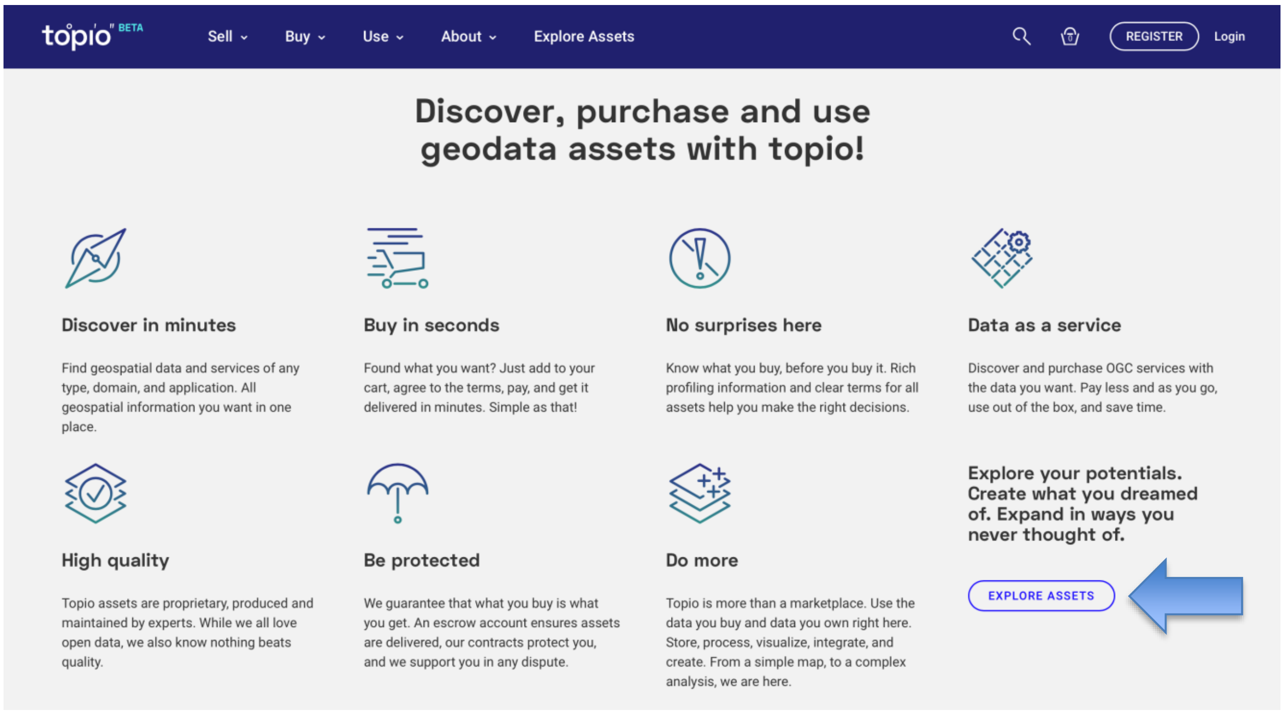
No trading fees

We only charge you a fee for processing payments – no more.
We make our money from services, not you.

Publish your work. Sell in new territories. Grow your revenue.

Figure 17: Overview of the platform's benefits for suppliers

Below the supplier benefit list, there is also an overview of the benefits of buying assets from Topio, as depicted in Figure 18. At the end of the list, there is a button (indicated with a blue arrow) redirecting to the “Asset Search” page, listing all the currently available assets in Topio (for more information, please refer to Section 2.1.2).



Discover, purchase and use geodata assets with topio!

Discover in minutes

Find geospatial data and services of any type, domain, and application. All geospatial information you want in one place.

Buy in seconds

Found what you want? Just add to your cart, agree to the terms, pay, and get it delivered in minutes. Simple as that!

No surprises here

Know what you buy, before you buy it. Rich profiling information and clear terms for all assets help you make the right decisions.

Data as a service

Discover and purchase OGC services with the data you want. Pay less and as you go, use out of the box, and save time.

High quality

Topio assets are proprietary, produced and maintained by experts. While we all love open data, we also know nothing beats quality.

Be protected

We guarantee that what you buy is what you get. An escrow account ensures assets are delivered, our contracts protect you, and we support you in any dispute.

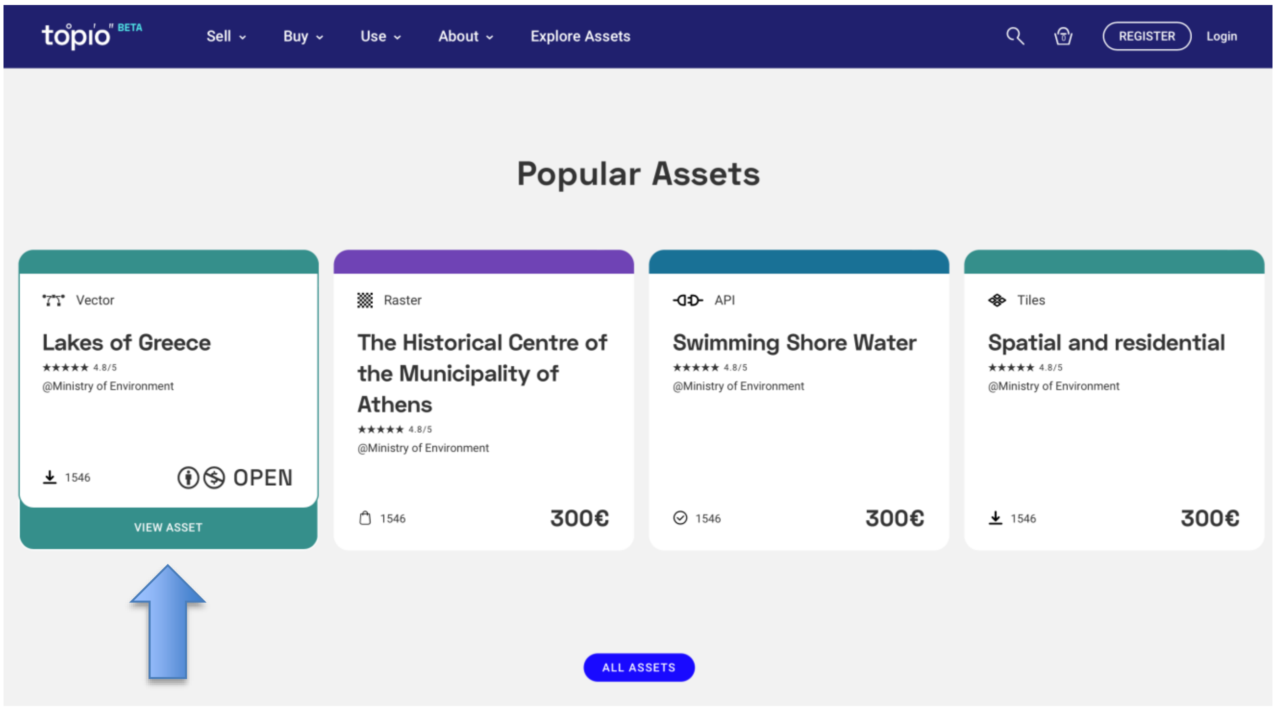
Do more

Topio is more than a marketplace. Use the data you buy and data you own right here. Store, process, visualize, integrate, and create. From a simple map, to a complex analysis, we are here.

Explore your potentials. Create what you dreamed of. Expand in ways you never thought of.

Figure 18: Overview of the platform's benefits for buyers

Next, the prospective user is presented with a vertically scrollable collection of popular assets that other users mostly search for in the platform, as illustrated in Figure 19. By hovering over an item, a “View Asset” button is revealed, which transfers the user to the corresponding “Asset View” page (blue arrow in the figure). There is also an “All Assets” button, which forwards the user to the “Asset Search” page, where all the available assets are listed.


topio BETA Sell Buy Use About Explore Assets REGISTER Login

Popular Assets

Vector
Lakes of Greece
★★★★★ 4.8/5
@Ministry of Environment
1546 OPEN
VIEW ASSET

Raster
The Historical Centre of the Municipality of Athens
★★★★★ 4.8/5
@Ministry of Environment
1546 300€

API
Swimming Shore Water
★★★★★ 4.8/5
@Ministry of Environment
1546 300€

Tiles
Spatial and residential
★★★★★ 4.8/5
@Ministry of Environment
1546 300€

ALL ASSETS

Figure 19: Popular assets list

Scrolling further down, the user is presented with a teaser video, describing the idea behind Topio, as illustrated in Figure 20. The video is directly playable by clicking on the play button. Finally, at the bottom of the page (Figure 21) there is a section listing the main differences of Topio from other data marketplaces, followed by a small section that summarizes the benefits of becoming a supplier or buyer at Topio, with links to the corresponding register pages (see next section).

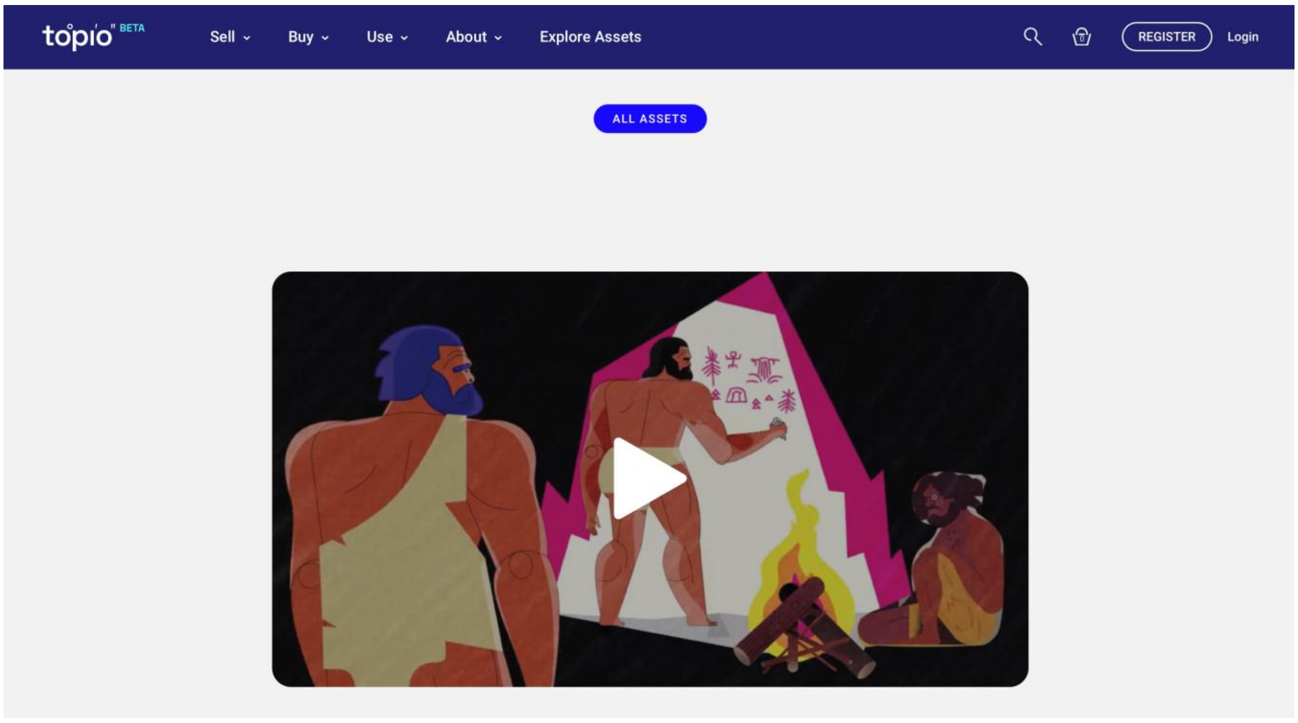
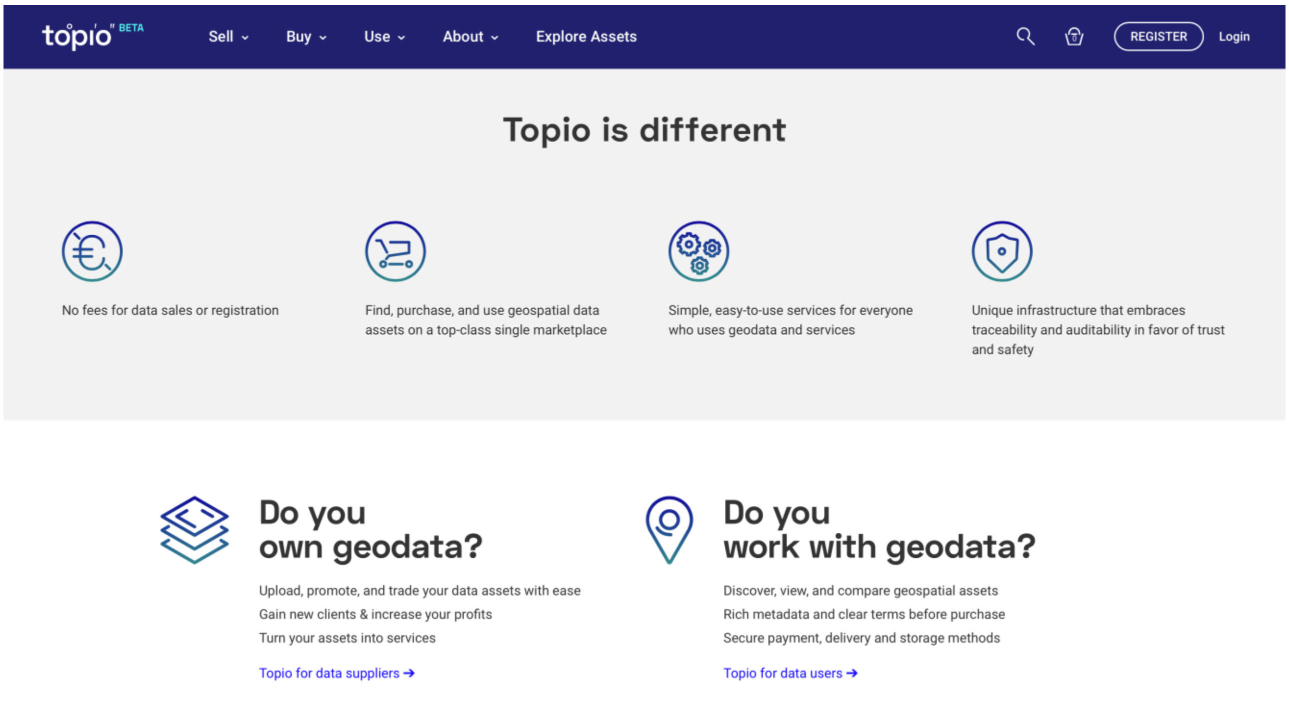


Figure 20: Teaser video



Topio is different

No fees for data sales or registration

Find, purchase, and use geospatial data assets on a top-class single marketplace

Simple, easy-to-use services for everyone who uses geodata and services

Unique infrastructure that embraces traceability and auditability in favor of trust and safety

Do you own geodata?

Upload, promote, and trade your data assets with ease
Gain new clients & increase your profits
Turn your assets into services

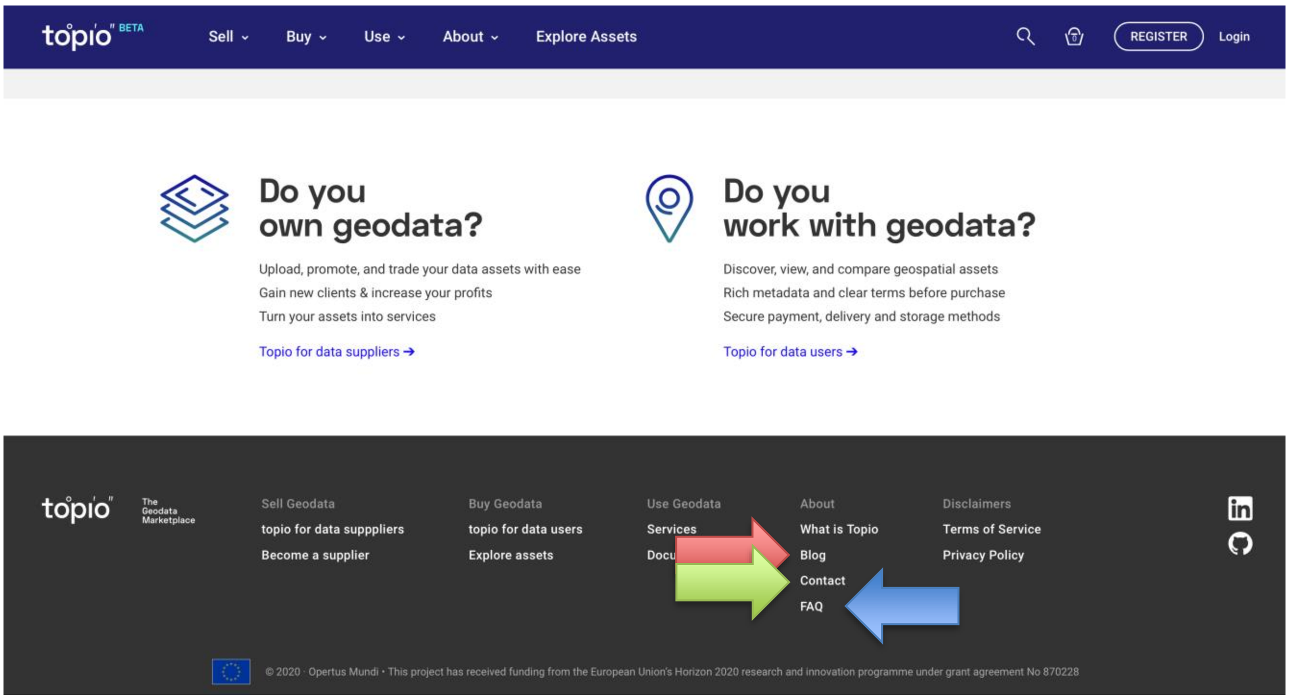
Topio for data suppliers →

Do you work with geodata?

Discover, view, and compare geospatial assets
Rich metadata and clear terms before purchase
Secure payment, delivery and storage methods

Topio for data users →

Figure 21: Why Topio is different



Do you own geodata?

Upload, promote, and trade your data assets with ease
Gain new clients & increase your profits
Turn your assets into services

Topio for data suppliers →

Do you work with geodata?

Discover, view, and compare geospatial assets
Rich metadata and clear terms before purchase
Secure payment, delivery and storage methods

Topio for data users →

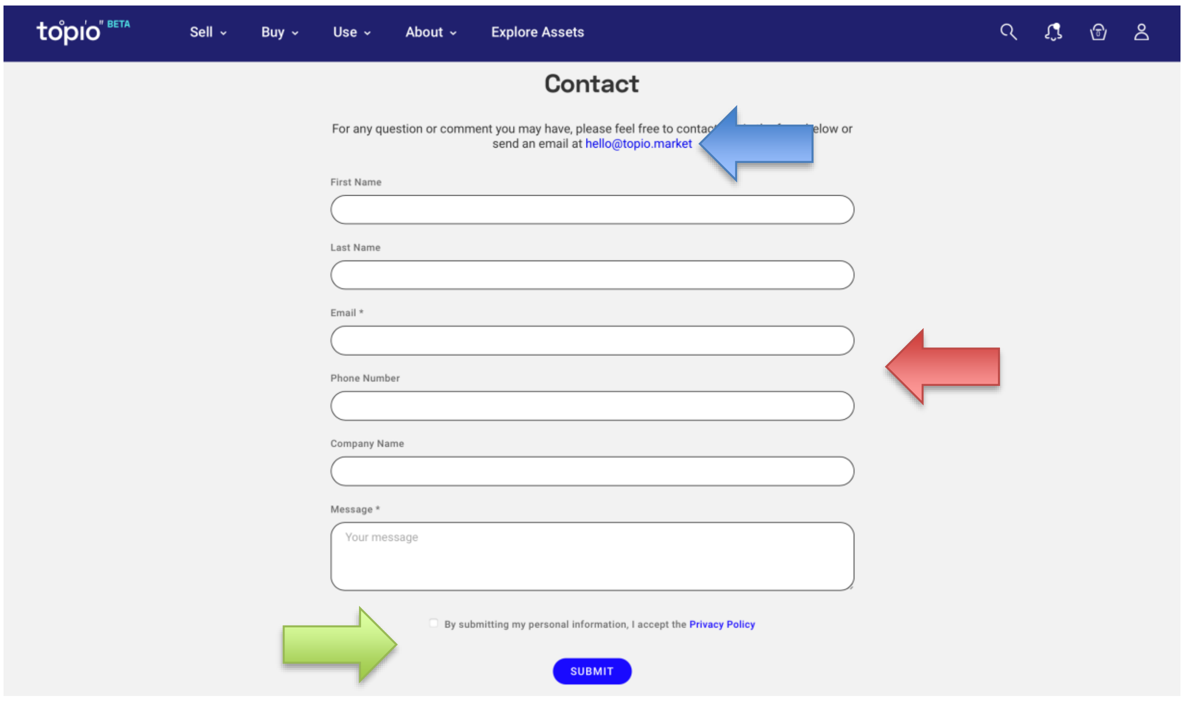
© 2020 · Opertus Mundi • This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 870228

Figure 22: Footer - Access the FAQ or Blog pages

At the bottom of the landing page (Figure 22) lies the footer of Topio, containing useful links and information regarding selling, buying, and using geospatial data assets, as well as links to the terms of service and privacy policy of the platform. There are also links to the "Contact", "FAQ" and "Blog" pages.

2.1.1.1. Contact

To visit the "Contact" page, the user must click on the corresponding link at the footer of Topio (green arrow in Figure 22). She is then redirected to the page depicted in Figure 23. The "Contact" page contains a form to be filled by the user in case she desires to communicate with the Topio.



For any question or comment you may have, please feel free to contact … below or send an email at hello@topio.market

By submitting my personal information, I accept the Privacy Policy

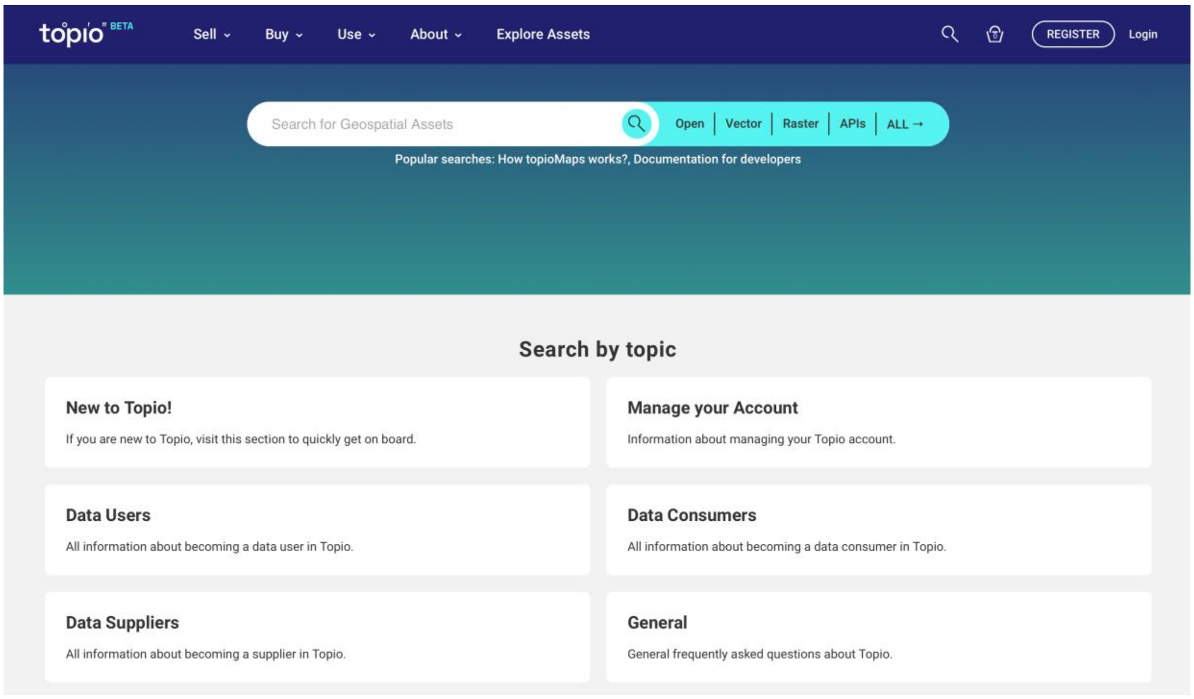
Figure 23: The contact page

2.1.1.2. FAQ

To visit the "FAQ" page, the user must click on the corresponding link at the footer of Topio (blue arrow in Figure 22). She is then redirected to the page depicted in Figure 24. The "FAQ" page contains 6 categories containing frequently asked questions, specifically:

- "New to Topio", containing useful tips for the new users,
- "Manage your Account", answering account-related questions,
- "Data Users", providing information about becoming a data user in Topio
- "Data Consumers", providing information about becoming a data consumer in Topio,
- "Data Suppliers", providing information about becoming a data supplier in Topio, and
- "General", containing answers to more general frequently asked questions.

Clicking on one of the categories, the user is redirected to the corresponding page containing all related questions and answers. As an example, Figure 25 illustrates the "Data Suppliers" FAQ page. If available, each question may contain videos that demonstrate the corresponding functionality, as shown in the figure.



Popular searches: How topioMaps works?, Documentation for developers

Search by topic

New to Topio!
If you are new to Topio, visit this section to quickly get on board.

Manage your Account
Information about managing your Topio account.

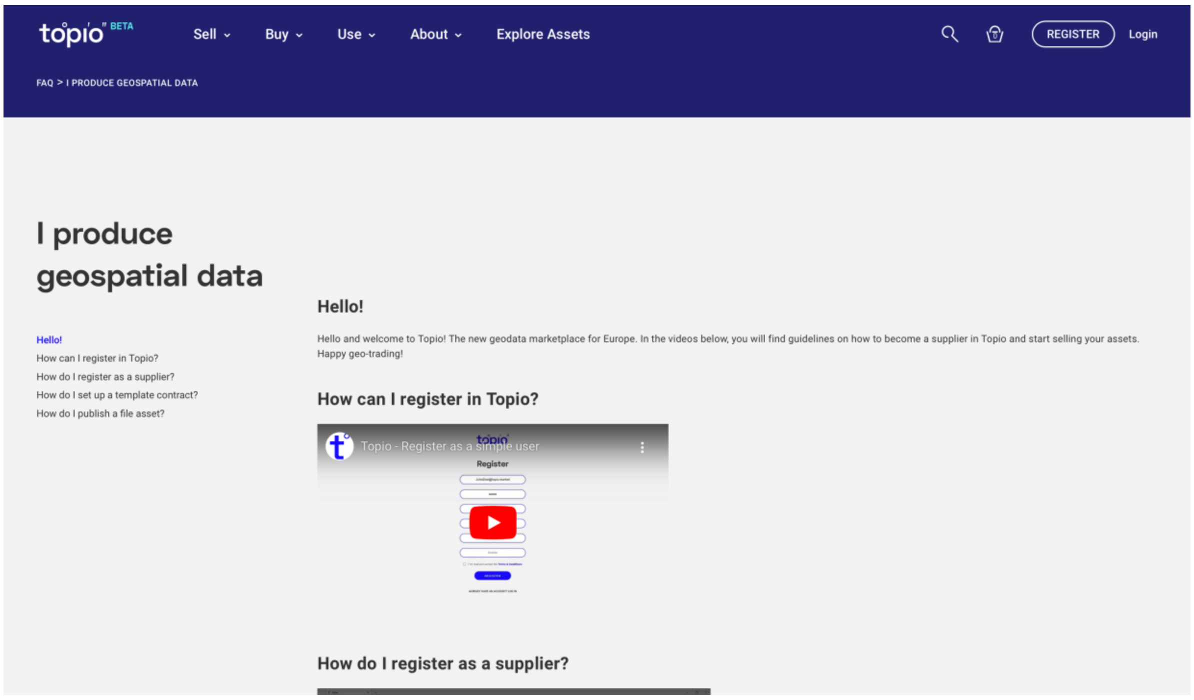
Data Users
All information about becoming a data user in Topio.

Data Consumers
All information about becoming a data consumer in Topio.

Data Suppliers
All information about becoming a supplier in Topio.

General
General frequently asked questions about Topio.

Figure 24: The FAQ page



FAQ > I PRODUCE GEOSPATIAL DATA

I produce geospatial data

Hello!
How can I register in Topio?
How do I register as a supplier?
How do I set up a template contract?
How do I publish a file asset?

Hello!

Hello and welcome to Topio! The new geodata marketplace for Europe. In the videos below, you will find guidelines on how to become a supplier in Topio and start selling your assets. Happy geo-trading!

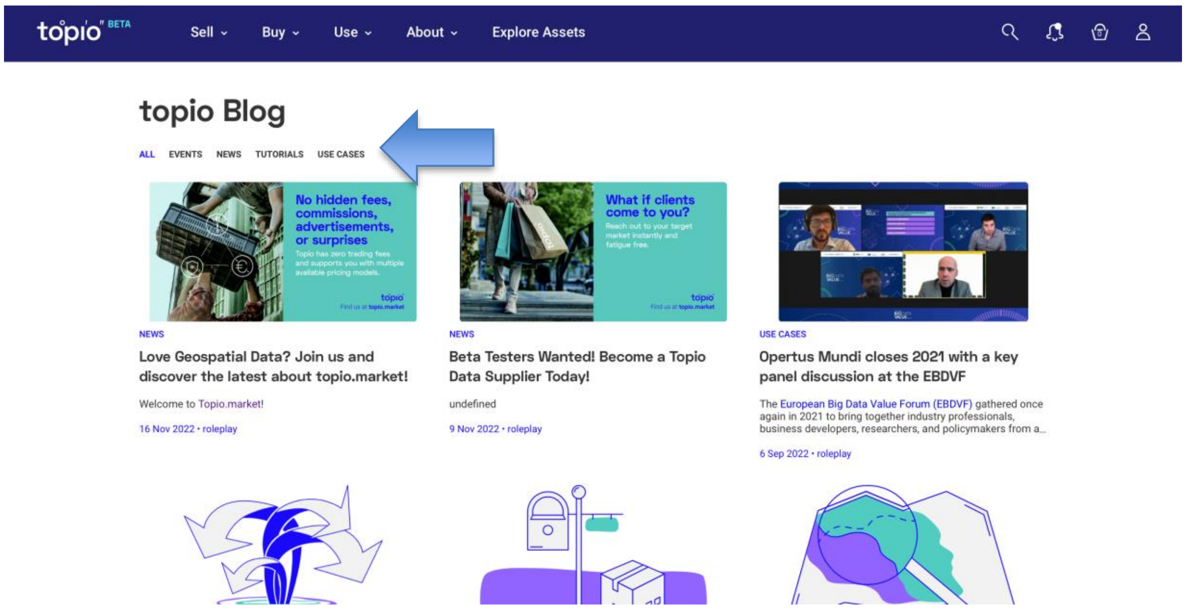
How can I register in Topio?

How do I register as a supplier?

Figure 25: FAQ – Data Suppliers

2.1.1.3. Blog

To visit the "Blog" page, a user must click on the corresponding link at the footer of Topio (red arrow in Figure 24). She is then redirected to the page depicted in Figure 26. The "Blog" page contains 4 categories of news posts regarding Topio (blue arrow in Figure 26). Each blog entry is consisted of an image, its category, a title, a short description and the date it was posted. Clicking on the title of a post, redirects to its page (Figure 27).



topio Blog

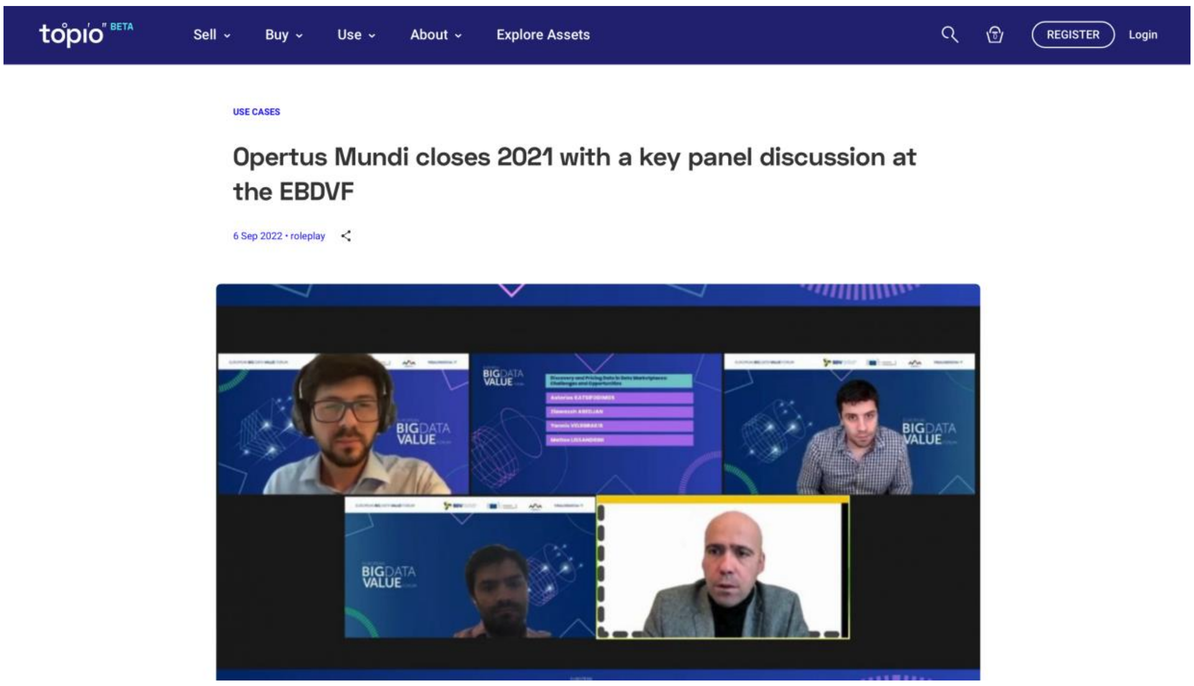
NEWS
Love Geospatial Data? Join us and discover the latest about topio.market!
Welcome to Topio.market!
16 Nov 2022 • roleplay

NEWS
Beta Testers Wanted! Become a Topio Data Supplier Today!
undefined
9 Nov 2022 • roleplay

USE CASES
Opertus Mundi closes 2021 with a key panel discussion at the EBDVF
The European Big Data Value Forum (EBDVF) gathered once again in 2021 to bring together industry professionals, business developers, researchers, and policymakers from a...
6 Sep 2022 • roleplay

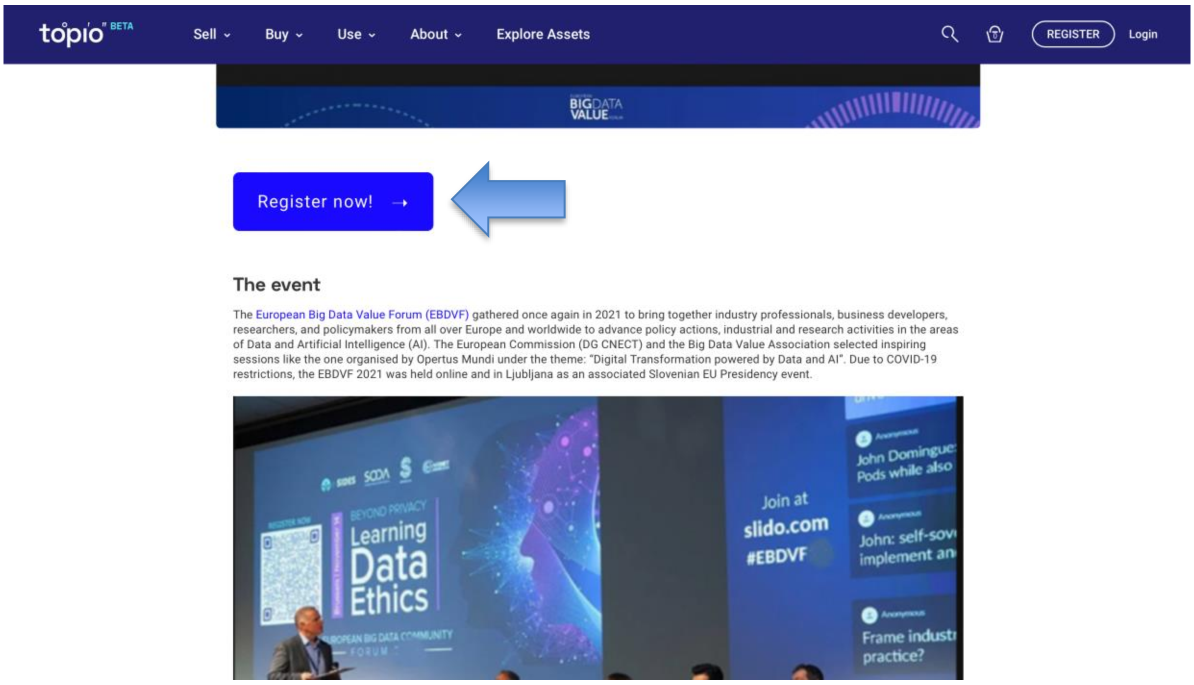
Figure 26: The Blog page

A post's page may contain its category, title, and publication date at the top. The post itself may contain images and text, as well as buttons to register to Topio (blue arrow in Figure 28). At the bottom of a post's page, there is a link that prompts to register to Topio (blue arrow in Figure 29), links to next and previous posts (red arrow in Figure 29) and a list of related blog posts (green arrow in Figure 29).



USE CASES
Opertus Mundi closes 2021 with a key panel discussion at the EBDVF
6 Sep 2022 • roleplay

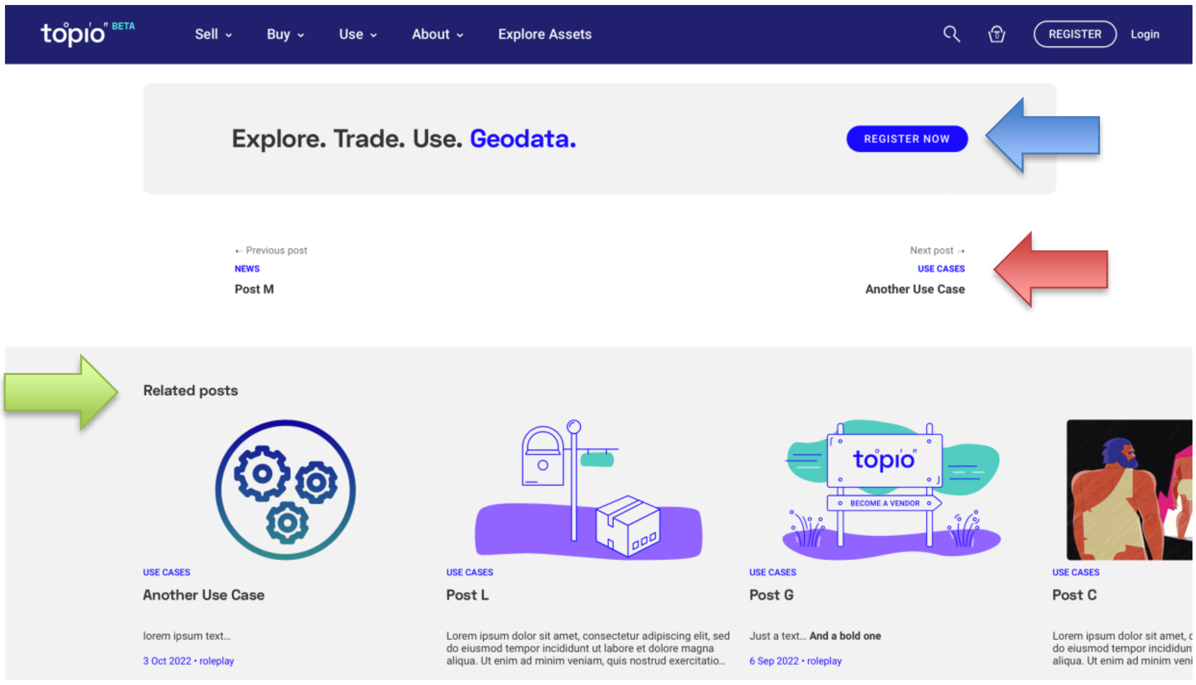
Figure 27: A post's page



The event

The European Big Data Value Forum (EBDVF) gathered once again in 2021 to bring together industry professionals, business developers, researchers, and policymakers from all over Europe and worldwide to advance policy actions, industrial and research activities in the areas of Data and Artificial Intelligence (AI). The European Commission (DG CNECT) and the Big Data Value Association selected inspiring sessions like the one organised by Opertus Mundi under the theme: "Digital Transformation powered by Data and AI". Due to COVID-19 restrictions, the EBDVF 2021 was held online and in Ljubljana as an associated Slovenian EU Presidency event.

Figure 28: Scrolling down a post's page



USE CASES
Another Use Case
lorem ipsum text...
3 Oct 2022 • roleplay

USE CASES
Post L
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitatio...

USE CASES
Post G
Just a text... **And a bold one**
6 Sep 2022 • roleplay

USE CASES
Post C
Lorem ipsum dolor sit amet, c
do eiusmod tempor incididun
aliqua. Ut enim ad minim veni

Figure 29: The bottom of a post's page

2.1.2. Sell

As depicted in Figure 30, on the top of the landing page, there is a horizontal bar containing links to the core Topio functionalities. This bar stays visible at the top while the user scrolls down the landing page. By clicking on the "Sell" link, a drop-down box is shown, informing the user of the benefits of becoming a supplier in Topio, along with a "Become a Supplier" button (which appears in various other areas of the home page). By clicking on that button, the user is redirected to the "Register as a Supplier" page (see next section).

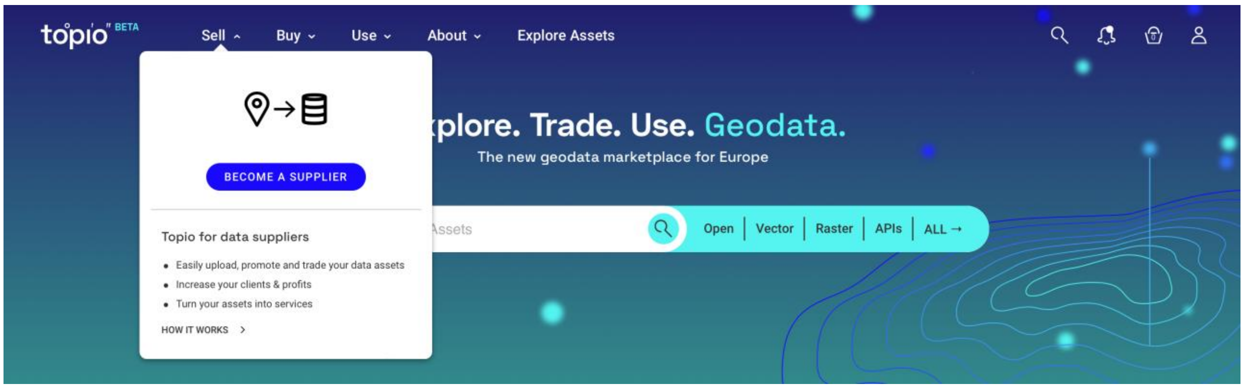

topio BETA
Sell
Buy
Use
About
Explore Assets
BECOME A SUPPLIER
Topio for data suppliers
Easily upload, promote and trade your data assets
Increase your clients & profits
Turn your assets into services
HOW IT WORKS
xplore. Trade. Use. Geodata.
The new geodata marketplace for Europe
Assets
Open
Vector
Raster
APIs
ALL →

Figure 30: Sell menu

To access more information regarding the benefits of becoming a supplier, the user can click on the "How it Works" button of the drop-down box, which redirects to a corresponding informative page, containing more details, as illustrated in Figure 31.

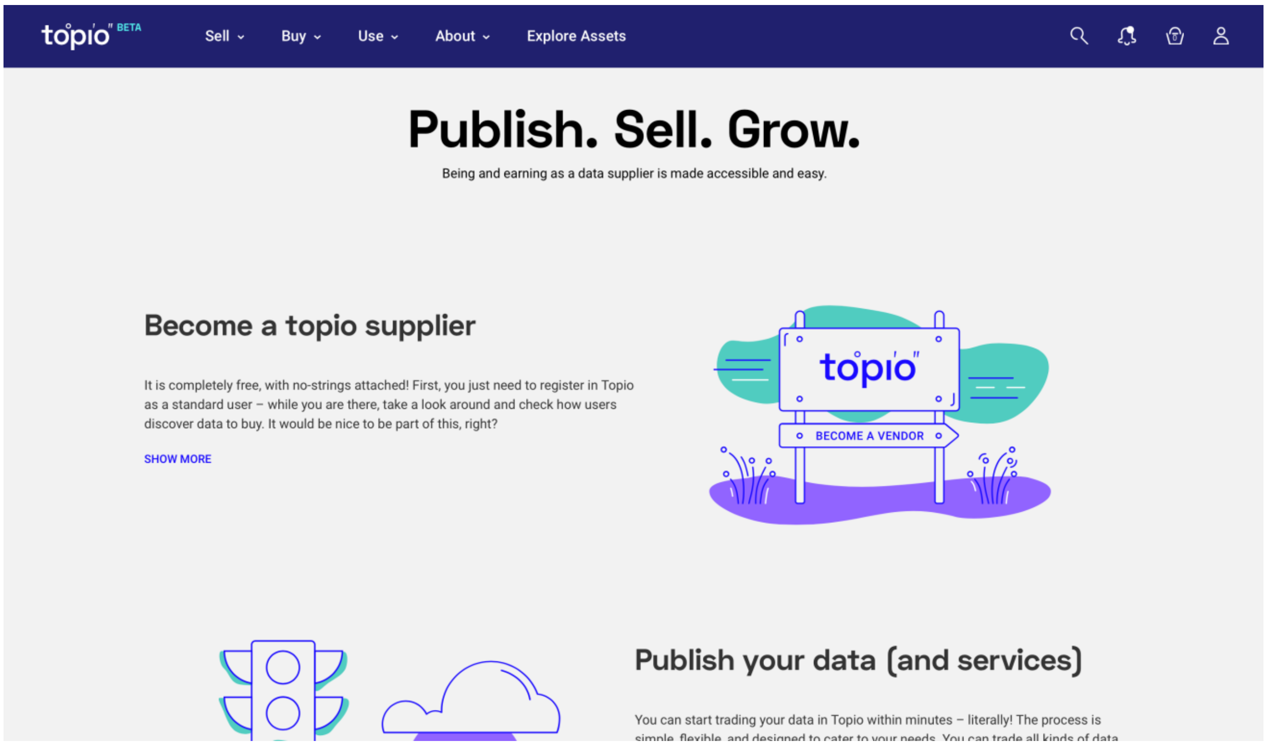
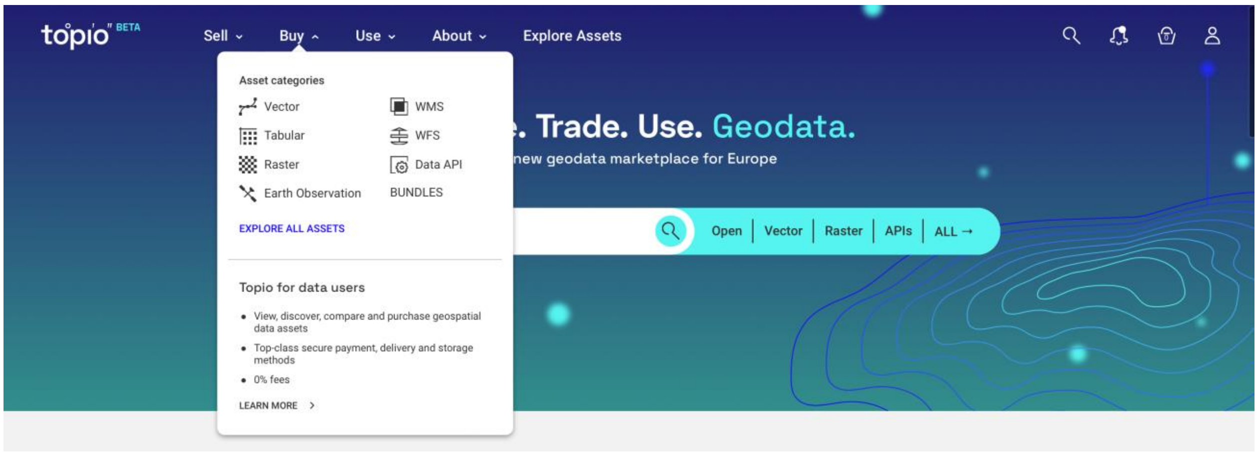

topio BETA
Sell
Buy
Use
About
Explore Assets
Publish. Sell. Grow.
Being and earning as a data supplier is made accessible and easy.
Become a topio supplier
It is completely free, with no-strings attached! First, you just need to register in Topio as a standard user – while you are there, take a look around and check how users discover data to buy. It would be nice to be part of this, right?
SHOW MORE
topio
BECOME A VENDOR
Publish your data (and services)
You can start trading your data in Topio within minutes – literally! The process is

Figure 31: Benefits for sellers

2.1.3. Buy

The user can also have access to the "Buy" menu (accessible via the link next to the "Sell" menu), as shown in Figure 32. It is consisted of a list of benefits of becoming a geospatial data buyer and user, along with a list of the asset types offered by the platform. By clicking on the "Explore All Assets" link, the user is redirected to the asset exploration page, which will be covered in a following subsection.

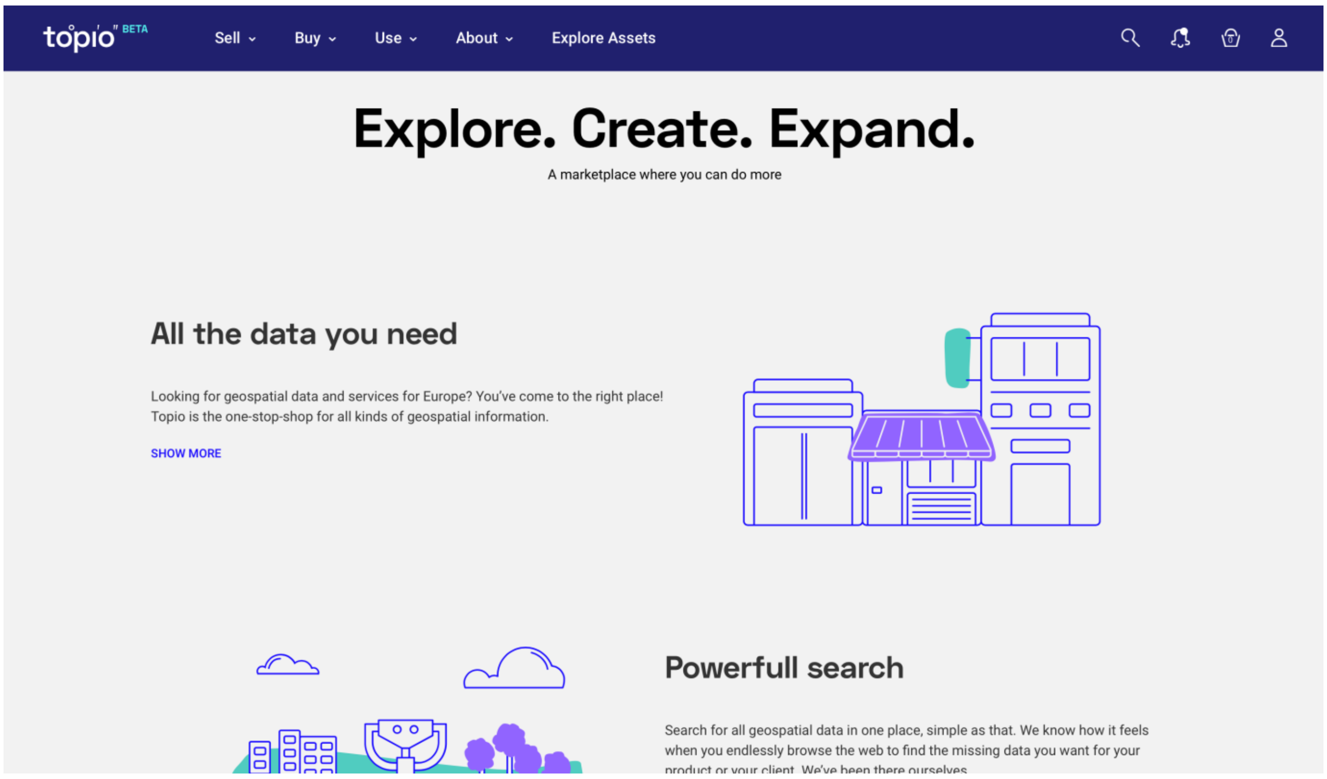


Topio for data users

- View, discover, compare and purchase geospatial data assets
- Top-class secure payment, delivery and storage methods
- 0% fees

Figure 32: Buy asset menu

The user can click on the “Learn More” button to be redirected in the corresponding informative page containing more details regarding the benefits of becoming a data consumer and user in the Topio platform (Figure 33).



Explore. Create. Expand.

A marketplace where you can do more

All the data you need

Looking for geospatial data and services for Europe? You've come to the right place! Topio is the one-stop-shop for all kinds of geospatial information.

Powerfull search

Search for all geospatial data in one place, simple as that. We know how it feels when you endlessly browse the web to find the missing data you want for your product or your client. We've been there ourselves.

Figure 33: Benefits for consumers

2.1.4. Use

By clicking on the “Use” button (Figure 34), a user can view the various Value-Added Services (VAS) offered by the platform. She can access the VAS either per category (i.e., via API, Notebooks and Maps), or choose to list all available services via the “Explore All Services” button (arrow in Figure 34). Doing so, will redirect the user to the VAS page (Figure 35), where the user can view details regarding each VAS by clicking on the “View Service” button, which becomes visible upon hovering the mouse over each VAS tile (arrow in Figure 35).

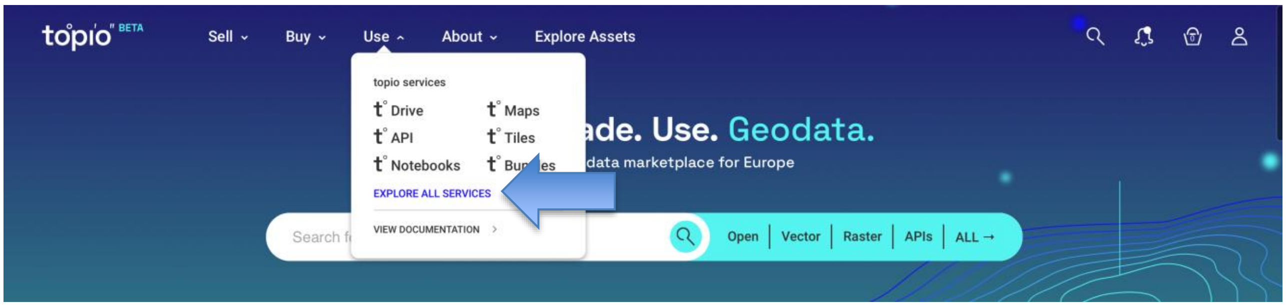


Figure 34: Use menu

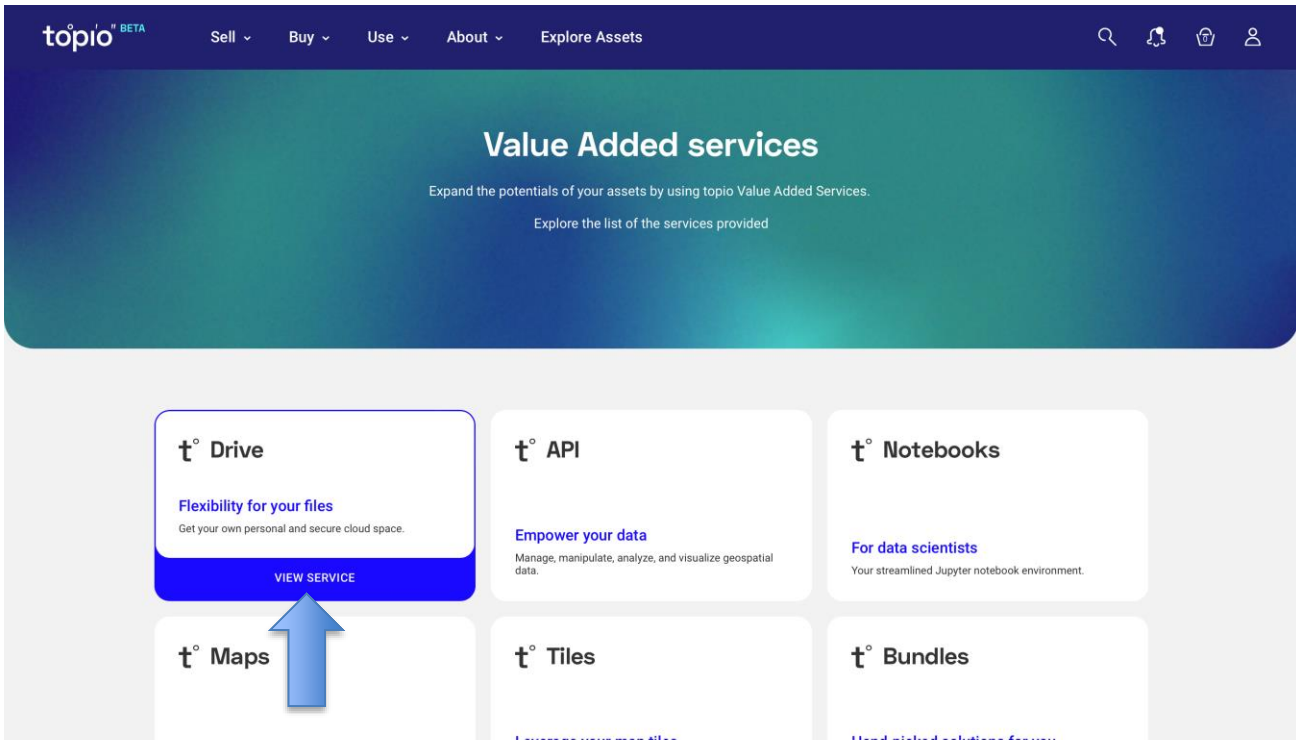

topio BETA
Sell
Buy
Use
About
Explore Assets
Value Added services
Expand the potentials of your assets by using topio Value Added Services.
Explore the list of the services provided
Drive
Flexibility for your files
Get your own personal and secure cloud space.
VIEW SERVICE
API
Empower your data
Manage, manipulate, analyze, and visualize geospatial data.
Notebooks
For data scientists
Your streamlined Jupyter notebook environment.
Maps
Tiles
Bundles

Figure 35: All VAS page

2.1.5. About

The horizontal bar (Figure 16) at the top of the home page also contains an “About” link. Hovering over the link reveals three different options: “What is Topio”, “Contact” and “FAQ” (Figure 36). Clicking on “What is Topio” redirects to a page containing more information regarding the platform, depicted in Figure 37. Clicking on “Contact” reveals our contact details in case the user desires to contact us and “FAQ” redirects to the FAQ page, described in Section 2.1.1.2.

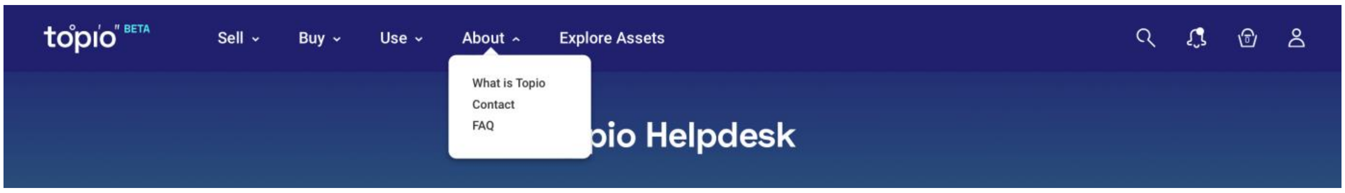
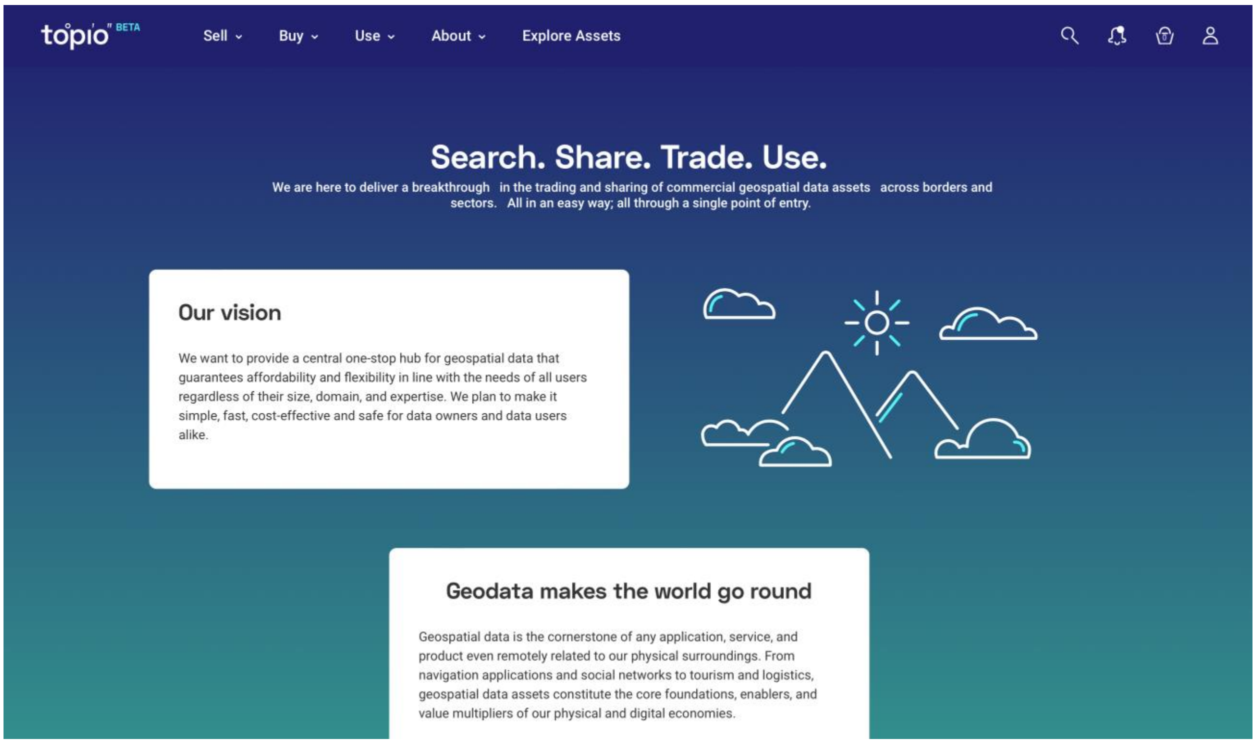


Figure 36: The About list



Search. Share. Trade. Use.

We are here to deliver a breakthrough in the trading and sharing of commercial geospatial data assets across borders and sectors. All in an easy way; all through a single point of entry.

Our vision

We want to provide a central one-stop hub for geospatial data that guarantees affordability and flexibility in line with the needs of all users regardless of their size, domain, and expertise. We plan to make it simple, fast, cost-effective and safe for data owners and data users alike.

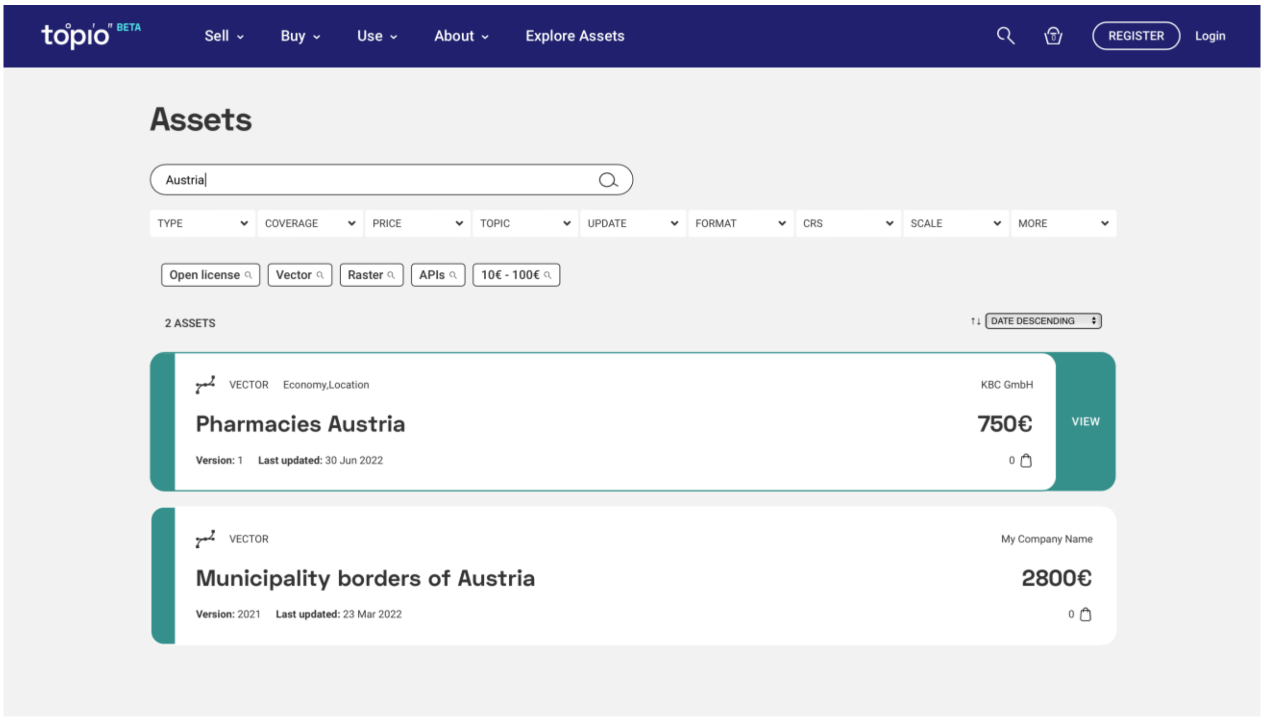
Geodata makes the world go round

Geospatial data is the cornerstone of any application, service, and product even remotely related to our physical surroundings. From navigation applications and social networks to tourism and logistics, geospatial data assets constitute the core foundations, enablers, and value multipliers of our physical and digital economies.

Figure 37: The What is Topio page

2.1.6. Explore

The final, right-most link of the horizontal toolbar of Topio's home page ("Explore Assets"), redirects the user to the asset exploration page (as mentioned above, the asset exploration page is also accessible via the "Buy" menu of the horizontal toolbar), illustrated in Figure 38. Here, the user is presented with all the available assets in the platform, in a paginated manner (Figure 38). The user can search, filter, and view any asset available in Topio. More information regarding searching and viewing assets is covered in Section 2.1.2.



Assets

Austria

Open license | Vector | Raster | APIs | 10€ - 100€

2 ASSETS

DATE DESCENDING

VECTOR Economy,Location — KBC GmbH
Pharmacies Austria — 750€
Version: 1 Last updated: 30 Jun 2022 — 0

VECTOR — My Company Name
Municipality borders of Austria — 2800€
Version: 2021 Last updated: 23 Mar 2022 — 0

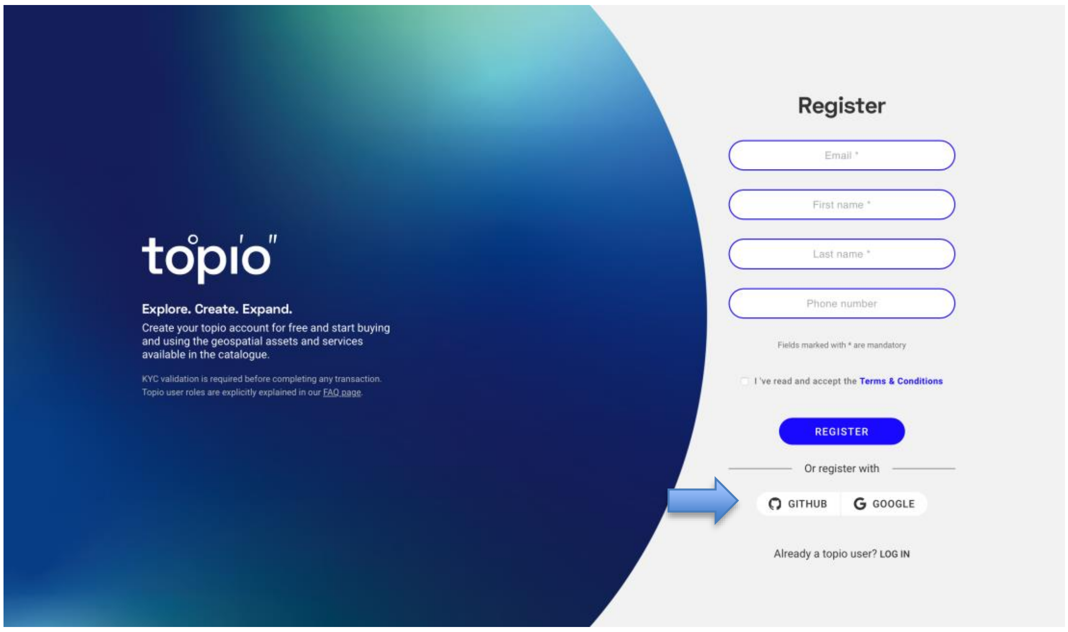
Figure 38: The asset exploration page

2.2. Register

2.2.1. Simple user

As mentioned in the previous section, by clicking on the "Register" link (Figure 16), the user can register to Topio. Doing so, the user will be redirected to the register page (Figure 39). A user can register as a *simple* user by completing the following steps.

To open a simple user account in Topio, the user must only provide only her email, first and last name (optionally also her phone number), as depicted in Figure 39. The user has the option to register via an external identity provider, such as Google and GitHub (blue arrow in Figure 39). In that case, the corresponding fields obtained from the provider are automatically filled. The user then must agree with the platform's terms and conditions and finally click on the "Register" button. She then receives an email with a link to verify her email. After the email verification, the user receives a follow up email notifying her about the account activation. The email also contains a one-time password which must be used during the first sign-in to the platform. During the first sign-in, the platform prompts the user to change her password, as indicated in Figure 40. The user is now registered to Topio and can view the automated metadata generated for geospatial assets.


topio

Explore. Create. Expand.

Create your topio account for free and start buying and using the geospatial assets and services available in the catalogue.

KYC validation is required before completing any transaction. Topio user roles are explicitly explained in our FAQ page.

Register

Email *

First name *

Last name *

Phone number

Fields marked with * are mandatory

I 've read and accept the Terms & Conditions

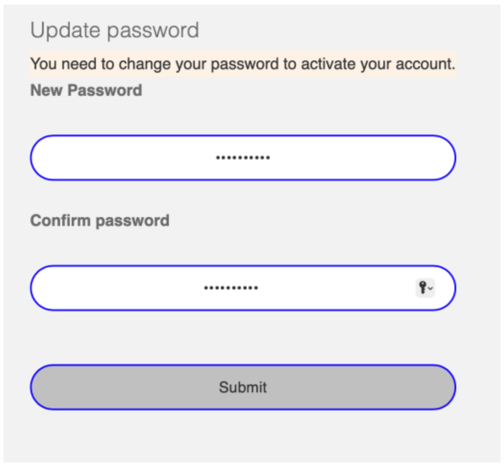
REGISTER

Or register with

GITHUB GOOGLE

Already a topio user? LOG IN

Figure 39: Register page


Update password

You need to change your password to activate your account.

New Password

••••••••••

Confirm password

••••••••••

Submit

Figure 40: The user must reset the one-time password

To purchase, or sell assets, the user will have to register at a later point as a *consumer* or *supplier*. These cases are covered in the next two subsections.

2.2.2. Consumer

As already mentioned, when a user creates an account on Topio, she is registered as a simple user, i.e., she can browse through the platform and explore its geospatial assets. If, at some point, she desires to purchase an asset, she must become an asset consumer. This process can be initiated either when she first attempts to make a purchase, or manually via her "Dashboard", or "Home Page". Figure 41 illustrates the "Dashboard" page, after the initial registration. To become a consumer, the user must click on the "Become a Consumer" button (blue arrow in Figure 41). Doing so, triggers the "Become a Consumer" wizard that guides the user through the registration. The wizard steps are represented by tabs, to ease navigation. The tabs are placed on the top, and the current tab is annotated with blue color. At the first step of the wizard, the user must select whether the account is individual or professional (see Figure 42). Next, she must provide additional information regarding her, or her organization, in case the account is professional. Of course, the already provided information during initial registration is filled-in (see Figure 43). The final step of the wizard (Figure 44) contains a summary of the provided information. To finalize the consumer registration, the user must click on the "Confirm and Create Account" button.

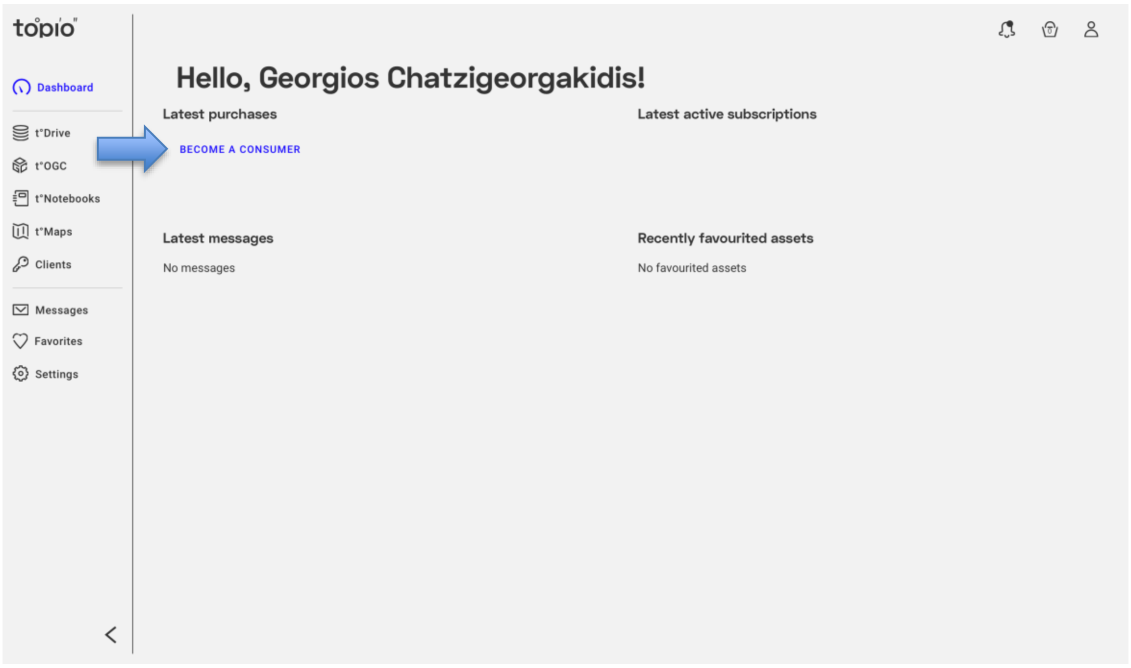


Figure 41: The Dashboard

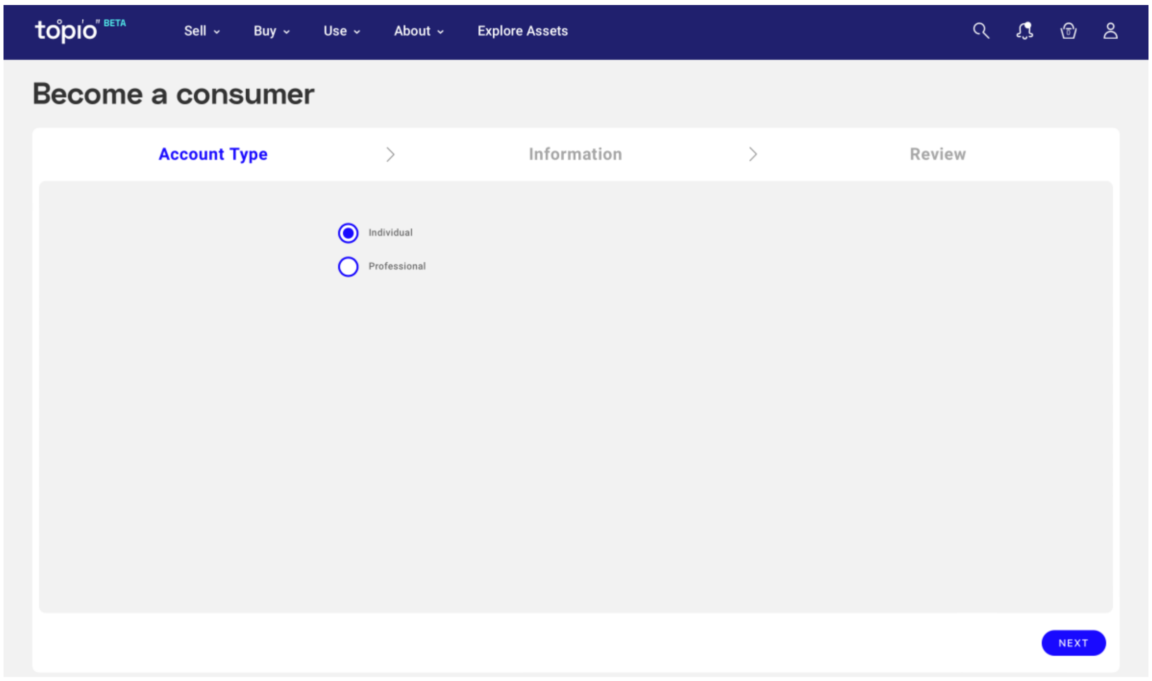
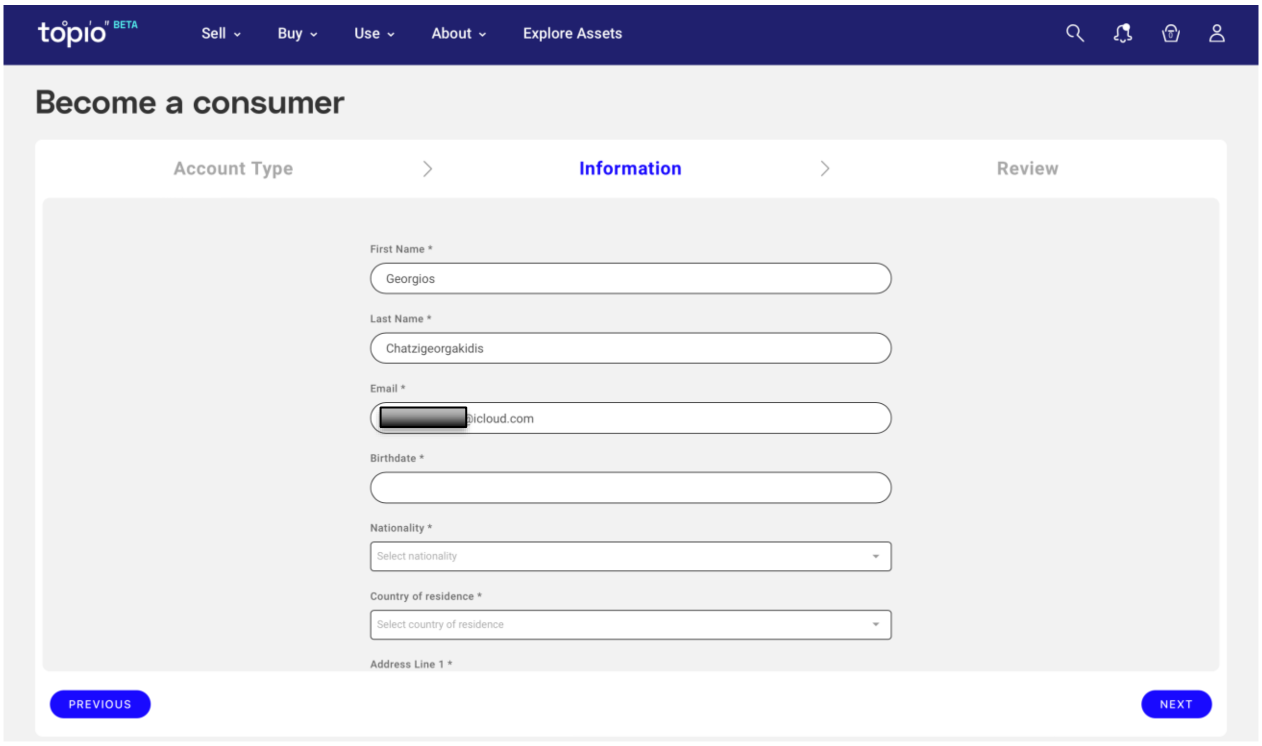


Figure 42: First step of the become a consumer wizard



First Name *
Georgios

Last Name *
Chatzigeorgakidis

Email *
[redacted]@icloud.com

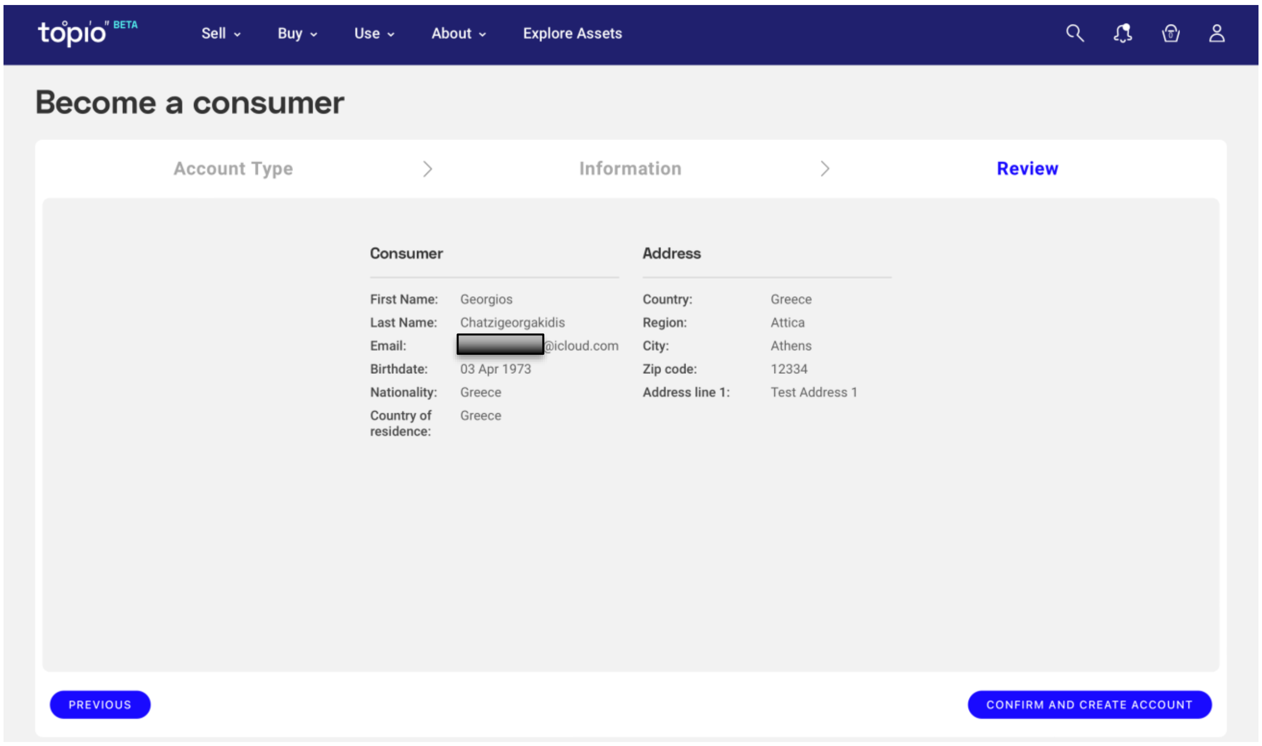
Birthdate *

Nationality *
Select nationality

Country of residence *
Select country of residence

Address Line 1 *

Figure 43: Second step of the become a consumer wizard



Consumer		Address	
First Name:	Georgios	Country:	Greece
Last Name:	Chatzigeorgakidis	Region:	Attica
Email:	[redacted]@icloud.com	City:	Athens
Birthdate:	03 Apr 1973	Zip code:	12334
Nationality:	Greece	Address line 1:	Test Address 1
Country of residence:	Greece		

Figure 44: Third step of the become a consumer wizard

To be able to make purchases, the user must later provide her KYC/UBO information, as well as any credit or debit cards she desires[60]. These processes are detailed in Section 2.8.11. After becoming a consumer, the user can visit her dashboard to access the additional functionality (Figure 45). There, she is reminded about her pending KYC/UBO, as shown with a blue arrow in the figure. She can click on the "Go to Settings" button to visit the corresponding KYC/UBO information provision page, or on "I'll do it later" to dismiss the reminder.

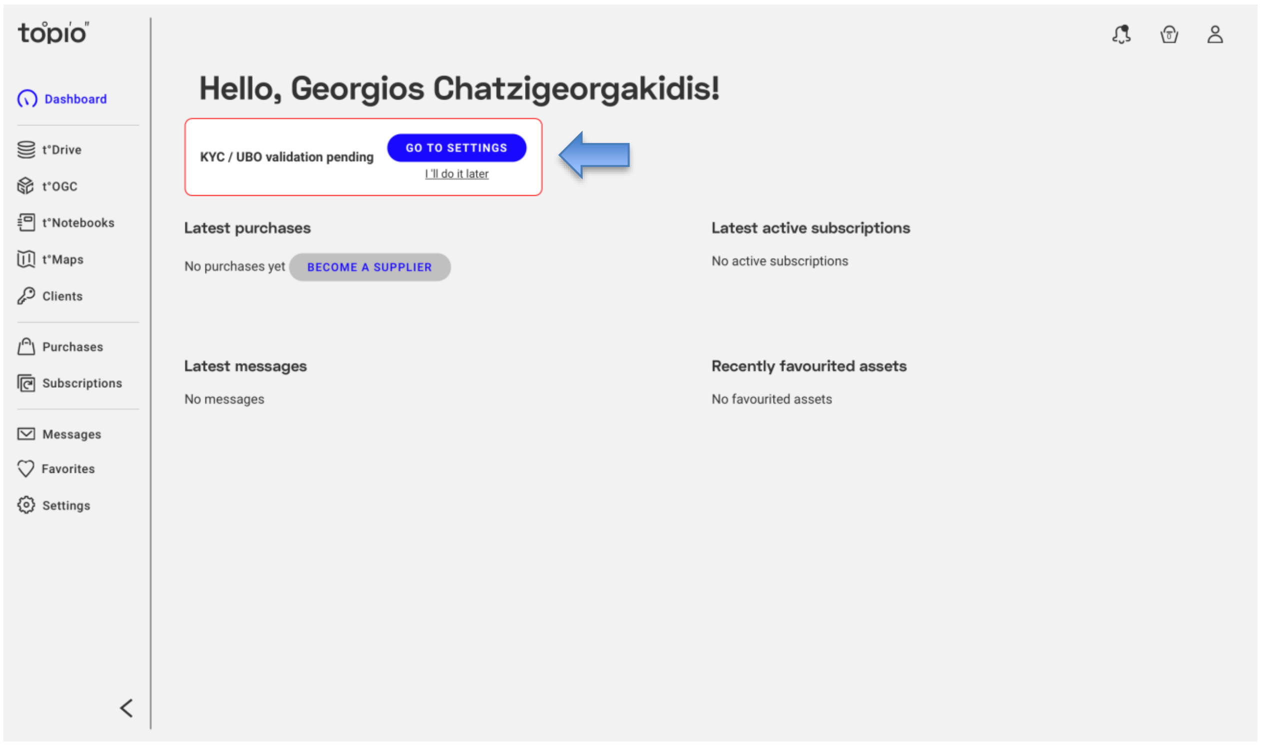


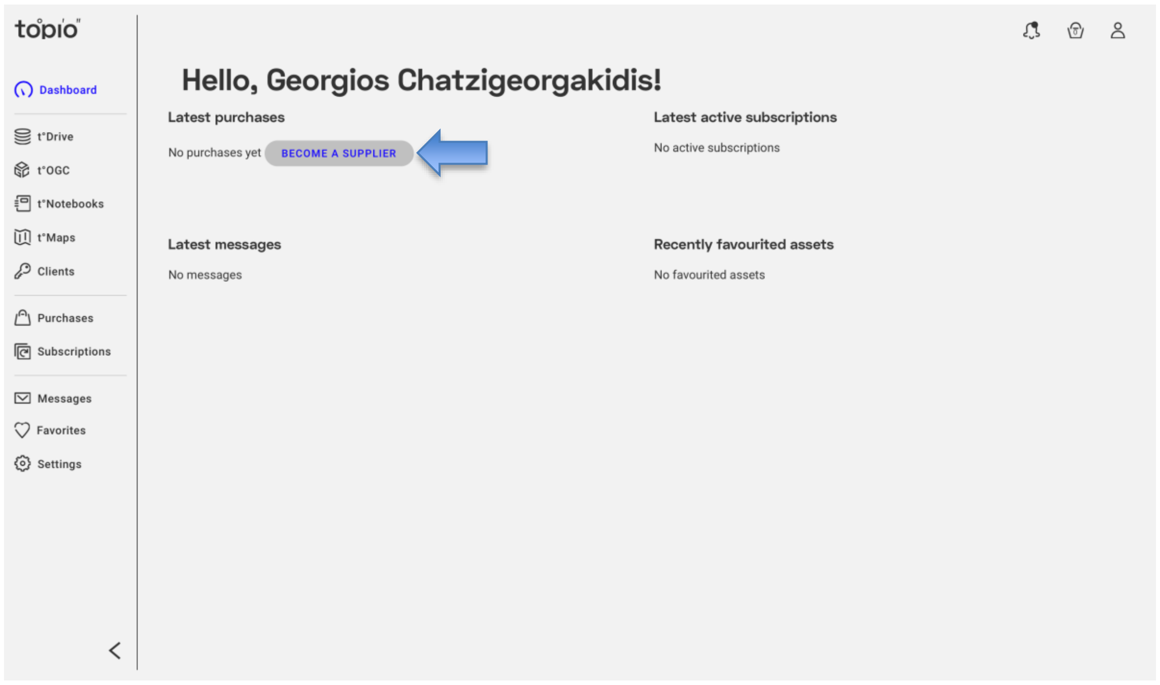
Figure 45: KYC/UBO validation reminder for consumers

2.2.3. Asset supplier

If a user wants to become a geospatial asset supplier, either via her "Dashboard" page after she registered as a consumer (blue arrow in Figure 46), or straight from the landing page, she can complete the process as follows.

Initially, the user is transferred to the "Become a Supplier" wizard, depicted in Figure 47. The wizard consists of 4 steps, guiding the prospective supplier through the registration process. At the first page of the wizard ("Company"), the user must provide some basic information regarding her company, such as VAT number, name, website, address, country, region, city, and ZIP code. The user then must press the "Next" button on the bottom right, to move to the second page of the wizard.

[60] Please note that due to the legal framework establishing the operation of marketplaces in EU (see D3.1 for details), all asset consumers in Topio are considered (and hence must register) as legal entities, with KYC/KYB processes in place to guarantee the regulatory conformance of the marketplace (e.g., anti-laundering)


topio

Dashboard
t°Drive
t°OGC
t°Notebooks
t°Maps
Clients
Purchases
Subscriptions
Messages
Favorites
Settings

Hello, Georgios Chatzigeorgakidis!

Latest purchases

No purchases yet BECOME A SUPPLIER

Latest active subscriptions

No active subscriptions

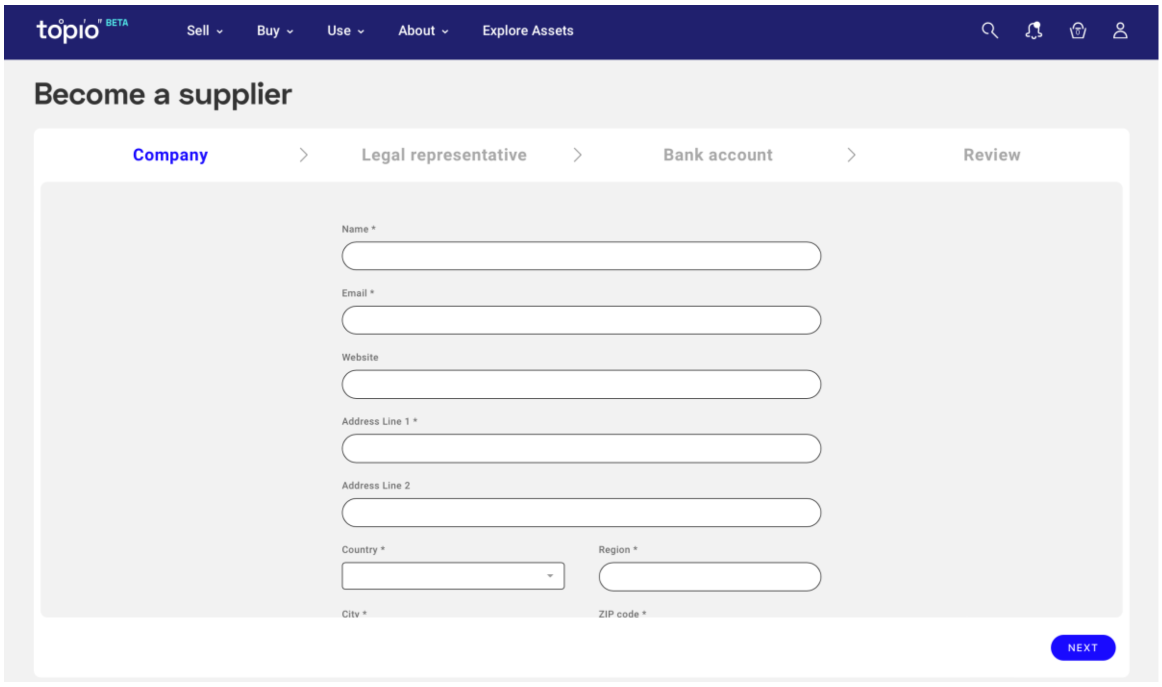
Latest messages

No messages

Recently favourited assets

No favourited assets

Figure 46: Initiate the become a supplier wizard


topio BETA Sell Buy Use About Explore Assets

Become a supplier

Company > Legal representative > Bank account > Review

Name *

Email *

Website

Address Line 1 *

Address Line 2

Country *

Region *

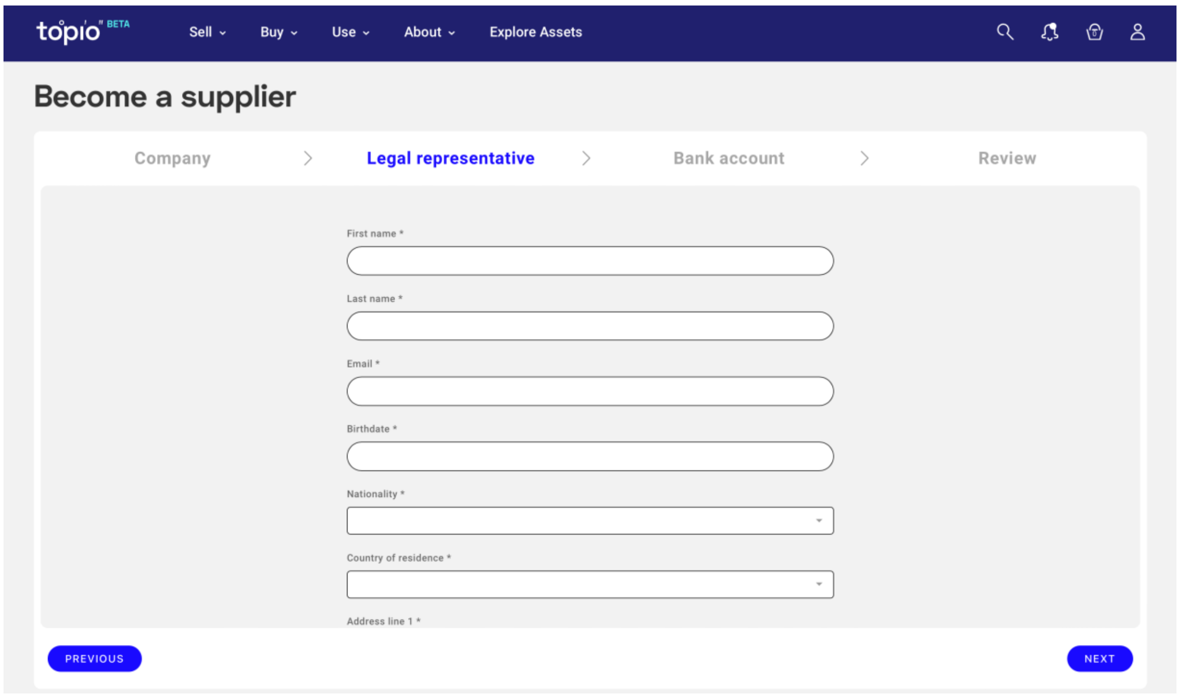
City *

ZIP code *

NEXT

Figure 47: Become a supplier wizard (step 1)

At the second page of the wizard (Figure 48), the user must provide the personal details of the suppliers' legal representative. These include, first and last name, email, date of birth, nationality, country of residence, address, city, region, ZIP code, and country of origin. All fields are mandatory. Again, to move to the next page of the wizard, the user must click on the "Next" button. She can always go back to the previous page by clicking the “Previous” button at the bottom left of the page.



Become a supplier

Company > **Legal representative** > Bank account > Review

First name *

Last name *

Email *

Birthdate *

Nationality *

Country of residence *

Address line 1 *

Figure 48: Become a supplier wizard (step 2).

At the next page (Figure 49), the prospective supplier must fill-in the details of her (or her company’s) bank account, where she will receive the payments for the geospatial assets she sells via the platform. Such details include legal account holder name, IBAN, BIC, address, city, region, ZIP code, and country. Again, all fields are mandatory. To move to the next page, the user must click on “Next” at the bottom right of the page (or “Previous”, if desired).

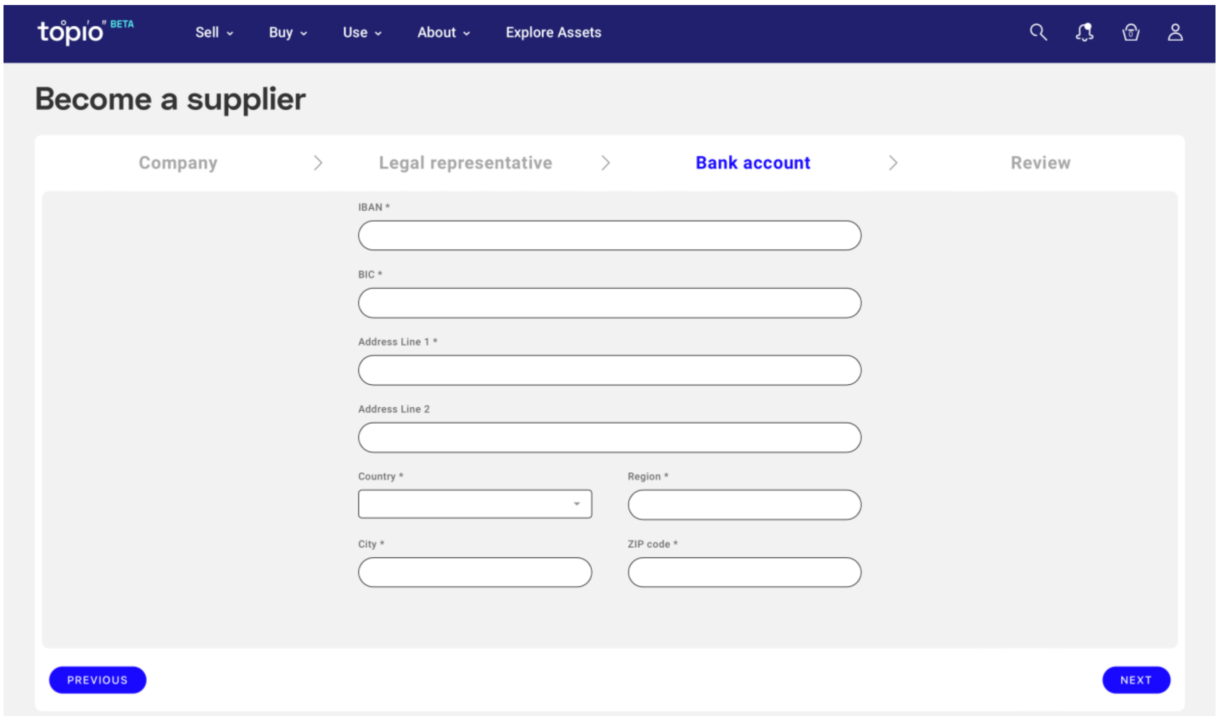
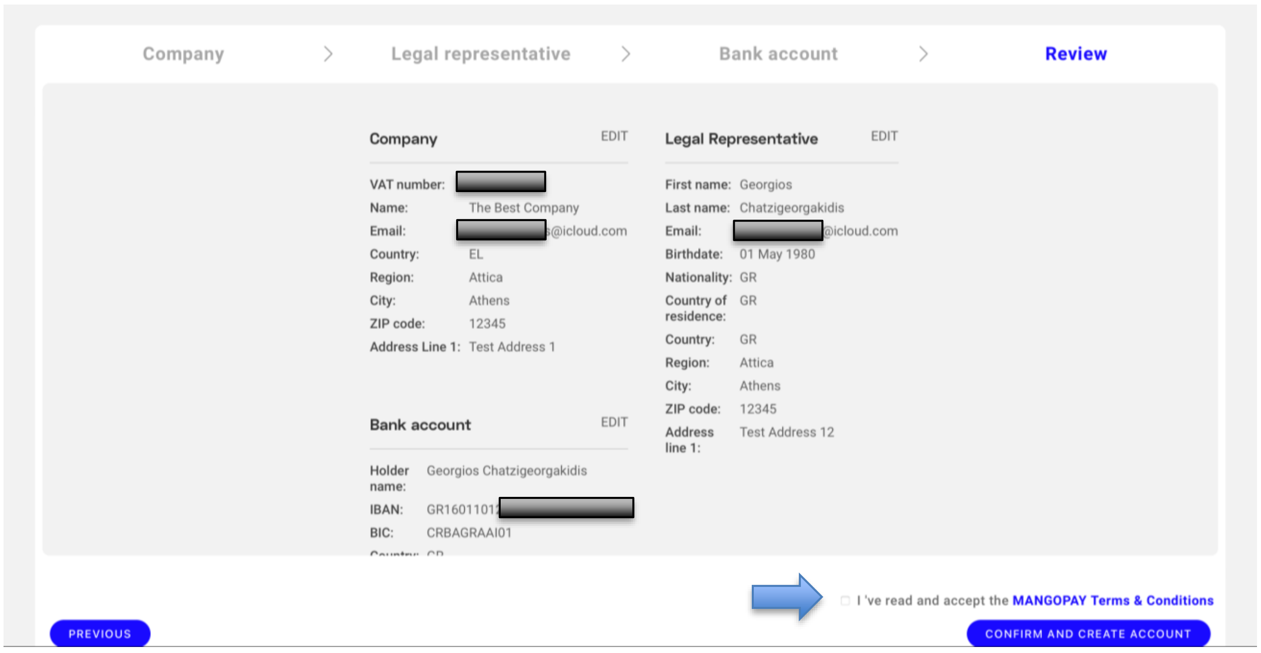


Figure 49: Become a supplier wizard (step 3)

Finally, moving to the last wizard page (Figure 50), the user is presented with a summary of all the information she has provided since she started the registration process. If she desires to alter some of the information, she can click on the “Previous” button on the bottom left. To finalize the supplier registration, after accepting the MANGOPAY terms and conditions (blue arrow in Figure 50), the user must click on the “Confirm and Create Account” button. Then, the supplier registration process is finalized, the corresponding backend workflow that carries out the registration is initiated, and the user is presented with a corresponding informative screen (Figure 51).

As in the case of consumers, to be able to offer assets for sale via Topio, the supplier must later provide her KYC/KYB information, as described in 2.8.11. After becoming a supplier, the user can visit her dashboard to access the additional functionality (Figure 52). There, she is reminded about her pending KYC/UBO, as shown with a blue arrow in the figure. She can click on the “Go to Settings” button to visit the corresponding KYC/UBO information provision page, or on “I’ll do it later” to dismiss the reminder. To speed the asset publishing process, the new supplier can start publishing assets right after completing the “Become a Supplier” wizard, even before the KYC/UBO information provision. In that case, the published assets are available to prospective consumers, with the difference that they are now added to the consumers’ wishlist, available through the “Favorites” tab via her dashboard. Once the supplier is KYB validated, the consumers are notified that the assets in their wishlists are available.



Company > Legal representative > Bank account > Review

Company	EDIT
VAT number:	[redacted]
Name:	The Best Company
Email:	[redacted]@icloud.com
Country:	EL
Region:	Attica
City:	Athens
ZIP code:	12345
Address Line 1:	Test Address 1

Legal Representative	EDIT
First name:	Georgios
Last name:	Chatzigeorgakidis
Email:	[redacted]@icloud.com
Birthdate:	01 May 1980
Nationality:	GR
Country of residence:	GR
Country:	GR
Region:	Attica
City:	Athens
ZIP code:	12345
Address line 1:	Test Address 12

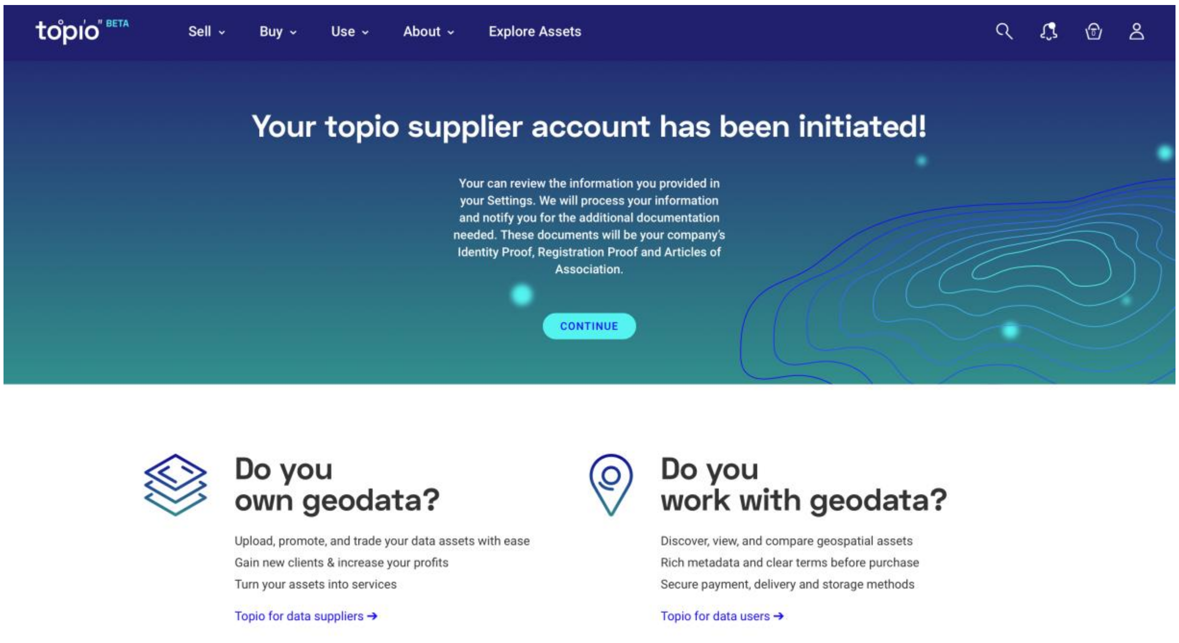
Bank account	EDIT
Holder name:	Georgios Chatzigeorgakidis
IBAN:	GR16011012[redacted]
BIC:	CRBAGRAAI01
Country:	GR

☐ I 've read and accept the **MANGOPAY Terms & Conditions**

PREVIOUS

CONFIRM AND CREATE ACCOUNT

Figure 50: Become a supplier wizard (step 4)



Your topio supplier account has been initiated!

Your can review the information you provided in your Settings. We will process your information and notify you for the additional documentation needed. These documents will be your company's Identity Proof, Registration Proof and Articles of Association.

CONTINUE

Do you own geodata?

Upload, promote, and trade your data assets with ease
Gain new clients & increase your profits
Turn your assets into services

Topio for data suppliers →

Do you work with geodata?

Discover, view, and compare geospatial assets
Rich metadata and clear terms before purchase
Secure payment, delivery and storage methods

Topio for data users →

Figure 51: Finished vendor registration

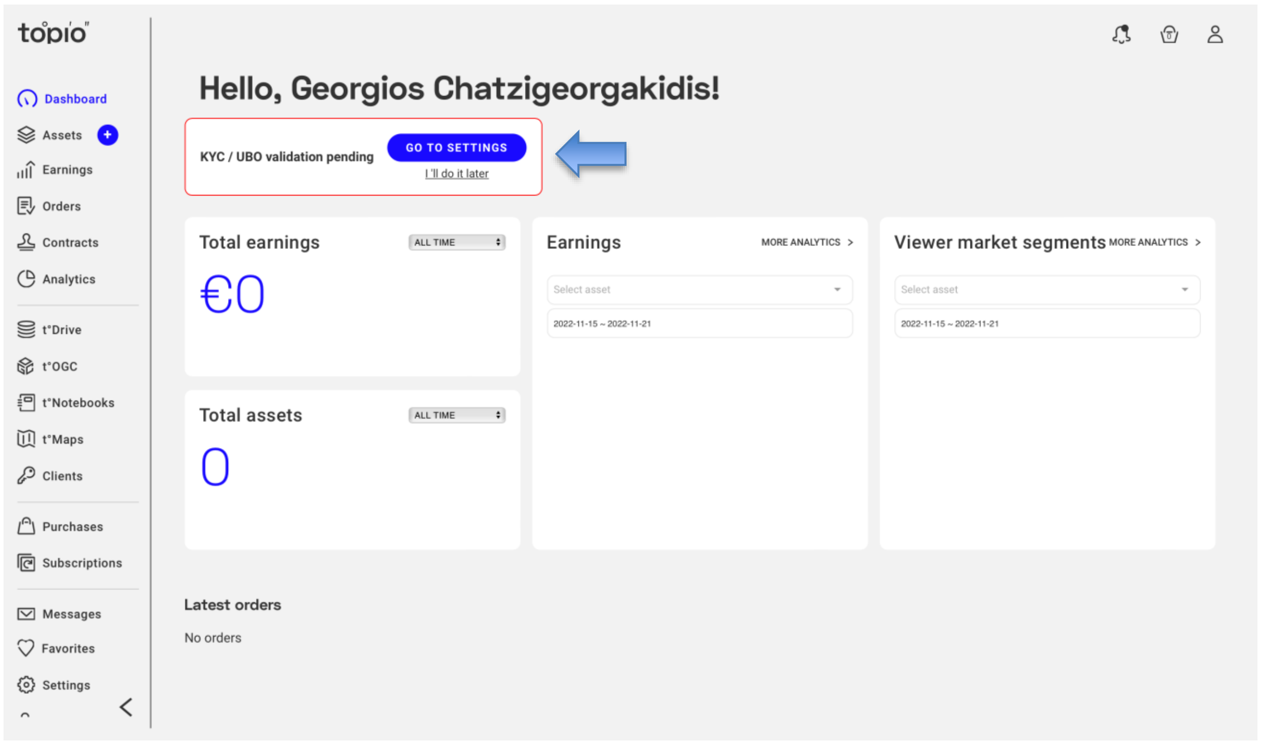


Figure 52: KYC/UBO validation reminder for suppliers

2.3. Asset Search

As mentioned in Section 2.1, from the landing page of the marketplace a user can search for geospatial data assets through the "*Search for Geospatial Assets*" bar (**Figure 16**). This search bar combines two search operations concerning *assets* and terms (i.e., *keywords*) characterizing the marketed assets. After the user specifies terms and hits *Enter* in the search bar, qualifying assets are shown in a list as illustrated in Figure 53. Each asset in the list is shown with a different background color to indicate its type, also with some summary information (type, application domains, starting price, etc.), allowing users to easily find whether something in the list is relevant to their needs. By hovering the mouse over an asset in this list, a "*View*" button appears to allow access to all information available about the corresponding asset (see Section 2.4).

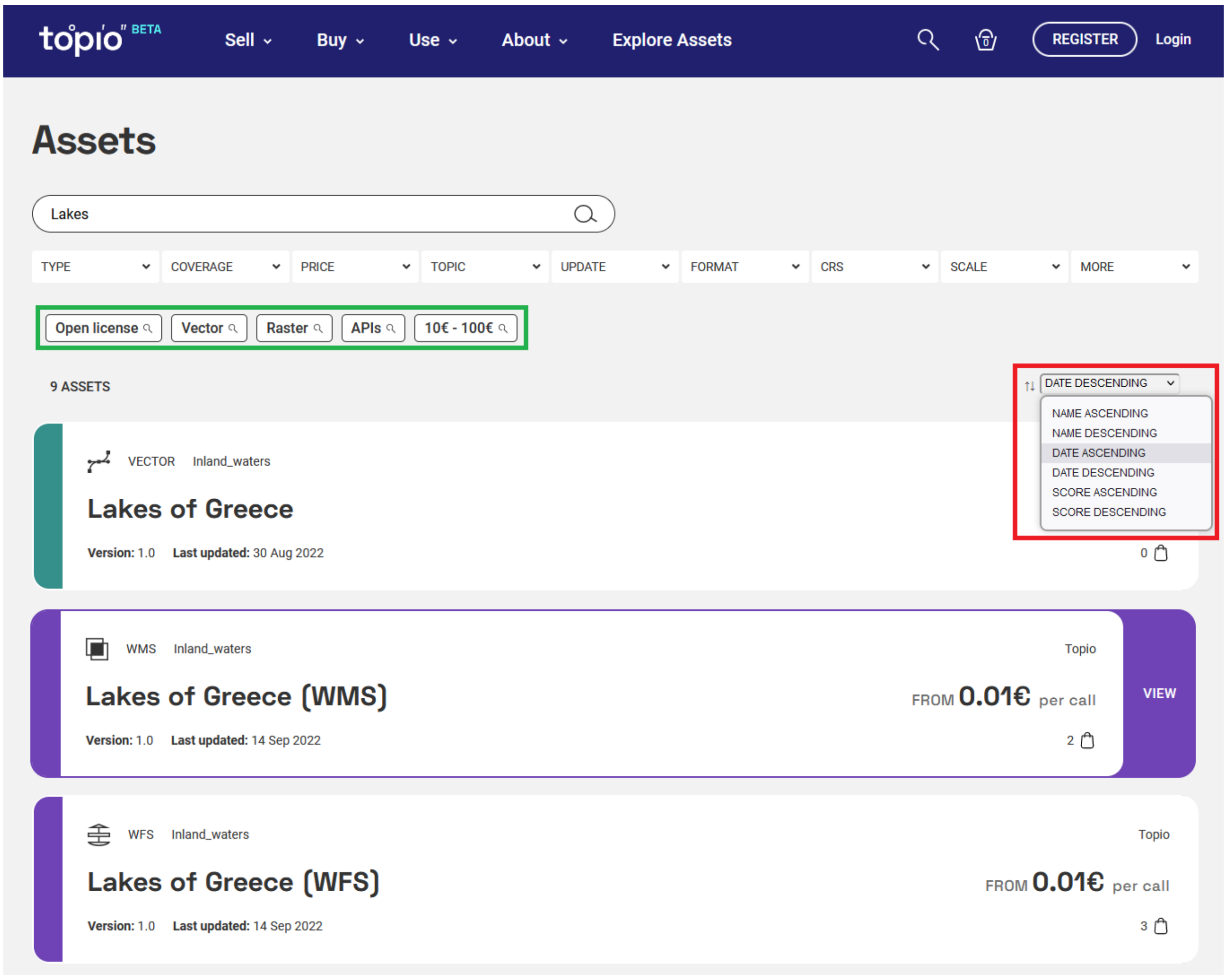


Figure 53: List of assets qualifying to search criteria

Optionally, this list of assets can be further *filtered* by several extra options (tabs concerning *asset type, coverage, price, topic, time of update, format, CRS, scale*, etc.), each of them with its own conditions. Some of the filtering conditions (Figure 54) may come from a *closed dictionary* (e.g., asset types, topics, file formats), others can be *user-specified* (e.g., price limit or range), enabling potential customers to narrow down their selection to assets that mostly match their preferences based on multiple filtering criteria. In addition, the spatial area of interest may be specified either by drawing a rectangle on map or by selecting the country from the dropdown list on the upper left corner of the map, so that only assets with *coverage* that overlaps the specified area are included in the results. Of course, users can always add extra filtering conditions, modify those filters already applied (e.g., increase the price range, or add extra file formats), or even *cancel* some or all of them (by clicking on the × symbol next to each filter below the search bar), thus triggering an update in the listed assets. In addition, several *frequently used filters* (*Open license, Vector, Raster, APIs, 10€ - 100€*) are available below the search bar (highlighted in a green box in Figure 53), exactly as in the landing page. These offer a handy shortcut to quickly specify such a filter without going through the aforementioned tabs that provide various options and more detailed conditions under each tab.

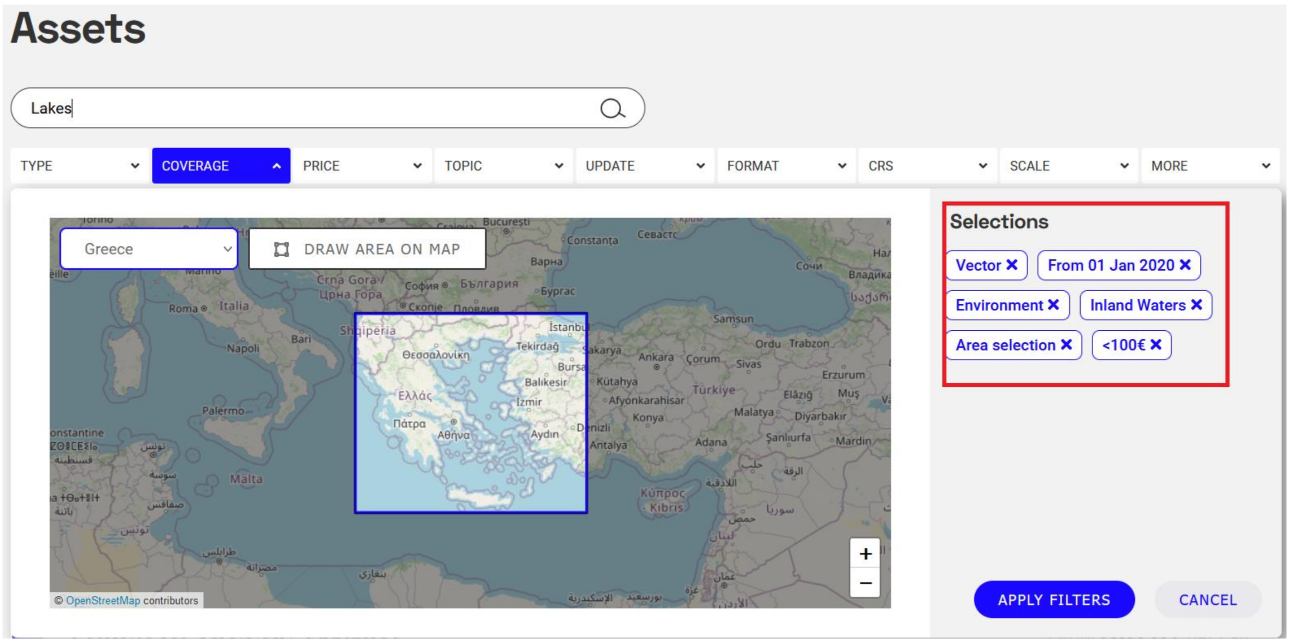
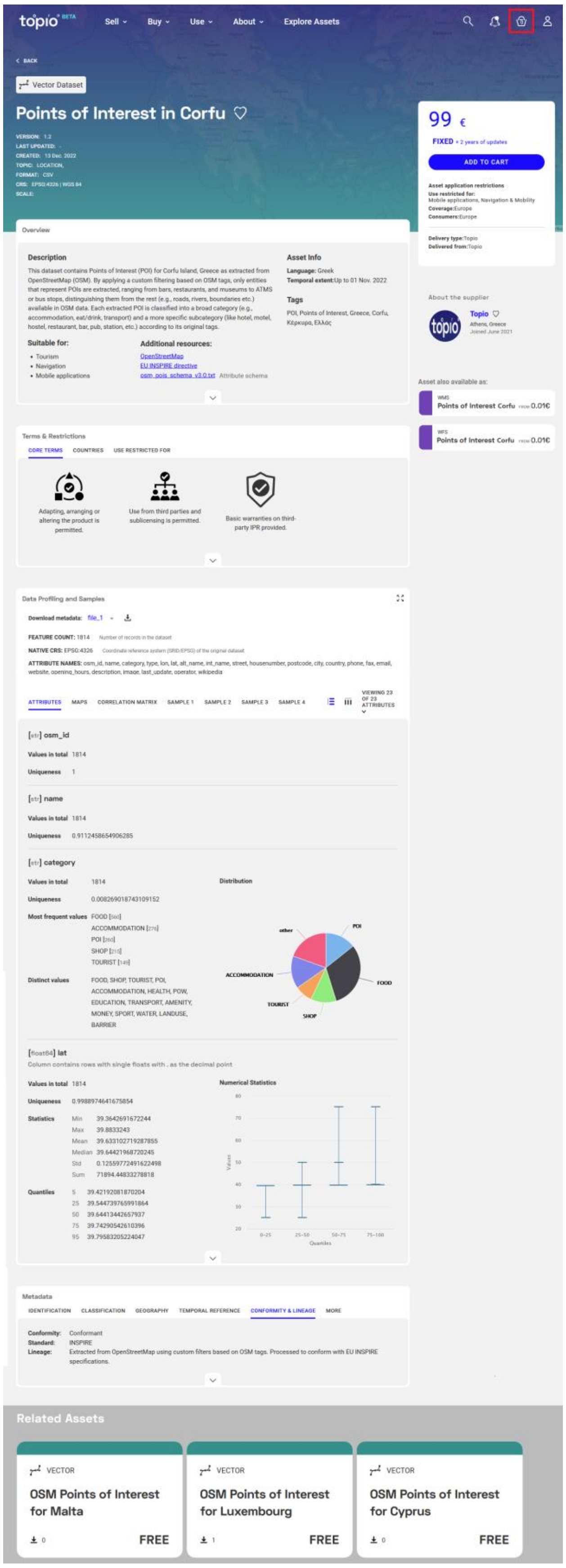


Figure 54: Several filtering criteria applied for refined search

2.4. Asset View

When a user chooses to view an asset, she is shown a web page like the one in Figure 55. This *Asset View* provides complete information about this particular asset, enabling a prospective customer to inspect its technical characteristics, examine the terms and conditions regarding its acquisition, assess its suitability for her needs through intuitive data profiling with visualizations and statistics, get samples and complete metadata for closer inspection, and of course purchase the asset. Note that depending on the asset type (data file, API, etc.), this View Asset page is adjusted accordingly, but maintains a common layout, as detailed next.

- ***Asset identity***. At the top, it shows the *identity* of the asset with concise information about its contents and purpose. This includes the *type* of the asset (vector, raster, service, etc.), the *version(s)* available for purchase, a *topic* suggested by its supplier about its broader application, as well as dates of *creation date* and *last update* (if available) of this asset. This summary also includes the original file *format*, coordinate reference system (*CRS*), original *scale*, etc. A registered user can also mark a data asset as *favorite* by clicking on the heart button ♥ next to this.
- ***Asset Overview***. Below this basic identification, the user can find an *overview* of this asset's contents as provided by the supplier. This includes a short *description* of the asset, the specific application domain(s) it is most *suitable for* use, asset information (language, temporal extent, etc.), as well as a collection of characteristic *tags* (i.e., keywords also useful during asset search). *Additional resources* about this asset (e.g., detailed documentation, description of the attributes, exemplary use cases) can also appear as links to external files or webpages (e.g., at the supplier's website).



Points of Interest in Corfu

VERSION: 1.2
LAST UPDATED: -
CREATED: 13 Dec. 2022
TOPIC: LOCATION,
FORMAT: CSV
CRS: EPSG:4326 | WGS 84
SCALE:

99 €

FIXED + 2 years of updates

Asset application restrictions
Use restricted for:
Mobile applications, Navigation & Mobility
Coverage:Europe
Consumers:Europe

Delivery type:Topio
Delivered from:Topio

Overview

Description

This dataset contains Points of Interest (POI) for Corfu Island, Greece as extracted from OpenStreetMap (OSM). By applying a custom filtering based on OSM tags, only entities that represent POIs are extracted, ranging from bars, restaurants, and museums to ATMS or bus stops, distinguishing them from the rest (e.g., roads, rivers, boundaries etc.) available in OSM data. Each extracted POI is classified into a broad category (e.g., accommodation, eat/drink, transport) and a more specific subcategory (like hotel, motel, hostel, restaurant, bar, pub, station, etc.) according to its original tags.

Suitable for:
- Tourism
- Navigation
- Mobile applications

Additional resources:
OpenStreetMap
EU INSPIRE directive
osm_pois_schema_v3.0.txt Attribute schema

Asset Info
Language: Greek
Temporal extent:Up to 01 Nov. 2022

Tags
POI, Points of Interest, Greece, Corfu, Κέρκυρα, Ελλάς

About the supplier
Topio
Athens, Greece
Joined June 2021

Asset also available as:
WMS Points of Interest Corfu FROM 0.01€
WFS Points of Interest Corfu FROM 0.01€

Terms & Restrictions

Adapting, arranging or altering the product is permitted.

Use from third parties and sublicensing is permitted.

Basic warranties on third-party IPR provided.

Data Profiling and Samples

Download metadata: file_1

FEATURE COUNT: 1814 Number of records in the dataset

NATIVE CRS: EPSG:4326 Coordinate reference system (SRID/EPSG) of the original dataset

ATTRIBUTE NAMES: osm_id, name, category, type, lon, lat, alt_name, int_name, street, housenumber, postcode, city, country, phone, fax, email, website, opening_hours, description, image, last_update, operator, wikipedia

[str] osm_id

Values in total 1814
Uniqueness 1

[str] name

Values in total 1814
Uniqueness 0.9112458654906285

[str] category

Values in total 1814
Uniqueness 0.008269018743109152
Most frequent values FOOD [560]
ACCOMMODATION [276]
POI [260]
SHOP [215]
TOURIST [149]
Distinct values FOOD, SHOP, TOURIST, POI, ACCOMMODATION, HEALTH, POW, EDUCATION, TRANSPORT, AMENITY, MONEY, SPORT, WATER, LANDUSE, BARRIER

Distribution

[float64] lat
Column contains rows with single floats with . as the decimal point

Values in total 1814
Uniqueness 0.9988974641675854

Statistics		
	Min	39.3642691672244
	Max	39.8833243
	Mean	39.633102719287855
	Median	39.64421968720245
	Std	0.12559772491622498
	Sum	71894.44833278818

Quantiles		
	5	39.42192081870204
	25	39.544739765991864
	50	39.64413442657937
	75	39.74290542610396
	95	39.79583205224047

Numerical Statistics

Metadata

Conformity: Conformant
Standard: INSPIRE
Lineage: Extracted from OpenStreetMap using custom filters based on OSM tags. Processed to conform with EU INSPIRE specifications.

Related Assets

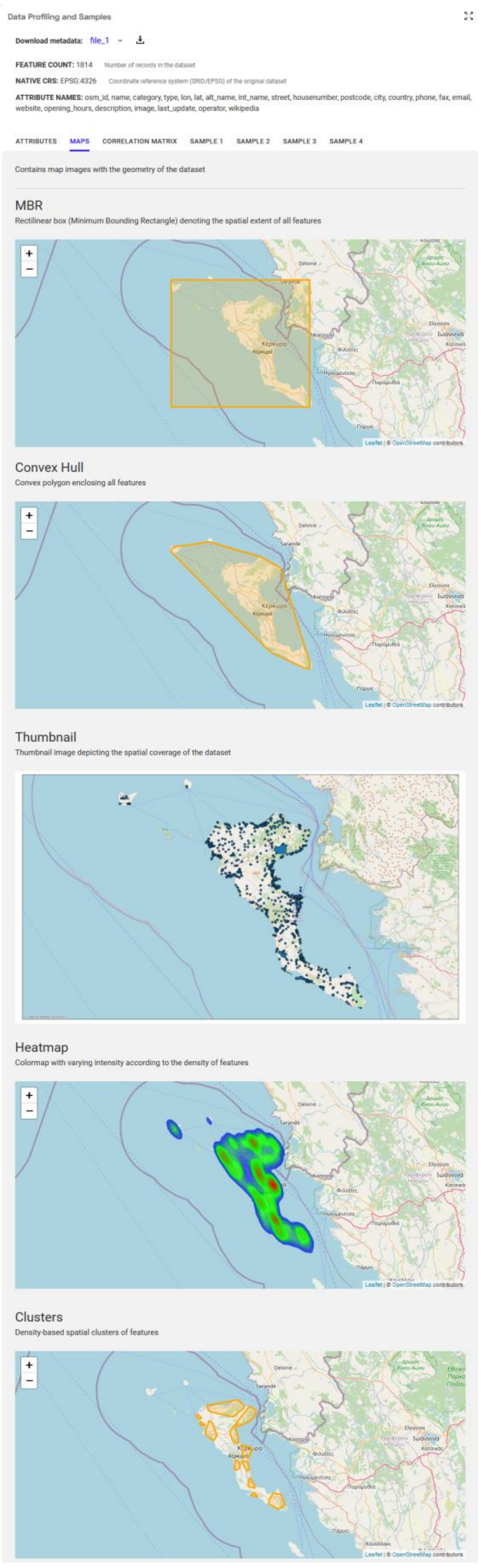
VECTOR
OSM Points of Interest for Malta
0 FREE

VECTOR
OSM Points of Interest for Luxembourg
1 FREE

VECTOR
OSM Points of Interest for Cyprus
0 FREE

Figure 55: Asset View

- ***Purchase options***. On the top right side of the asset webpage, a user can view the *purchase options* available for this asset. She can choose one of the *pricing models* designated by the supplier for this asset and check the respective price and any extra costs (e.g., for delivery with physical means). The name of the *supplier* who delivers the asset is shown with a URL, as well as information regarding the asset format (physical / digital). When the user presses the "*Add to cart*", this asset is added to her shopping cart provided that she is signed in the marketplace. By clicking on the cart icon (enclosed in a red box in Figure 55), the list of assets currently in the user's cart is displayed.
- ***Terms and Restrictions***. This section provides succinct information about terms and restrictions regarding the use of the asset. The type of the license, the countries where this asset may be utilized, the application domains where its use is allowed, etc., are described here as specified by the asset supplier in the template contract associated with this asset (see Section 2.8.5).
- ***Data profiling and samples***. To increase transparency and trust in the marketplace, this section is crucial for prospective customers as it offers comprehensive and intuitive information about each asset before purchase. Depending on the asset type, this section is adjusted accordingly. For *data file assets* (e.g., vector, raster), this profiling concerns *automated metadata*, such as coverage, spatial distribution, attribute statistics, quality indicators, etc., as well as indicative *samples* per data asset. All these are calculated by the platform before an asset becomes available for purchase in the marketplace; registered users can view this data profile and also download it in JSON format for closer inspection. Specifically for *vector data assets*, this automated metadata offers detailed information per thematic attribute, pie charts or histograms for its own distribution of values, lists of distinct and most frequent values, as well as various statistics (e.g., boxplots of quantiles for numerical values). Several types of *map-based plots* are also available (MBR, convex hull, heatmaps and clusters, etc., as shown in Figure 56). Such plots are prepared in advance and can be rendered as high-resolution images or superimposed in interactive maps. For numerical attributes, a *correlation matrix* indicates the degree of their pairwise correlation after analyzing their attribute values. In this matrix, cell values close to 1 indicate that the respective pair of attributes are highly correlated, whereas cell values close to -1 indicate anti-correlated attributes. Prospective customers are offered with one or more *data samples*, which may be provided by the supplier, but they can be also automatically extracted by the platform. Each available sample can be viewed directly through the platform, but it can be downloaded as well.



Data Profiling and Samples

Download metadata: file_1

FEATURE COUNT: 1814 Number of records in the dataset

NATIVE CRS: EPSG:4326 Coordinate reference system (SRID/EPSG) of the original dataset

ATTRIBUTE NAMES: osm_id, name, category, type, lon, lat, alt_name, int_name, street, housenumber, postcode, city, country, phone, fax, email, website, opening_hours, description, image, last_update, operator, wikipedia

Contains map images with the geometry of the dataset

MBR

Rectilinear box (Minimum Bounding Rectangle) denoting the spatial extent of all features

Convex Hull

Convex polygon enclosing all features

Thumbnail

Thumbnail image depicting the spatial coverage of the dataset

Heatmap

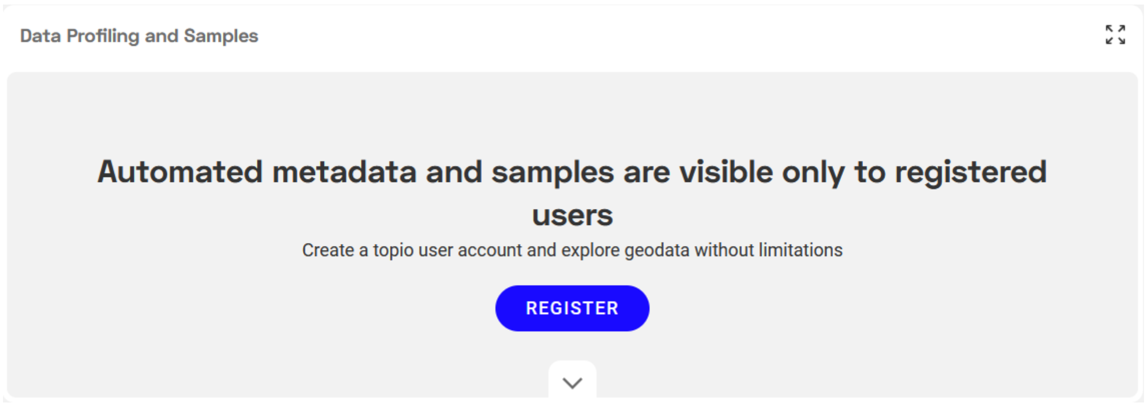
Colormap with varying intensity according to the density of features

Clusters

Density-based spatial clusters of features

Figure 56: Automated spatial metadata in interactive maps available to registered users

Of course, this detailed inspection of all aspects for an asset is available to registered users only. Visitors of the marketplace which have not been registered or not signed in as users cannot view any profiles, nor can they download any samples available per data asset. As illustrated in Figure 57, such visitors are shown a “greyed out” panel with no decipherable metadata information and they are urged to register or sign in the platform to explore the profile of each data asset in full.


Data Profiling and Samples

Automated metadata and samples are visible only to registered users

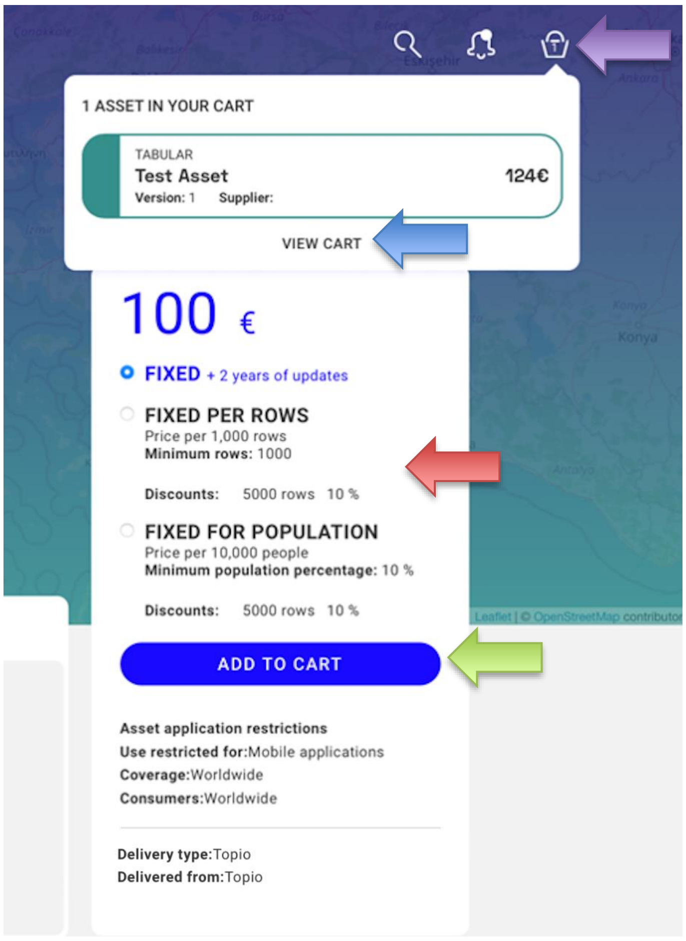
Create a topio user account and explore geodata without limitations

Figure 57: Automated metadata not visible by unregistered users

- ***Standard Metadata***. This section at the bottom of the Asset View page displays metadata elements as provided by the supplier regarding this asset and are organized is several categories. Clicking on each tab, the user can get detailed information about the identification of this asset, its classification, its geographical properties (e.g., MBR, scale), temporal reference, conformity and lineage, and more. Users can download this metadata for closer inspection.

2.5. Purchase Assets

To make a purchase, a registered user (*i.e., asset consumer*) simply adds an *asset* to the shopping cart. In case the asset is a data file, the user must first choose one of the available pricing options from the corresponding list (red arrow in Figure 58) at the asset’s view page (selection is enabled only if more than one pricing options are available, otherwise the pricing is fixed, e.g., Figure 59) and then press the button “Add to cart” (green arrow in both Figure 58 and Figure 59).


1 ASSET IN YOUR CART

TABULAR
Test Asset 124€
Version: 1 Supplier:

VIEW CART

100 €

FIXED + 2 years of updates

FIXED PER ROWS
Price per 1,000 rows
Minimum rows: 1000

Discounts: 5000 rows 10 %

FIXED FOR POPULATION
Price per 10,000 people
Minimum population percentage: 10 %

Discounts: 5000 rows 10 %

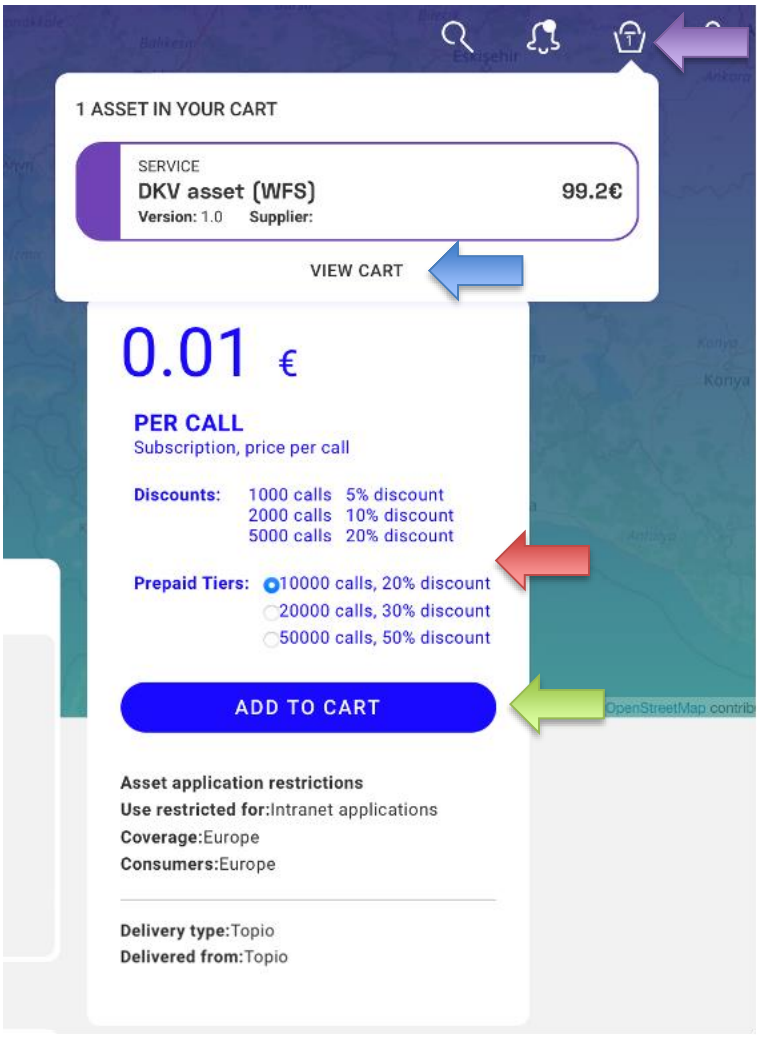
ADD TO CART

Asset application restrictions
Use restricted for: Mobile applications
Coverage: Worldwide
Consumers: Worldwide

Delivery type: Topio
Delivered from: Topio

Figure 58: Add a data file asset to the shopping cart

Upon adding an asset to her cart, the user is presented with a notification confirming the action, which contains the asset and its final price (including VAT) and a button ("View Cart") that redirects to the shopping cart (blue arrow in both Figure 58 and Figure 59). To finalize the purchase, the user must then simply click on the "View Cart" button, or *manually* navigate to her cart by clicking on the cart icon (purple arrow in both Figure 58 and Figure 59).



1 ASSET IN YOUR CART

SERVICE
DKV asset (WFS) 99.2€
Version: 1.0 Supplier:

VIEW CART

0.01 €

PER CALL
Subscription, price per call

Discounts:	1000 calls	5% discount
	2000 calls	10% discount
	5000 calls	20% discount

Prepaid Tiers: (● 10000 calls, 20% discount) 20000 calls, 30% discount; 50000 calls, 50% discount

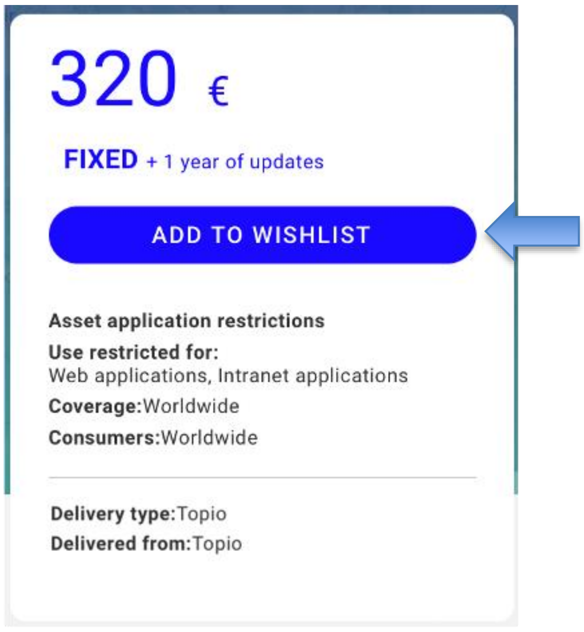
ADD TO CART

Asset application restrictions
Use restricted for:Intranet applications
Coverage:Europe
Consumers:Europe

Delivery type:Topio
Delivered from:Topio

Figure 59: Add an API asset to the shopping cart

In case the supplier hasn't completed her KYC/KYB validation yet (see Section 2.8.11), a prospective consumer cannot purchase the asset yet. Instead, she can add it to her "Favourites" list (see Section 2.7.4) by clicking on the "Add to Wishlist" button (blue arrow in Figure 60), which replaces the "Add to Cart" button described above.



320 €

FIXED + 1 year of updates

Asset application restrictions

Use restricted for:
Web applications, Intranet applications

Coverage:Worldwide

Consumers:Worldwide

Delivery type:Topio

Delivered from:Topio

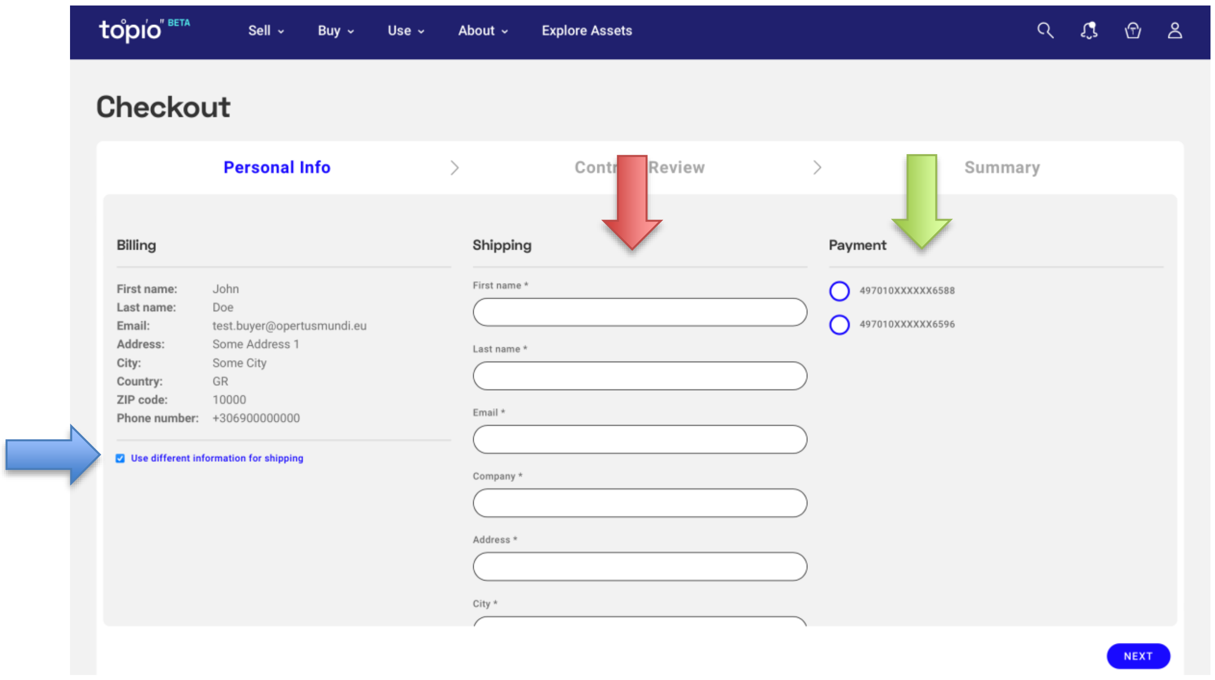
Figure 60: Add an asset to the wishlist

2.5.1. Checkout Wizard

After clicking on the “Proceed to Checkout” button, the user is redirected to the *checkout wizard*, where she must finalize her purchase. The steps of the wizard are similar both for data file, and API assets and are described next.

2.5.1.1. Personal Info

In the first page of the checkout wizard (Figure 61), the user is presented with her previously provided billing information. By default, the billing address is the same as the shipping address (the latter is only relevant if the selected assets will be delivered via physical means). If she desires so, she can use a different shipping address by checking the corresponding box (blue arrow). Doing so enables all the fields in the shipping info column (red arrow in the figure), where she is requested to provide the necessary details. Finally, on the rightmost column (green arrow in the figure), the user can select her payment method of preference for this purchase. In case she is registered as a simple user, she is requested at this point to also register as a consumer and provide any missing information. Finally, she can proceed to the next checkout wizard page using the “Next” button at the bottom.



Checkout

Billing

First name:	John
Last name:	Doe
Email:	test.buyer@opertusmundi.eu
Address:	Some Address 1
City:	Some City
Country:	GR
ZIP code:	10000
Phone number:	+306900000000

☑ Use different information for shipping

Shipping

First name *

Last name *

Email *

Company *

Address *

City *

Payment

○ 497010XXXXXX6588

○ 497010XXXXXX6596

Figure 61: Checkout: Personal information.

2.5.1.2. Contract Review

At the "Contract Review" page of the checkout wizard (Figure 62) the user is presented with the contracts associated with the geospatial data assets to be purchased. The page is divided into two separate columns. On the left column (green arrow in the figure), the contracts associated with the selected assets are listed with all the necessary information (asset title, provider, date of purchase and price). The current contract under review is annotated with blue font. If the user has agreed with one of the contracts, a check sign appears at the bottom right of the corresponding item in the list (purple arrow in the figure). On the right column of the page (orange arrow in the figure), the user can download the current contract in PDF format on her computer to review it by clicking on the "Download Contract" button (red arrow in the figure).

The user must either accept or decline a contract via the corresponding buttons (blue arrow in the figure). In case the user declines a contract, a pop-up window appears (Figure 63), asking the user to confirm the action. Upon clicking on the "Submit" button (blue arrow in Figure 63), the corresponding asset is removed from the contract review page, as indicated with a blue arrow in Figure 64.

After going through all the contracts, the user can either proceed to the "Summary" page ("Next" button, bottom right), or go to the previous "Personal Info" page ("Previous" button, bottom left). As in all wizards in the platform, the user can also navigate to various pages by just clicking on the corresponding tab at the top.

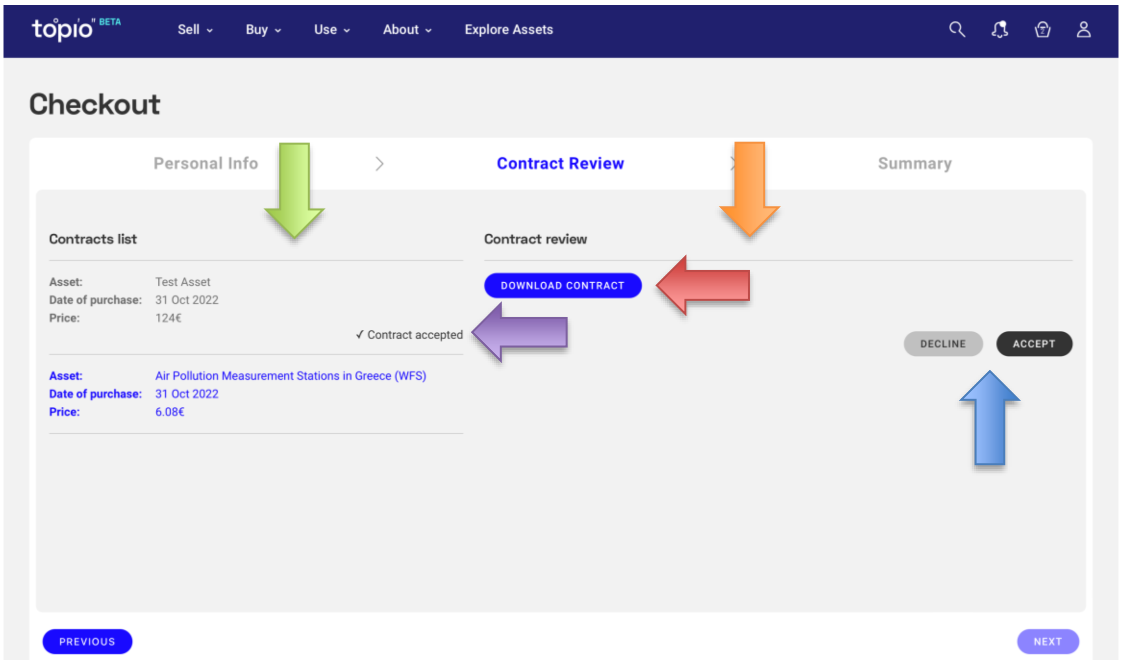


Figure 62: Checkout: Contract review

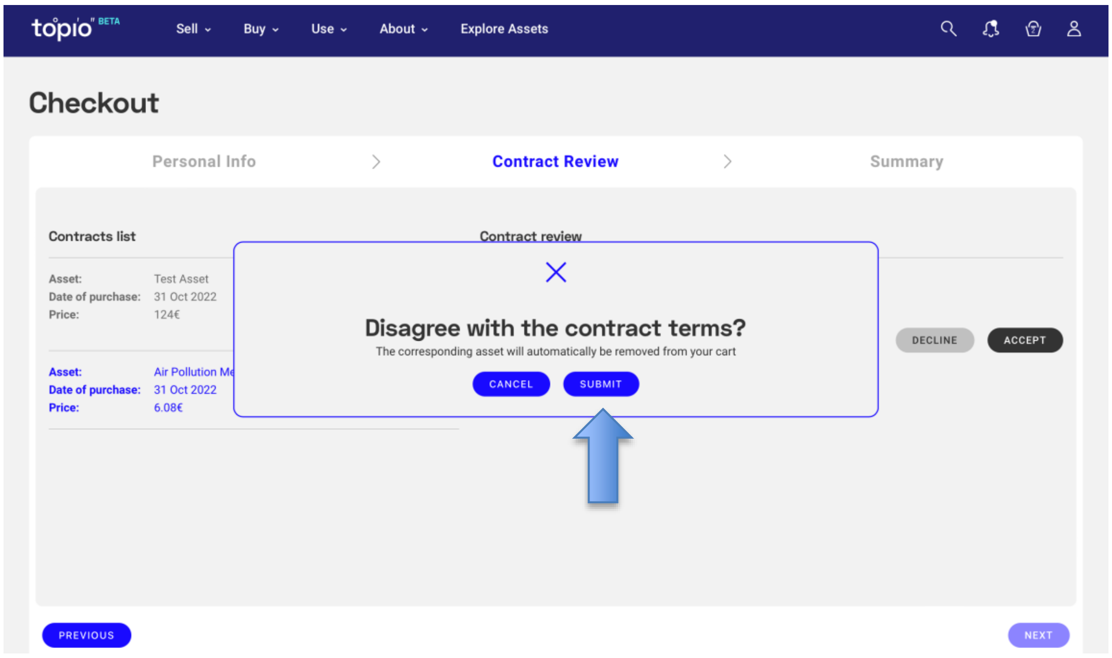
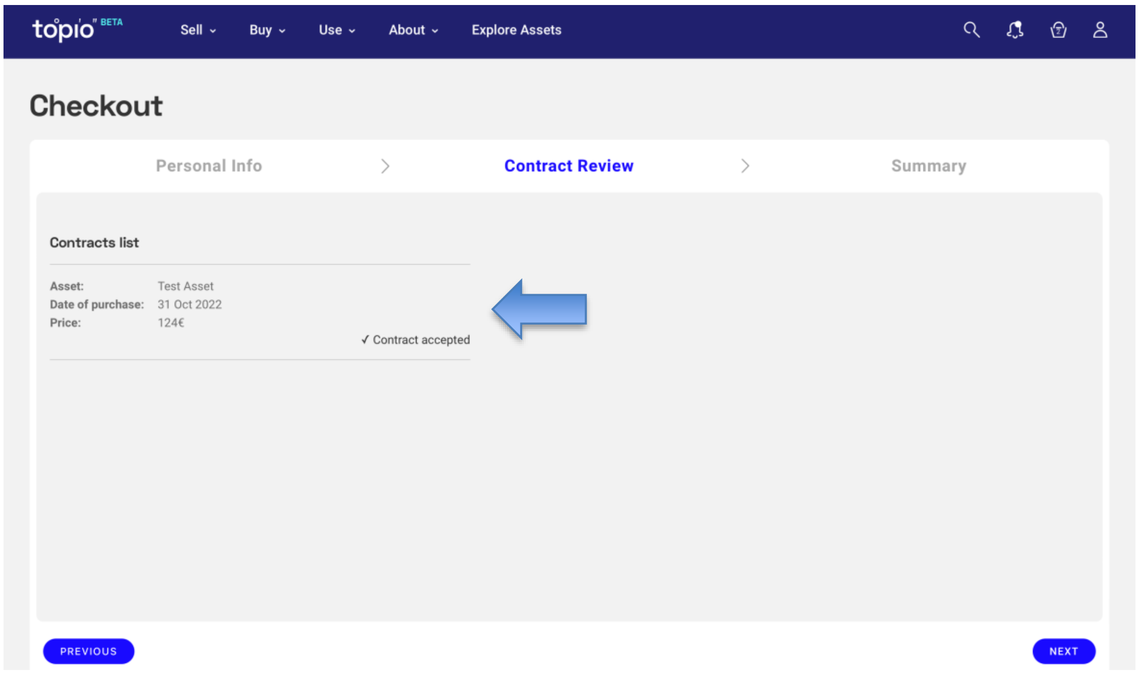


Figure 63: Checkout: Decline contract



Contracts list

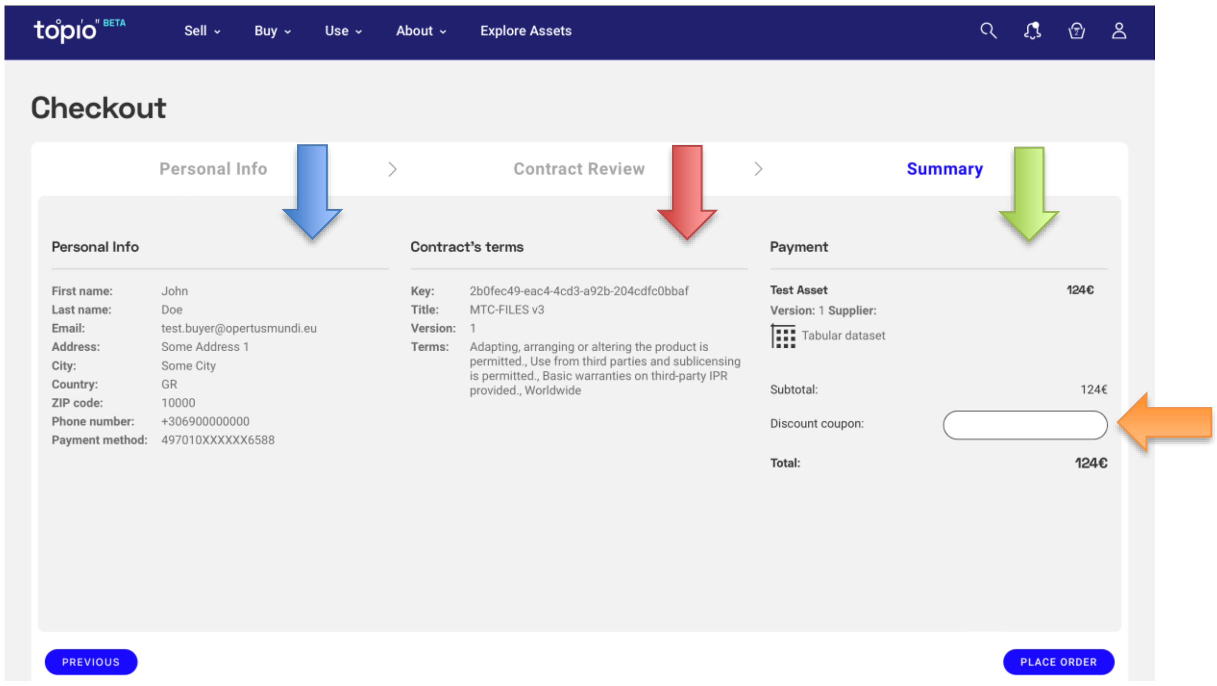
Asset:	Test Asset
Date of purchase:	31 Oct 2022
Price:	124€

✓ Contract accepted

Figure 64: Checkout: Only accepted contracts remain in the list

2.5.1.3. Summary

At the "Summary" page (Figure 65) of the wizard, the user is presented with all the information concerning her purchase, including her personal information ("Personal Info" column, blue arrow in the figure) and details regarding the contracts she has agreed with ("Contract Terms" column, red arrow in the figure). On the rightmost column of this page ("Payment" column, green arrow in the figure), the user is presented with a list of all the assets to be purchased, along with the final price. If she has a discount coupon (i.e., provided to her by a certain seller or the platform), she can enter the code in the corresponding box (orange arrow in the figure) and the discount is applied (the total amount of the purchase is reduced accordingly). Finally, the user can finalize the order via the "Place Order" button at the bottom.



Personal Info

First name:	John
Last name:	Doe
Email:	test.buyer@opertusmundi.eu
Address:	Some Address 1
City:	Some City
Country:	GR
ZIP code:	10000
Phone number:	+306900000000
Payment method:	497010XXXXXX6588

Contract's terms

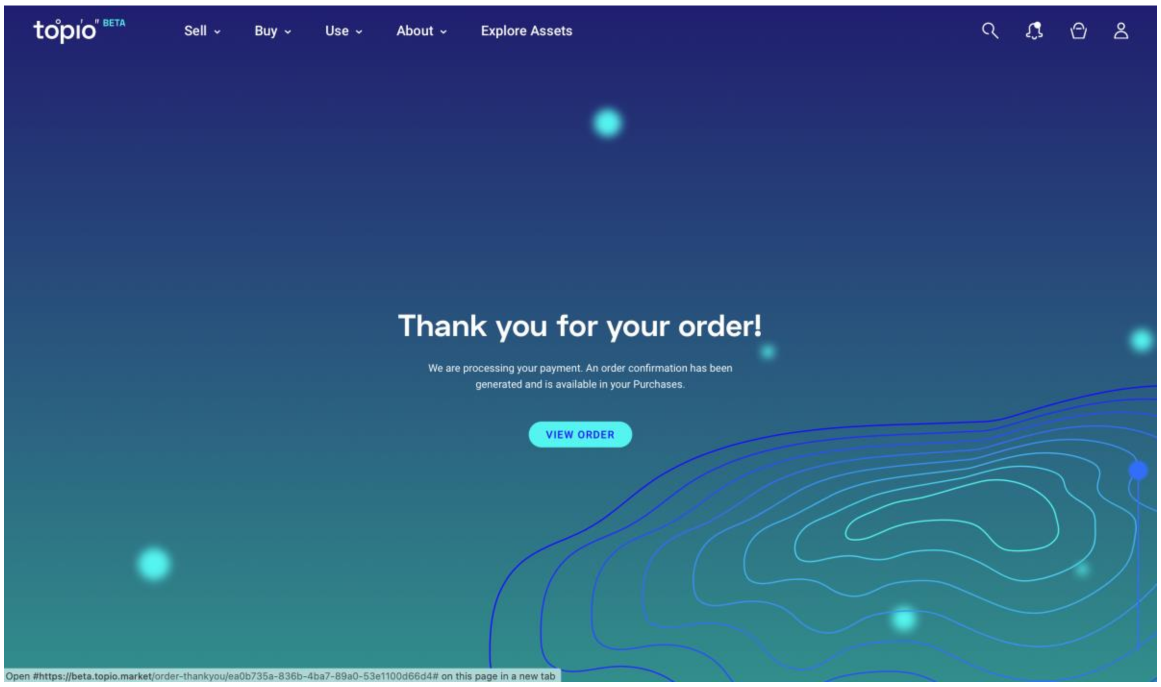
Key:	2b0fec49-eac4-4cd3-a92b-204cdfc0bbaf
Title:	MTC-FILES v3
Version:	1
Terms:	Adapting, arranging or altering the product is permitted., Use from third parties and sublicensing is permitted., Basic warranties on third-party IPR provided., Worldwide

Payment

Test Asset 124€
Version: 1 Supplier:
Tabular dataset

Subtotal: 124€
Discount coupon:
Total: 124€

Figure 65: Checkout: Summary



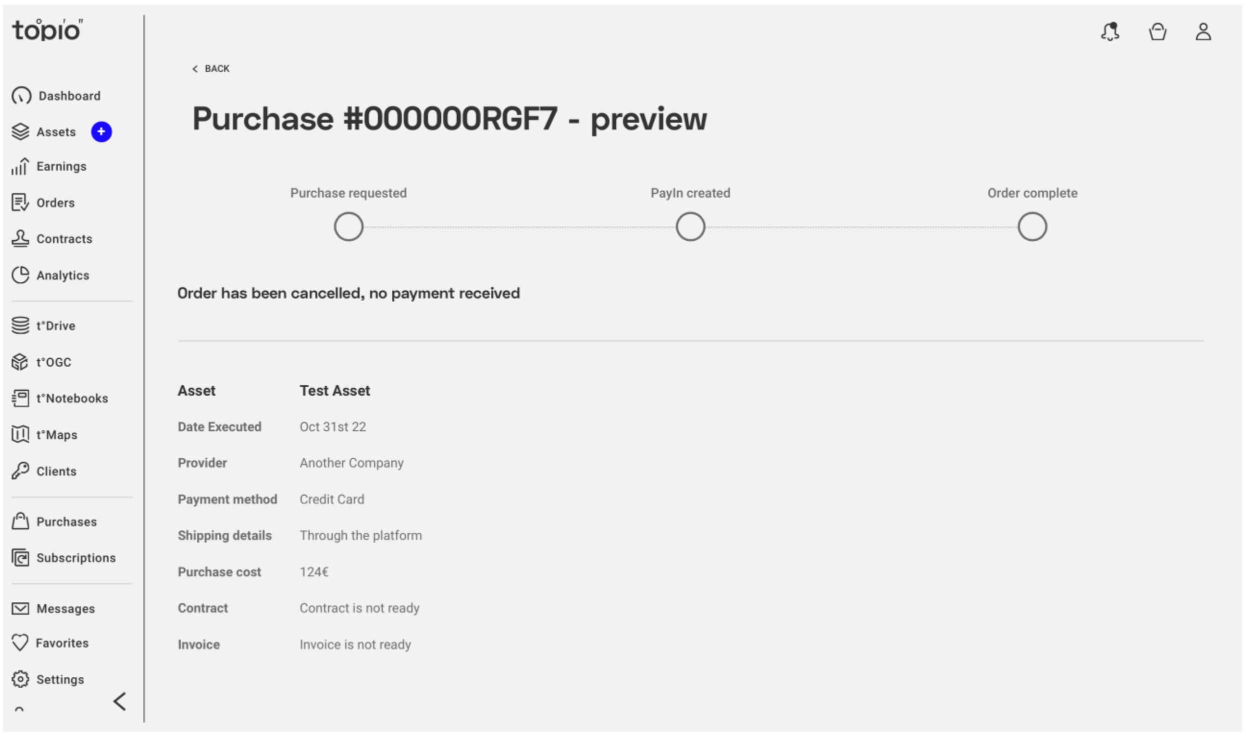
Thank you for your order!

We are processing your payment. An order confirmation has been generated and is available in your Purchases.

Open #https://beta.topio.market/order-thankyou/ea0b735a-836b-4ba7-89a0-53e1100d66d4# on this page in a new tab

Figure 66: Thank you message

The user is presented with a thank you message in a separate page (Figure 66). She can optionally click on the "View Order" button in this page, to view details. Doing so redirects her to the corresponding order page in her Dashboard (Figure 67), where she can view details regarding the payment, along with other useful information.



Purchase #000000RGF7 - preview

Purchase requested — PayIn created — Order complete

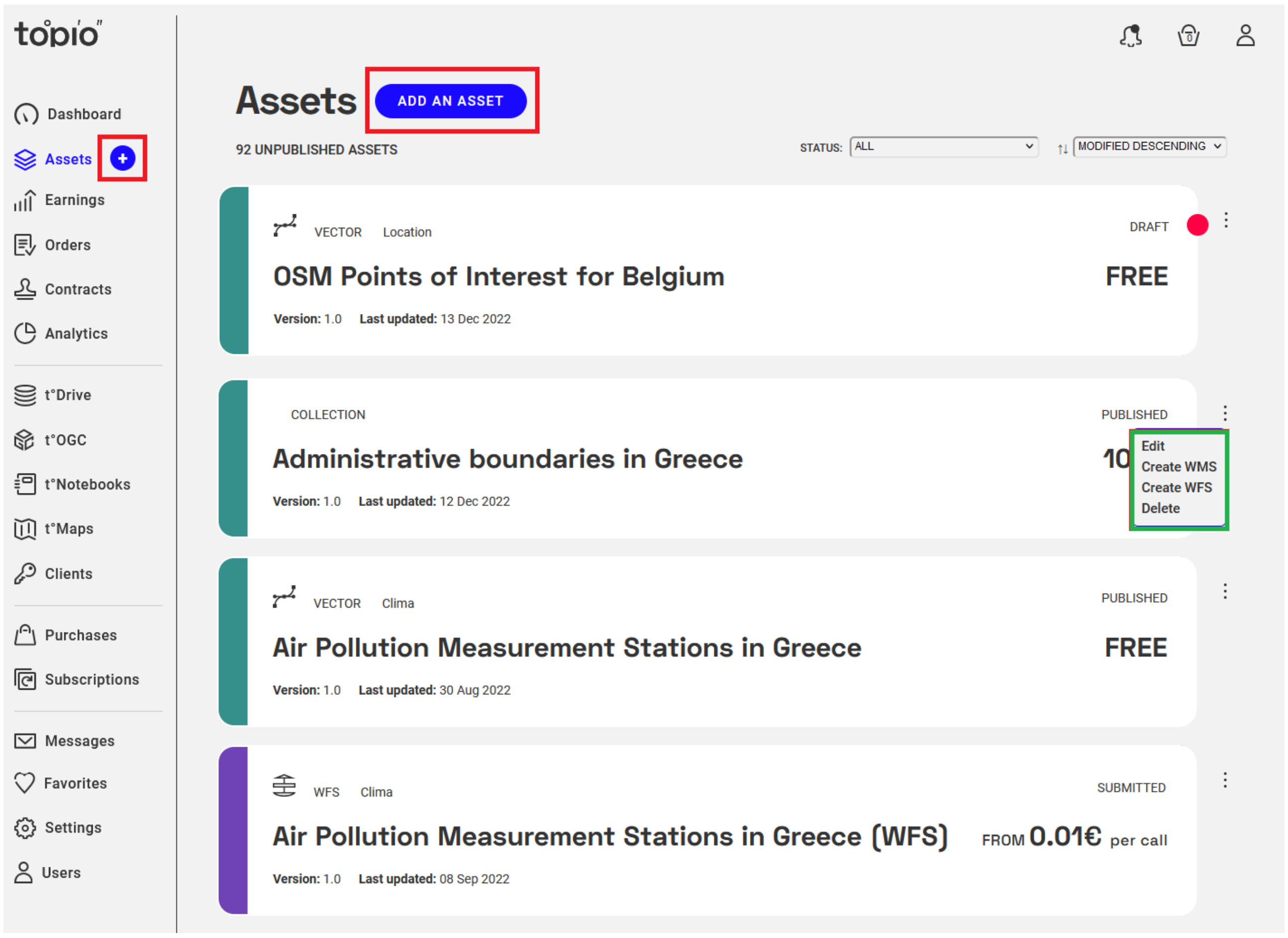
Order has been cancelled, no payment received

Asset	Test Asset
Date Executed	Oct 31st 22
Provider	Another Company
Payment method	Credit Card
Shipping details	Through the platform
Purchase cost	124€
Contract	Contract is not ready
Invoice	Invoice is not ready

Figure 67: View order

2.6. Publish Assets

A supplier can publish an asset directly from her *Dashboard*, which offers full control over all its activity in the marketplace as discussed in detail in Section 0. As shown in Figure 68, by either clicking on the + button in the sidebar (highlighted in a red box) or the button "*Add an Asset*", the supplier can publish from scratch a new asset in the marketplace easily, thanks to a step-by-step wizard. If publication of at an asset is not yet complete, this is displayed with a notification in red next to their state in the list. Of course, suppliers retain full control of their own assets and can apply certain actions on each of them (*edit, delete, create WMS, create WFS*), as highlighted in the green box in Figure 68 and will be discussed in Section 2.6.2.



Assets

92 UNPUBLISHED ASSETS

STATUS: ALL ↑↓ MODIFIED DESCENDING

VECTOR Location DRAFT
OSM Points of Interest for Belgium FREE
Version: 1.0 **Last updated:** 13 Dec 2022

COLLECTION PUBLISHED
Administrative boundaries in Greece 10
Version: 1.0 **Last updated:** 12 Dec 2022

VECTOR Clima PUBLISHED
Air Pollution Measurement Stations in Greece FREE
Version: 1.0 **Last updated:** 30 Aug 2022

WFS Clima SUBMITTED
Air Pollution Measurement Stations in Greece (WFS) FROM 0.01€ per call
Version: 1.0 **Last updated:** 08 Sep 2022

Figure 68: Adding an asset from the supplier's dashboard

2.6.1. Adding an asset

Once the supplier chooses to add an asset, a step-by-step *publishing wizard* is launched to assist her. Adding a data file asset to the marketplace has some differences from publishing other types of assets (e.g., API, EO), but generally the wizard involves similar steps with appropriate options at each step:

Asset type. The first step involves specification of the asset type (Figure 69). Currently, this can be one of the following:

- *Data file* (vector, tabular, raster, NetCDF). The supplier will be asked to upload a file asset or a collection of file assets that will be marketed through the platform. All types of geospatial data and formats (vector, raster), as well as *non-spatial* data (e.g., statistics) are allowed.
- *Open asset*. This option also concerns data files (of various types and formats as for data files) but offered with an open license for free.
- *API*. A supplier can create a new API (WMS, WFS) based on a file asset already traded through the marketplace or uploaded by the supplier in her own storage space in the platform.

- *Collection*. This concerns a collection of already published assets that can be purchased together.
- *Sentinel Hub*. This specifically concerns Earth Observation (EO) assets from Sentinel Hub concerning satellite imagery products from various open and proprietary EO collections.

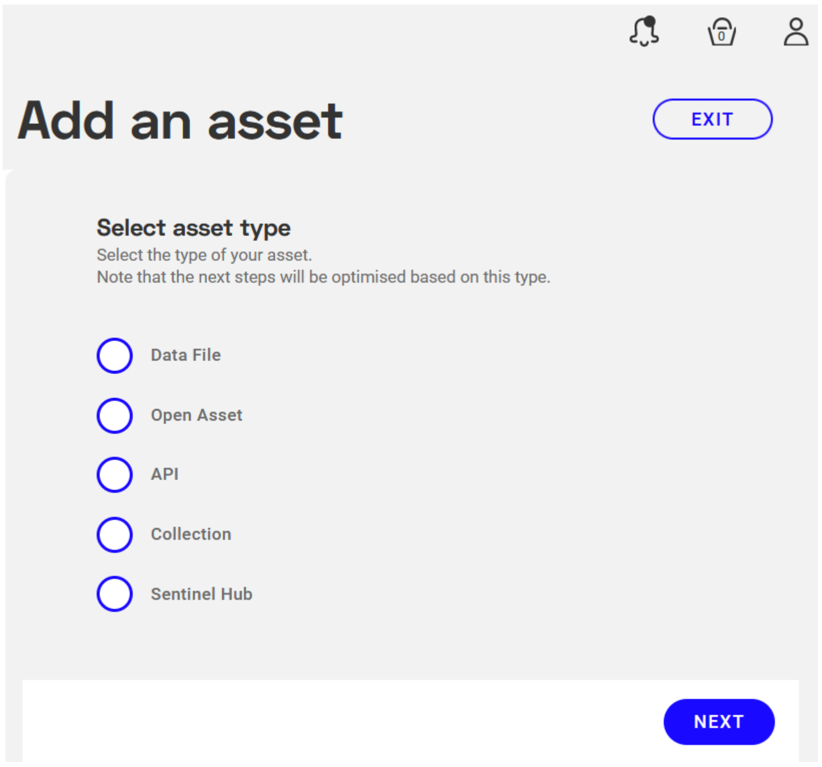

Add an asset
EXIT
Select asset type
Select the type of your asset.
Note that the next steps will be optimised based on this type.
Data File
Open Asset
API
Collection
Sentinel Hub
NEXT

Figure 69: Specifying the type of a new asset

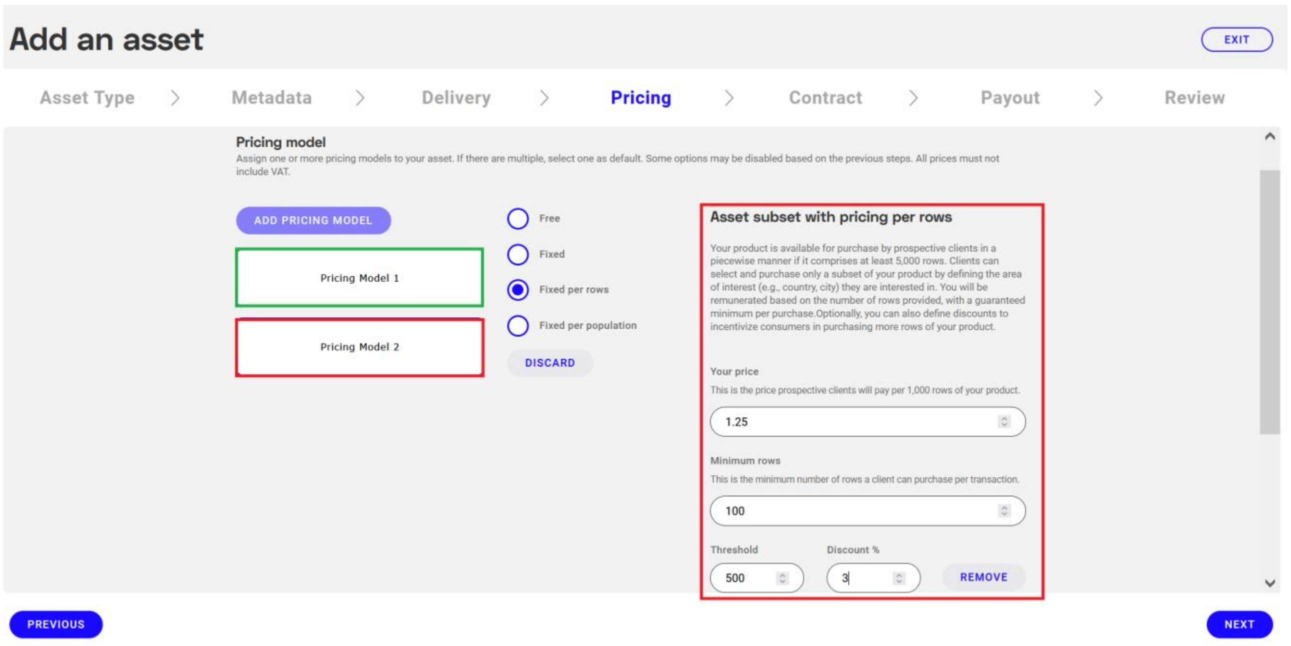
Metadata. The supplier must specify metadata information (through the UI or upload a file) for this new asset, such as title, description, format, keywords, coordinate reference system, etc. that will be validated by the Helpdesk and will be visible to prospective customers when the asset gets published. For asset APIs based on published assets, this step is disabled, since the platform already holds all the required metadata.

Delivery. Next, the supplier must specify how the asset will be delivered to customers after purchase. For data assets, two options are available:

(a) *By the platform*: If delivery will be effectuated *by the platform*, the supplier must specify how the platform will get access to this asset and store it internally for delivery. Suppliers can directly upload files to the repository, provide a valid web link (e.g., an ftp site, Dropbox) where the asset is available, or a path to their own storage space (Topio Drive). Note that this is the default option for APIs based on assets or data files in the platform.

(b) *By the supplier's own means*. Suppliers may prefer their own channel (e.g., physical media like CD or DVD, or digitally via an ftp server) for delivery if the data is too large or is updated frequently (e.g., daily) or they do not wish to upload it at their Topio Drive storage. For the particular case of external APIs published through the platform, these will be delivered by external means as specified by their supplier.

Pricing. The supplier must select at least one *pricing model* from those available for this type of asset, such as free of charge, one-time purchase with no updates, asset subset with pricing per number of rows or per population of the corresponding geographical area, etc. Multiple pricing models may be chosen as shown in Figure 70, e.g., a vector data asset may be marketed both as a one-time purchase (Pricing model 1), but portions of it may be available and charged according to the number of extracted rows (0-1000, 100-2000, etc.)


Add an asset

EXIT

Asset Type > Metadata > Delivery > **Pricing** > Contract > Payout > Review

Pricing model

Assign one or more pricing models to your asset. If there are multiple, select one as default. Some options may be disabled based on the previous steps. All prices must not include VAT.

ADD PRICING MODEL

Pricing Model 1

Pricing Model 2

Free [✓ Fixed per rows] Fixed Fixed per population

DISCARD

Asset subset with pricing per rows

Your product is available for purchase by prospective clients in a piecewise manner if it comprises at least 5,000 rows. Clients can select and purchase only a subset of your product by defining the area of interest (e.g., country, city) they are interested in. You will be remunerated based on the number of rows provided, with a guaranteed minimum per purchase.Optionally, you can also define discounts to incentivize consumers in purchasing more rows of your product.

Your price

This is the price prospective clients will pay per 1,000 rows of your product.

1.25

Minimum rows

This is the minimum number of rows a client can purchase per transaction.

100

Threshold	Discount %
500	3

REMOVE

PREVIOUS

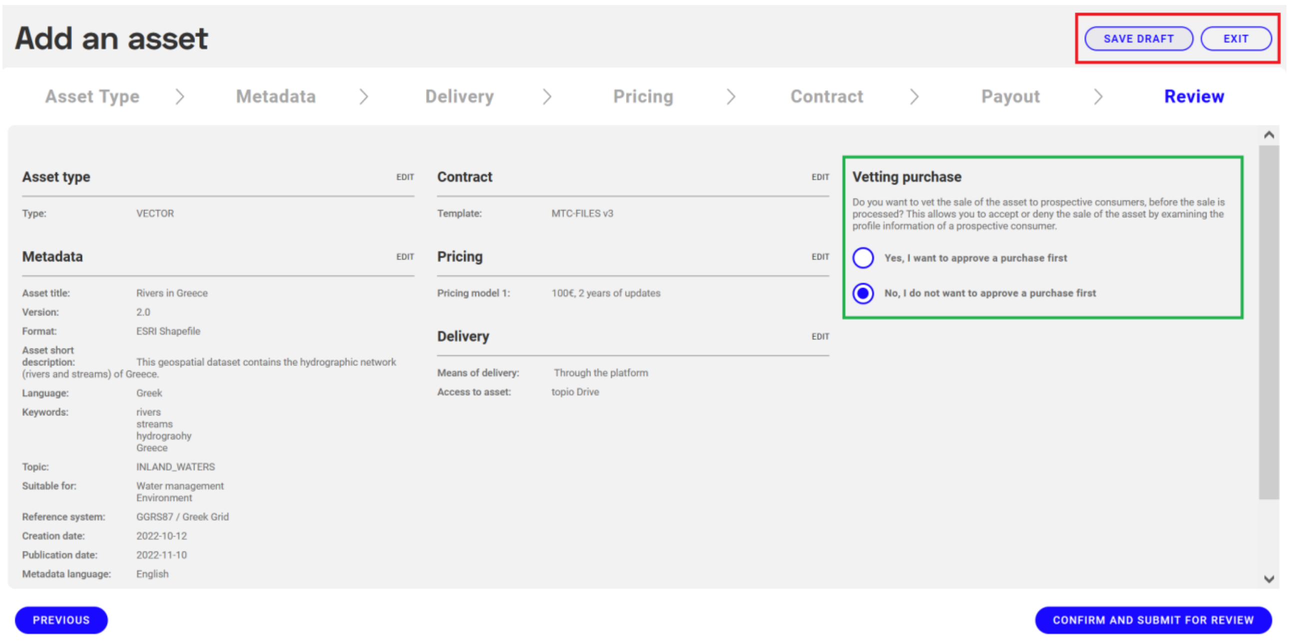
NEXT

Figure 70: Specifying the pricing model for a new data asset

Contracts. The supplier must then choose a *contract template*, which specifies the terms and conditions for selling this new data asset. Such contract templates cover the various categories of her own assets marketed through the platform (vector file assets, API assets, etc.), each with possibly different conditions (Section 2.8.5). In addition, suppliers can *create a new contract template* for this asset (the flow is discussed in Section 2.8.5), or even *upload their own custom contract*, essentially bypassing the standard publishing flow. Specifically for API assets, a predefined contract with all terms and conditions regarding access to this API can be previewed; the supplier can inspect the full text, download it as a pdf, but cannot edit any of the terms. This step in the wizard does not apply to *open assets*, but it is replaced with the typical *License* options for open data (CC-BY, CC-BY-SA, CC-BY-ND, ODbL, etc.).

Payout. Through the wizard, the supplier selects how she will receive all payments from future customers of her new asset: either reimbursed *through the platform* (a IBAN bank account must have been specified in her platform settings) or *by external means* (e.g., a direct money transfer to her bank account, to her PayPal account, or to her credit/debit card).

Review. At the final step, the supplier can check all her choices in the previous steps regarding this new asset (Figure 71). Any choices at a particular step can be modified by clicking on the respective tab (e.g., pricing). Most importantly, the supplier has also the option to *vet purchases* for data file assets, as highlighted in the green box. If this vetting option is activated, the supplier has to approve or deny each order by examining the profile information of a prospective customer *prior to every sale*. After examining all her choices and verifying they are in accordance with her preferences, the supplier can press the *confirmation* button (at the bottom right corner in Figure 71) and submit this asset for *review* by the Helpdesk of the platform.



Add an asset
SAVE DRAFT
EXIT
Asset Type > Metadata > Delivery > Pricing > Contract > Payout > Review
EDIT
PREVIOUS
CONFIRM AND SUBMIT FOR REVIEW

Asset type

Type:	VECTOR

Metadata

Asset title:	Rivers in Greece
Version:	2.0
Format:	ESRI Shapefile
Asset short description:	This geospatial dataset contains the hydrographic network (rivers and streams) of Greece.
Language:	Greek
Keywords:	rivers streams hydrograohy Greece
Topic:	INLAND_WATERS
Suitable for:	Water management Environment
Reference system:	GGRS87 / Greek Grid
Creation date:	2022-10-12
Publication date:	2022-11-10
Metadata language:	English

Contract

Template:	MTC-FILES v3

Pricing

Pricing model 1:	100€, 2 years of updates

Delivery

Means of delivery:	Through the platform
Access to asset:	topio Drive

Vetting purchase

Do you want to vet the sale of the asset to prospective consumers, before the sale is processed? This allows you to accept or deny the sale of the asset by examining the profile information of a prospective consumer.

Yes, I want to approve a purchase first [✓ No, I do not want to approve a purchase first]

Figure 71: Summary of asset information before submission to the platform

Each submitted asset automatically gets a unique persistent identifier (*PID*) for reference. In case of a new data file asset, the supplier is prompted to upload the file; no data uploading is necessary for API assets. While the asset is still under review by the Helpdesk or remains a draft with some steps still incomplete, it is shown with a red notification at the top of the list of assets in the dashboard. The new asset does not get immediately published, but only if it has been successfully approved by the platform's Helpdesk once all necessary details regarding the asset are considered valid. In case of issues regarding a submitted asset, the supplier is notified by the platform's Helpdesk with a message, so that she can make the necessary changes in some stage(s) of the publication process and resubmit the asset. Note that at any stage of the wizard, the supplier can *Save* her current choices as an *asset draft* and complete its publication later or *Exit* the wizard discarding this draft completely (using the buttons in the top right corner, highlighted in the red box in Figure 71).

2.6.2. Actions regarding an asset

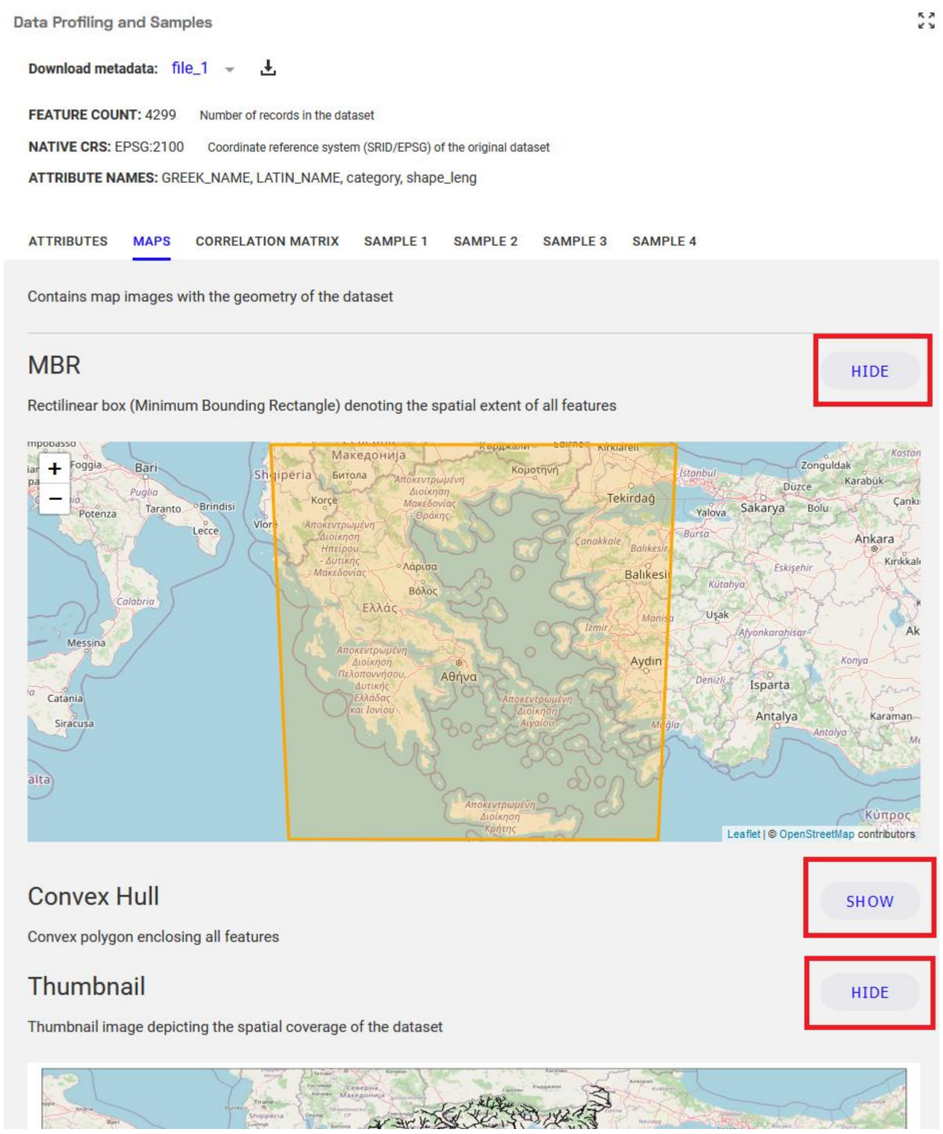
If a red notification is shown next to the Assets option in the dashboard, it indicates that there are assets unpublished or under review, which require some action from the supplier (e.g., some steps in the publishing wizard have not been performed yet). Such assets are also shown on the top of the assets list with a notification in red. In this case, the supplier may carry out several actions (enclosed in a red box in Figure 68):

Edit. A wizard is launched, which allows the supplier to edit any information previously specified regarding the asset (metadata, contract, pricing, delivery, payout). This follows most of steps presented when adding an asset from scratch; however, at each step the supplier can now examine her previously defined settings and modify them. At the last step, the supplier is prompted to *review* all specifications and confirm them before submitting the asset to the Helpdesk for review.

Create API. These options are available for published assets only and allow the supplier to respectively create a *Web Map Service* (WMS), or a *Web Feature Service* (WFS). The publication flow is similar to the one used for adding a new API asset, but actually skips the first two steps (*Asset type, API details*) in the wizard and starts directly from the metadata specification. The next steps in the wizard enable the supplier to choose pricing model(s), contract, payout method(s) and finally review all settings before submitting the new API asset for review by the Helpdesk.

Delete. This option allows the supplier to remove this asset from the marketplace, thus it will be no longer traded through the platform and will not be available to future customers. However, it will be retained in the platform (if it is a data file uploaded by the supplier) for reference, as it may have been already sold to customers.

Approve data profiles. Soon after a data asset is submitted to the Helpdesk, the platform automatically invokes a profiling service that computes automated metadata and samples from this dataset; suppliers cannot interfere in this automatic process running in the backend. Once this process terminates, the supplier receives a notification to view its calculated profile (i.e., automated metadata and samples). For each item in this draft profile, the supplier can toggle its visibility (*Show/Hide*), not only per thematic attribute (e.g., statistics and charts as shown in Figure 72), but also every map displaying spatial features and aggregates (e.g., clusters, heatmaps), as well as all automatically created samples. Especially for *samples*, the platform offers suppliers the capability to *replace* any of those automatically computed by their own samples.


Data Profiling and Samples

Download metadata: file_1

FEATURE COUNT: 4299 Number of records in the dataset

NATIVE CRS: EPSG:2100 Coordinate reference system (SRID/EPSG) of the original dataset

ATTRIBUTE NAMES: GREEK_NAME, LATIN_NAME, category, shape_leng

Contains map images with the geometry of the dataset

MBR

Rectilinear box (Minimum Bounding Rectangle) denoting the spatial extent of all features

Convex Hull

Convex polygon enclosing all features

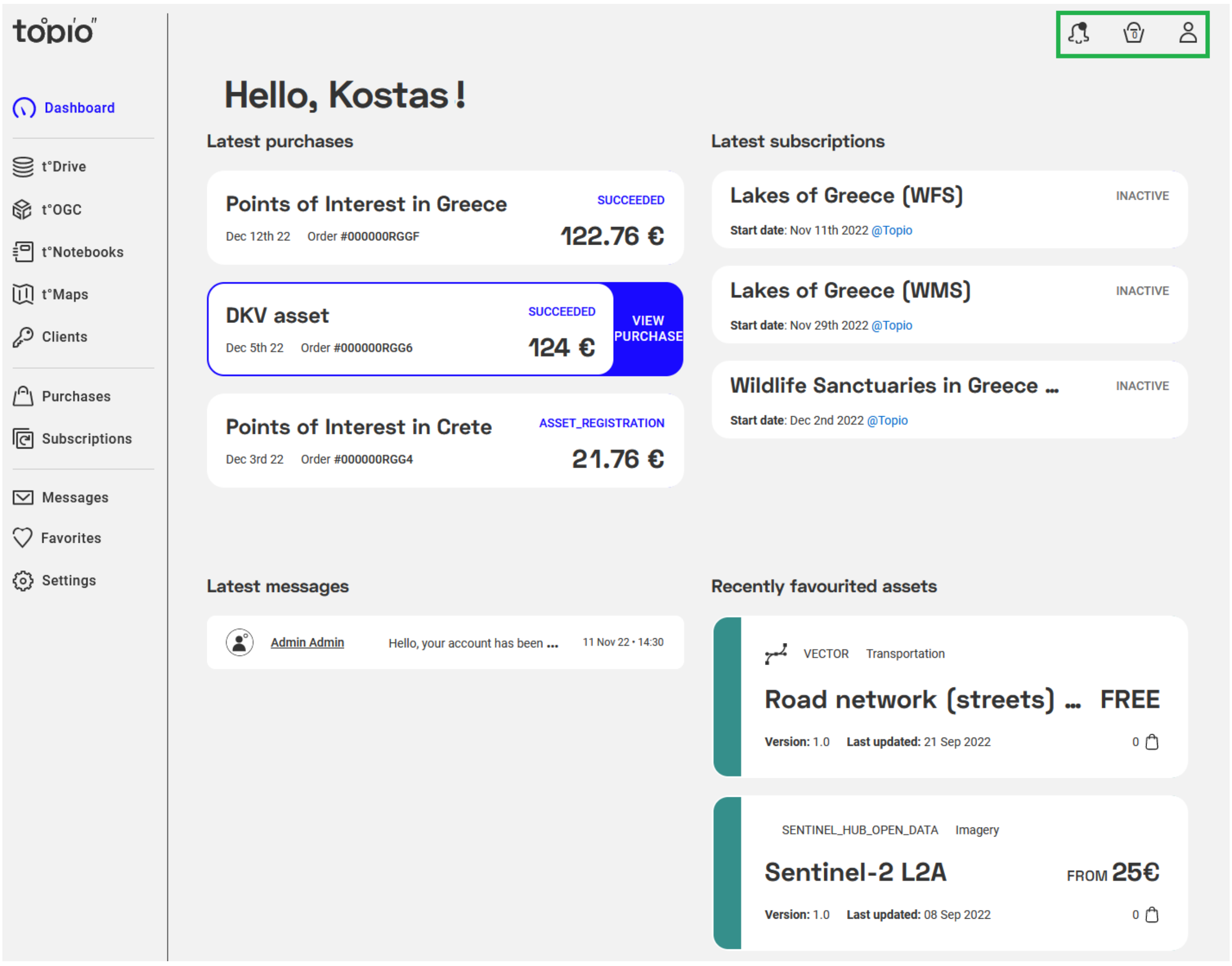
Thumbnail

Thumbnail image depicting the spatial coverage of the dataset

Figure 72: Choosing profile elements before publication of an asset

2.7. User Dashboard

Once a registered user signs in, her *Dashboard* page is shown with an overview of her recent activity, as illustrated in Figure 73. This includes a list of her most recently *purchased assets*, a list of latest *active subscriptions* to services offered through the platform, as well as a list of assets that have been recently added to her *favorites*. It also includes another panel listing *messages* from the marketplace, with a red notification indicating the number of unread messages in each thread. For a user that is not yet registered as a consumer, no purchases are displayed in the list and a button "*Become a consumer*" enables her to provide the necessary details (like nationality, full address, country of residence) for registered consumers so that she can purchase assets. If the user is already a consumer, then a button "*Become a supplier*" is shown, which prompts her to provide the necessary details (company address, VAT, Legal representative, Bank account) in order to be able to trade assets in the marketplace. Thanks to the icons shown on the top right (enclosed in a green box), a registered user has immediate access to *notifications* sent by the platform, her *cart* of assets under purchase, as well as her account details (*settings*, *profile*). On the *sidebar menu*, there are several available options that offer more detailed information to the user about her activity in the marketplace, as detailed next. Note that unregistered visitors of the marketplace have no access to the Dashboard and all this information.



Hello, Kostas !

Latest purchases

Points of Interest in Greece — SUCCEEDED — Dec 12th 22 Order #000000RGGF — 122.76 €

DKV asset — SUCCEEDED — Dec 5th 22 Order #000000RGG6 — 124 € — VIEW PURCHASE

Points of Interest in Crete — ASSET_REGISTRATION — Dec 3rd 22 Order #000000RGG4 — 21.76 €

Latest subscriptions

Lakes of Greece (WFS) — INACTIVE — Start date: Nov 11th 2022 @Topio

Lakes of Greece (WMS) — INACTIVE — Start date: Nov 29th 2022 @Topio

Wildlife Sanctuaries in Greece ... — INACTIVE — Start date: Dec 2nd 2022 @Topio

Latest messages

Admin Admin — Hello, your account has been ... — 11 Nov 22 • 14:30

Recently favourited assets

VECTOR Transportation — Road network (streets) ... — FREE — Version: 1.0 Last updated: 21 Sep 2022 — 0

SENTINEL_HUB_OPEN_DATA Imagery — Sentinel-2 L2A — FROM 25€ — Version: 1.0 Last updated: 08 Sep 2022 — 0

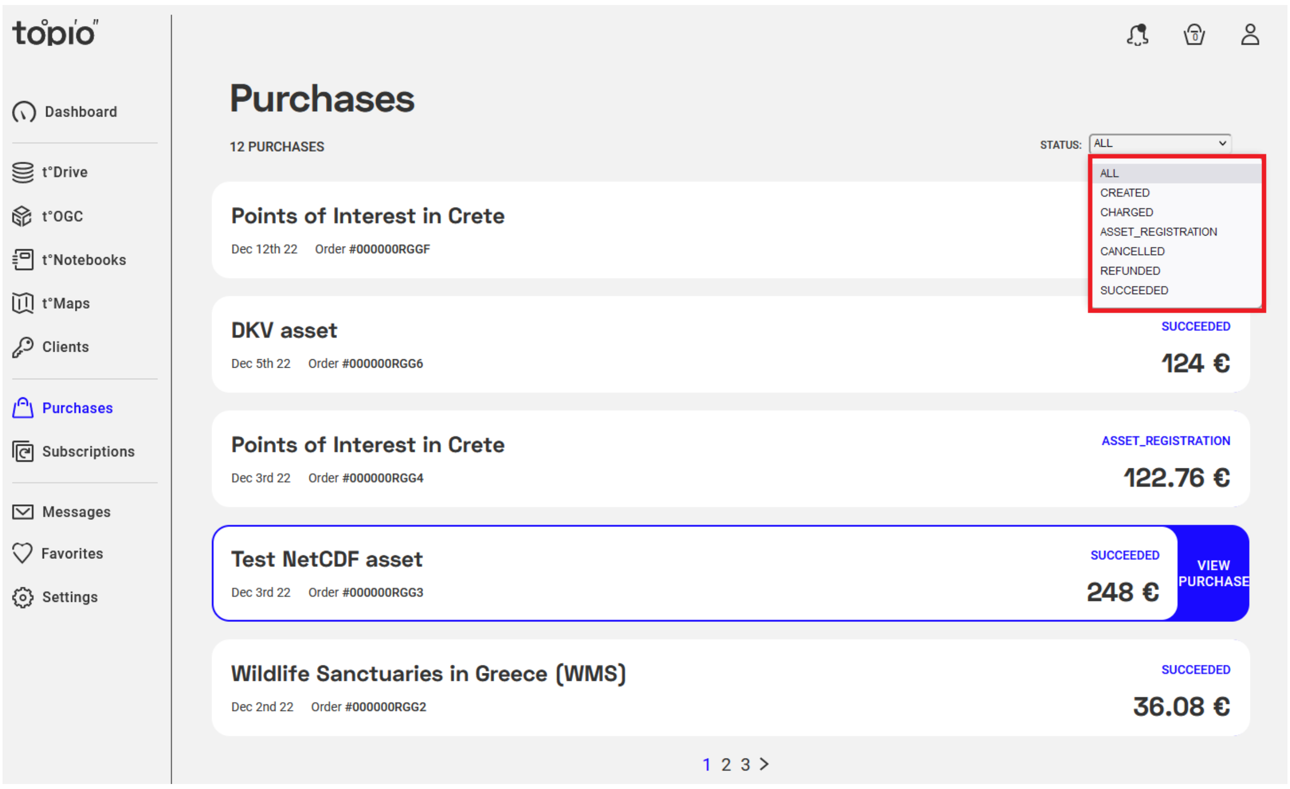
Figure 73: User's dashboard

2.7.1. Purchases

Once a user clicks on option "*Purchases*" from the sidebar (Figure 73), the full list of her purchased assets appears in reverse chronological order (most recent are shown first) as shown in Figure 74. The total number of purchases is visible on the top of the list. Pagination is also available, so that the user can navigate over the complete history of its purchased assets. As indicated by the red box, *filtering* can be applied on the contents of this list, so that the user can inspect purchased assets by their *status*: those that have been created, charged (payment under way with card), asset registration in the transaction history, cancelled, refunded, as well as completed (successful) purchases. Once the user hovers the mouse over a purchased item, it appears with a blue background and the user may click on it to view the complete purchase details.

On the right of each purchased item, a *More* button (shown with the three vertical dots) provides direct control over the purchase status (marked within a green box in Figure 74), so that the user can:

- *Acknowledge delivery* of the corresponding asset(s) to the platform. Once the user acknowledges that its purchased asset(s) have been delivered, this option is deactivated.
- *Cancel* the purchase if it is not yet complete; otherwise, this option is deactivated.
- *Delete* a completed purchase from her historical records is also possible; henceforth, this purchase will not be visible in this list.



Purchases

12 PURCHASES

STATUS: ALL

ALL
CREATED
CHARGED
ASSET_REGISTRATION
CANCELLED
REFUNDED
SUCCEEDED

Points of Interest in Crete
Dec 12th 22 Order #000000RGGF

DKV asset
Dec 5th 22 Order #000000RGG6
SUCCEEDED
124 €

Points of Interest in Crete
Dec 3rd 22 Order #000000RGG4
ASSET_REGISTRATION
122.76 €

Test NetCDF asset
Dec 3rd 22 Order #000000RGG3
SUCCEEDED
248 €
VIEW PURCHASE

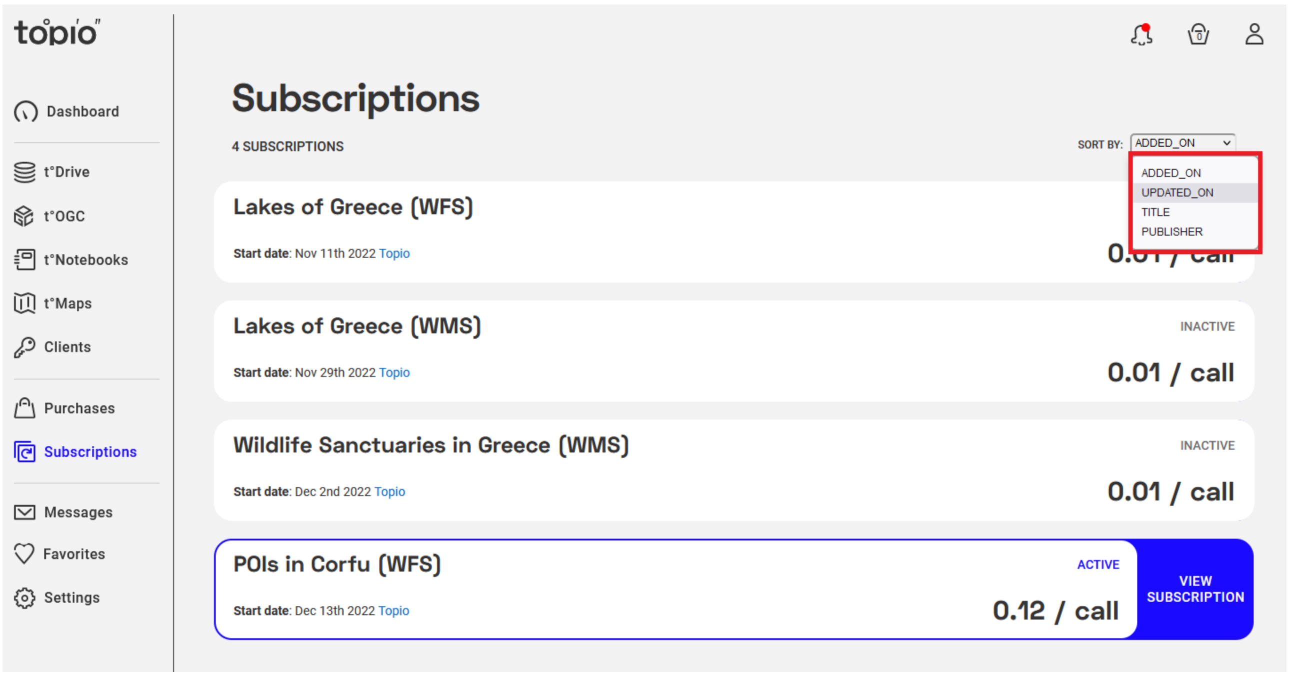
Wildlife Sanctuaries in Greece (WMS)
Dec 2nd 22 Order #000000RGG2
SUCCEEDED
36.08 €

1 2 3 >

Figure 74: User's list of purchased assets

2.7.2. Subscriptions

Clicking on option "*Subscriptions*" from the sidebar in her dashboard (Figure 73), a user can also view the list of all assets she is currently subscribed to. The total number of subscriptions is shown on the top of the list, which is navigable through pagination of its items. Typically, the user can sort all her subscriptions by several options (date *added, date updated, title, publisher*) enclosed in the red box. Once the user hovers the mouse over a particular subscription, this one appears with a blue background and the user may click on it to view its complete details (Figure 76). This includes the subscription plan, the supplier, start and expiration dates, we well as the assets associated and offered with this subscription.


topio

Dashboard
t°Drive
t°OGC
t°Notebooks
t°Maps
Clients
Purchases
Subscriptions
Messages
Favorites
Settings

Subscriptions

4 SUBSCRIPTIONS

SORT BY: ADDED_ON

ADDED_ON
UPDATED_ON
TITLE
PUBLISHER

Lakes of Greece (WFS)
Start date: Nov 11th 2022 Topio
0.01 / call

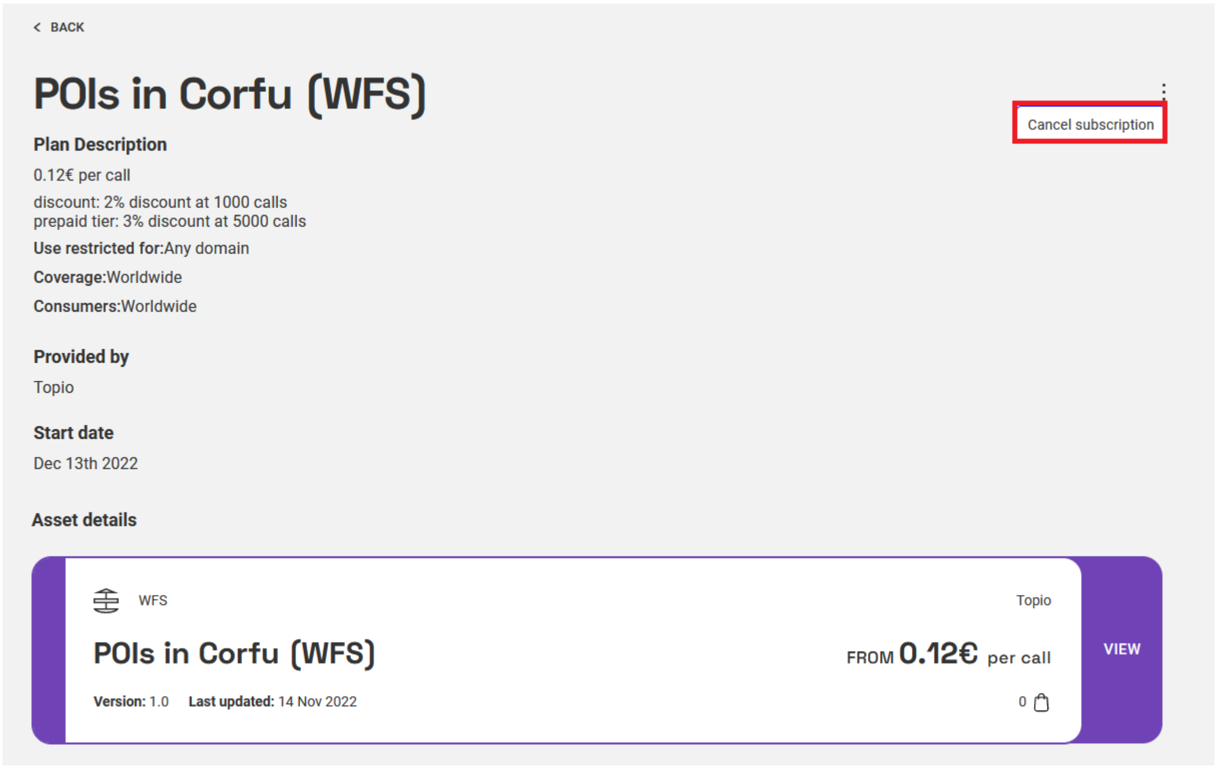
Lakes of Greece (WMS)
INACTIVE
Start date: Nov 29th 2022 Topio
0.01 / call

Wildlife Sanctuaries in Greece (WMS)
INACTIVE
Start date: Dec 2nd 2022 Topio
0.01 / call

POIs in Corfu (WFS)
ACTIVE
Start date: Dec 13th 2022 Topio
0.12 / call
VIEW SUBSCRIPTION

Figure 75: User's list of subscriptions

By clicking on a subscription in this list, its full details are shown as in Figure 76: the subscription plan with its cost and restrictions, the supplier, the starting date, as well as the particular asset details. Once again, the user can *Cancel* her subscription (option highlighted in the red box).


< BACK

POIs in Corfu (WFS)

Cancel subscription

Plan Description
0.12€ per call
discount: 2% discount at 1000 calls
prepaid tier: 3% discount at 5000 calls
Use restricted for: Any domain
Coverage: Worldwide
Consumers: Worldwide

Provided by
Topio

Start date
Dec 13th 2022

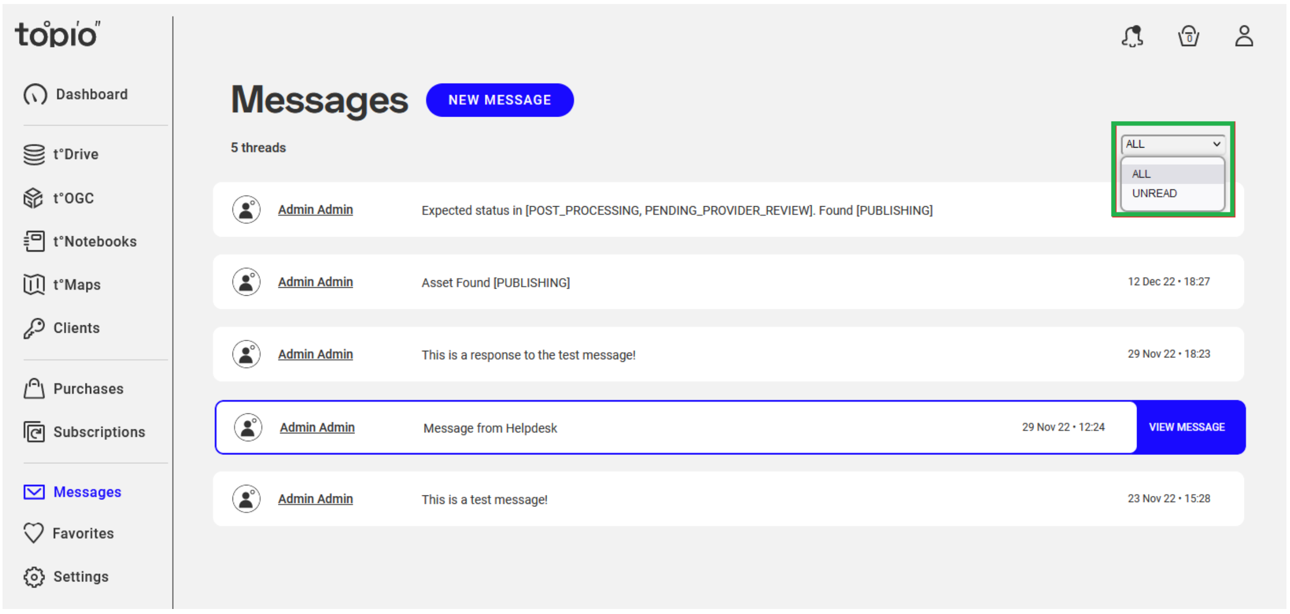
Asset details

WFS
Topio
POIs in Corfu (WFS)
FROM 0.12€ per call
Version: 1.0 **Last updated:** 14 Nov 2022
0
VIEW

Figure 76: Details of a user's subscription

2.7.3. Messages

The option "*Messages*" in the sidebar of the dashboard allows a user to view all her messages exchanged through the platform (Figure 77). As indicated the options in the green box, these messages can be filtered from those received in her *inbox*. By choosing *all* or *unread*, the respective messages in the list are filtered accordingly. Note that the sender, the subject, the beginning of the body text, as well as the timestamp are shown per message, so that the user immediately gets an overview of each one. Unread messages are highlighted in bold in this list.



Messages

5 threads

Admin Admin	Expected status in [POST_PROCESSING, PENDING_PROVIDER_REVIEW]. Found [PUBLISHING]	
Admin Admin	Asset Found [PUBLISHING]	12 Dec 22 • 18:27
Admin Admin	This is a response to the test message!	29 Nov 22 • 18:23
Admin Admin	Message from Helpdesk	29 Nov 22 • 12:24
Admin Admin	This is a test message!	23 Nov 22 • 15:28

Figure 77: Messages in a user's mailbox

By clicking on the button "*New Message*", the user can edit a new message to the platform (Topio Helpdesk) with all its details (subject, body text) as shown in Figure 78. The user is acting as a sender, indicated with the icon on the right (enclosed in a red box). By clicking on the *Send* button at the bottom, the message is dispatched through the platform.

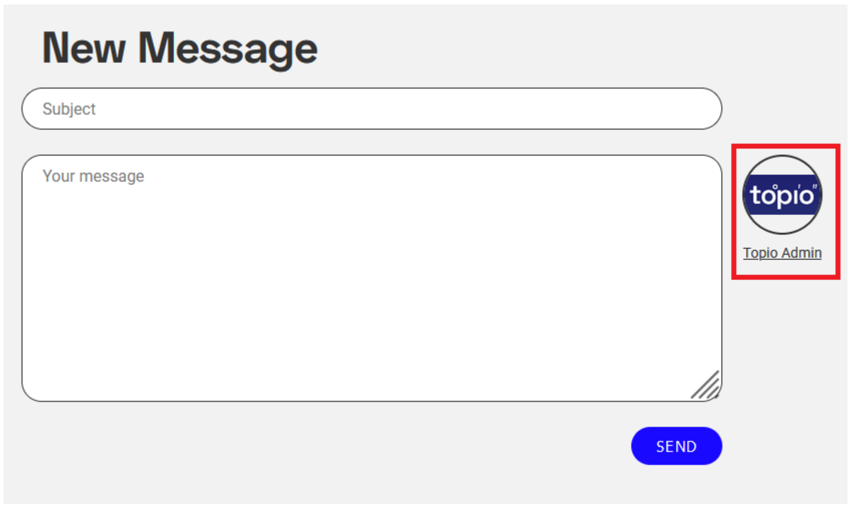
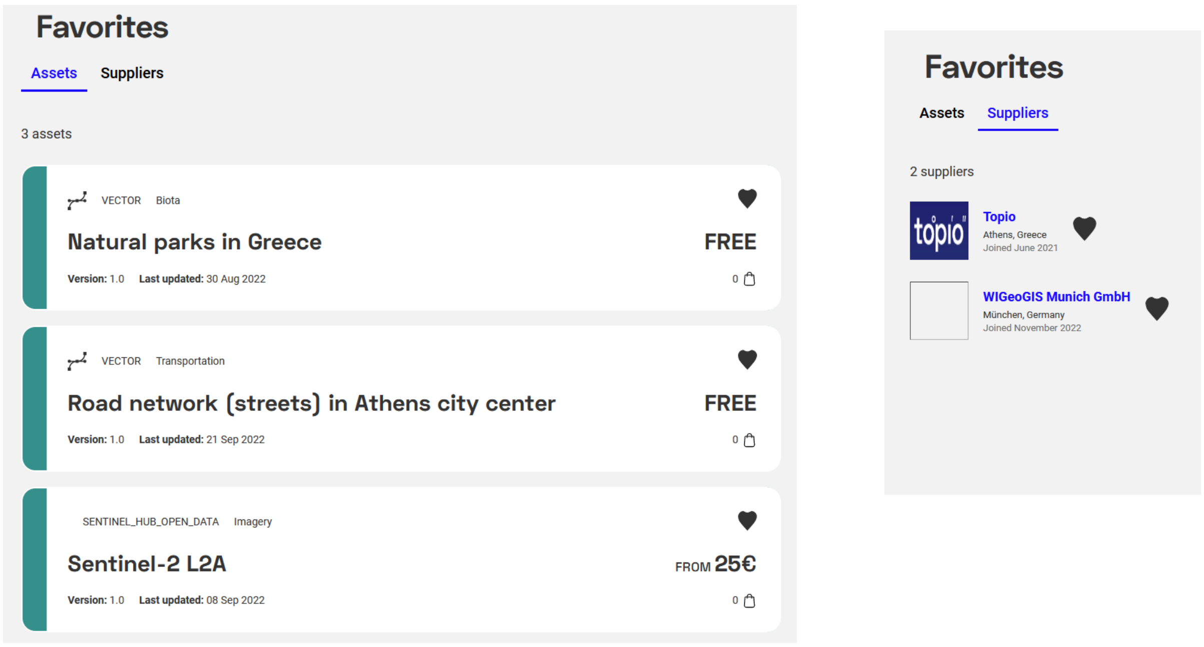


Figure 78: Editing a new message

2.7.4. Favorites

With the *Favorites* option in her dashboard, a user can alternatively view either the list of her favorite *assets* or her favorite *suppliers*, as depicted in Figure 79. Each list is populated with the user's own choices, collected when marking with the *heart* button ♥ a particular asset or supplier she likes while visiting the marketplace as a *registered user*. By clicking on a given asset in the list, the user opens the respective Asset View page. If the user changes her mind and no longer favors an asset or supplier, she can directly remove them from her favorites by toggling (deactivating) the *heart* button of that item in her list.



Favorites

Assets Suppliers

3 assets

VECTOR Biota
Natural parks in Greece — FREE
Version: 1.0 Last updated: 30 Aug 2022 — 0

VECTOR Transportation
Road network (streets) in Athens city center — FREE
Version: 1.0 Last updated: 21 Sep 2022 — 0

SENTINEL_HUB_OPEN_DATA Imagery
Sentinel-2 L2A — FROM 25€
Version: 1.0 Last updated: 08 Sep 2022 — 0

Favorites

Assets Suppliers

2 suppliers

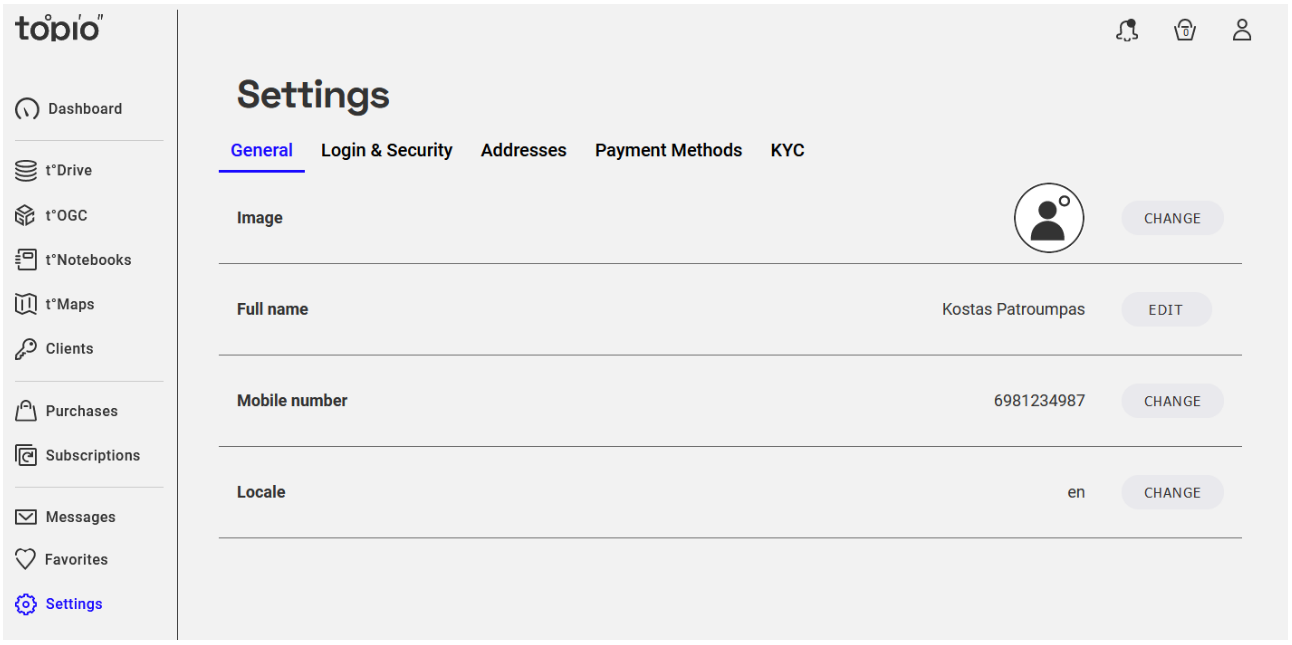
Topio
Athens, Greece
Joined June 2021

WIGeoGIS Munich GmbH
München, Germany
Joined November 2022

Figure 79: User' Favorites: (a) Assets (b) Suppliers

2.7.5. Settings

Once a user registers to the marketplace, she must provide all required information as detailed in Section 2.2.2. Afterwards, all this information is visible to her, but also editable under the *Settings* option in her dashboard, as depicted in Figure 80. Apart from general information, a user can view or modify her profile in the platform, including personal information, login credentials, address(es), payment method(s), and company information if the user is registered as a professional and not an individual.



Settings

General | Login & Security | Addresses | Payment Methods | KYC

Image		CHANGE
Full name	Kostas Patroumpas	EDIT
Mobile number	6981234987	CHANGE
Locale	en	CHANGE

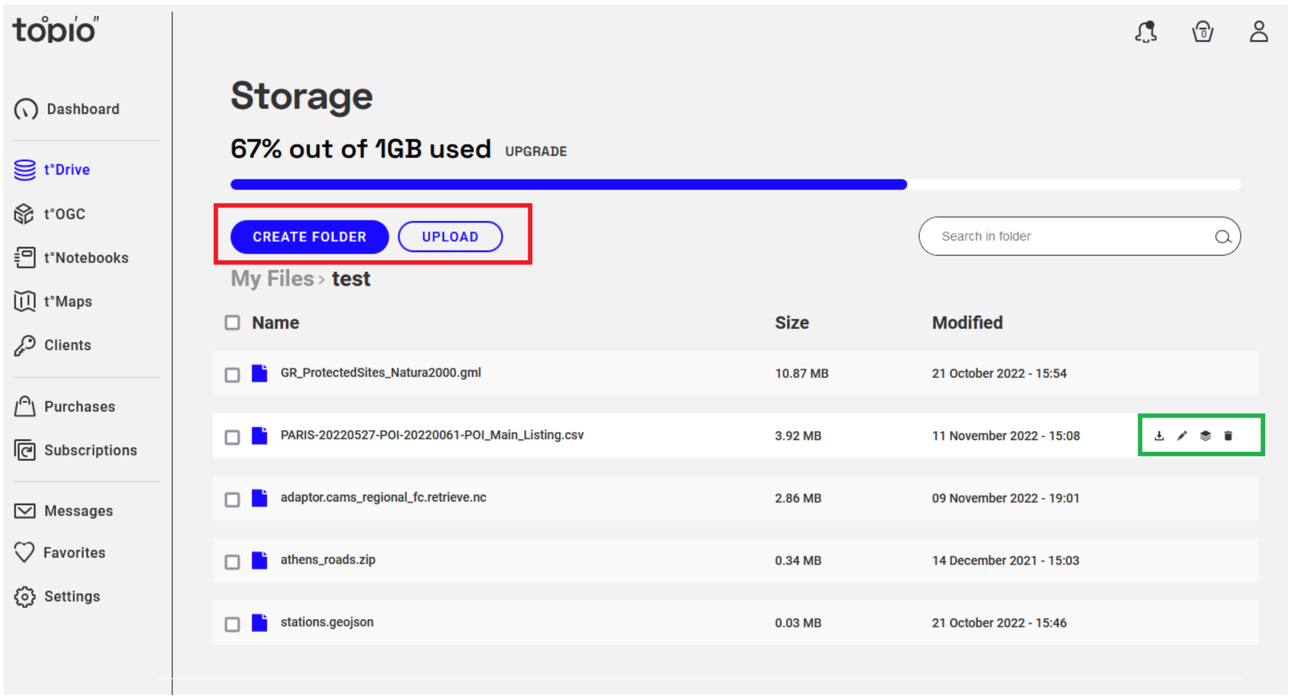
Figure 80: User settings

More specifically, all this functionality is offered via the following tabs under the settings:

- *General*. This tab shows information like a logo icon, the full name, the phone number of the user, and the locale. Each of these values can be edited.
- *Login & Security*. Though this tab, the user can view and modify her *credentials* for signing in the marketplace. Apart from a *password*, the user can also employ *two-step verification* for improved protection by making it more difficult for someone else to sign in the platform with her account. Besides the password, a contact method (also known as security info) is used, typically via *mobile phone* or *email*. If the user turns on two-step verification, she will get a security code to her email or mobile phone every time she signs in on a device that is not trusted.
- *Company information*. This includes the company name, its VAT, the domain (sector where this company is active), and the country it is registered in.
- *Addresses*. This shows at least one postal address (and possibly more than one) where the user is located in.
- *Payment methods*. With this tab, the user can view or specify how she will be paying for assets purchased through the marketplace. At least one payment method must be specified to enable the user to acquire assets and make transactions in the marketplace.
- *KYC*. This *Know Your Customer* step is *optional* for consumers in the platform, but it is *mandatory* for suppliers, and is set as detailed in Section 2.8.11. A notification (the red bullet) is shown if this step has not been completed, but it does not preclude a consumer from purchasing assets through the marketplace.

2.7.6. Topio Drive

Topio Drive is an *embedded* value-added service, integrated into the marketplace and its interface, and a value-multiplier for the provision of other value-added services. Essentially, it provides users with storage space to upload and maintain their data assets. As shown in Figure 81 , it resembles an interactive web front-end for managing and navigating through the storage space in a remote server. It enables the user to *create subfolders* within its allocated space and *upload files* (as indicated with the two buttons in the red box), and search for a file through the search bar. The user can navigate in the various folders and for a chosen file can apply several actions (*Download*, *Rename*, *Create private WMS/WFS service*, or *Delete*) as indicated with the buttons shown in the green box. The uploaded data files can be published as assets through the publication wizard or can be integrated with Topio APIs (WMS, WFS), Topio maps, and Topio Jupyter notebooks created by the user in the platform.



Storage

67% out of 1GB used

My Files > test

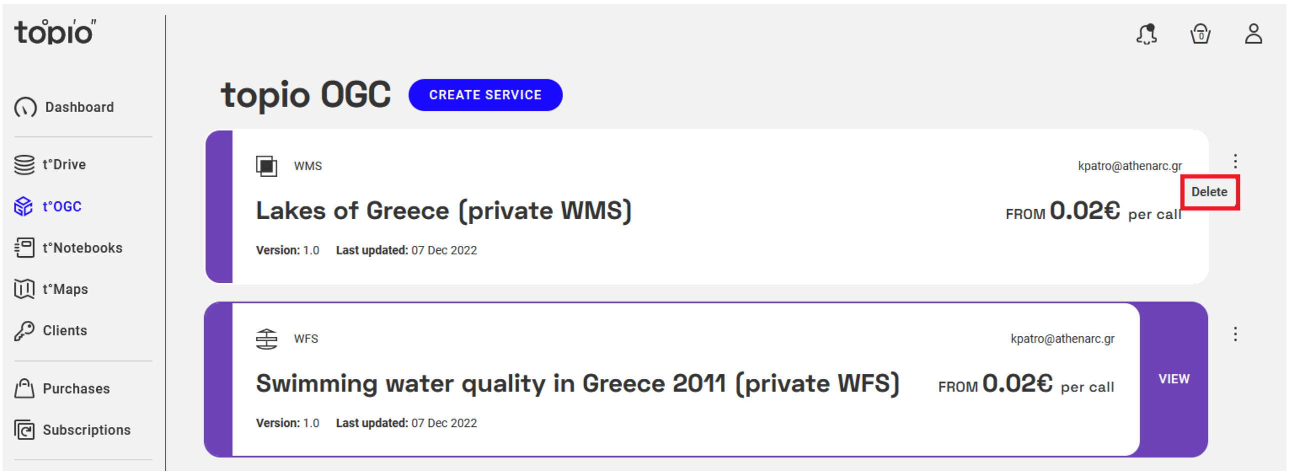
Name	Size	Modified
GR_ProtectedSites_Natura2000.gml	10.87 MB	21 October 2022 - 15:54
PARIS-20220527-POI-20220061-POI_Main_Listing.csv	3.92 MB	11 November 2022 - 15:08
adaptor.cams_regional_fc.retrieve.nc	2.86 MB	09 November 2022 - 19:01
athens_roads.zip	0.34 MB	14 December 2021 - 15:03
stations.geojson	0.03 MB	21 October 2022 - 15:46

Figure 81: Topio Drive of a user

2.7.7. Topio OGC

A consumer can create *private APIs* that offer OGC services (WMS, WFS) based on her data files or assets. Users must pay a subscription in order to use the platform's infrastructure and create such a service. Such OGC services are not publicly available, but only the user can invoke them through other value-added services in the platform (e.g., Topio Notebooks, Topio Maps). By clicking on menu *Topio OGC* from the sidebar, the user can list all the OGC services she has created and can use (Figure 82). Also, by clicking on the *Delete* button that appears next to such an item in the list, she can delete the service.

By clicking on button "*Create Service*", a user-friendly wizard opens and guides the user to provide some information about the service: its type (WMS or WFS), the geospatial file asset that will be used as its layer, some metadata (title, version, spatial reference system, encoding), and finally agree to the terms and the subscription cost to be paid to the platform. Soon afterwards, the service is launched and it is available for use.

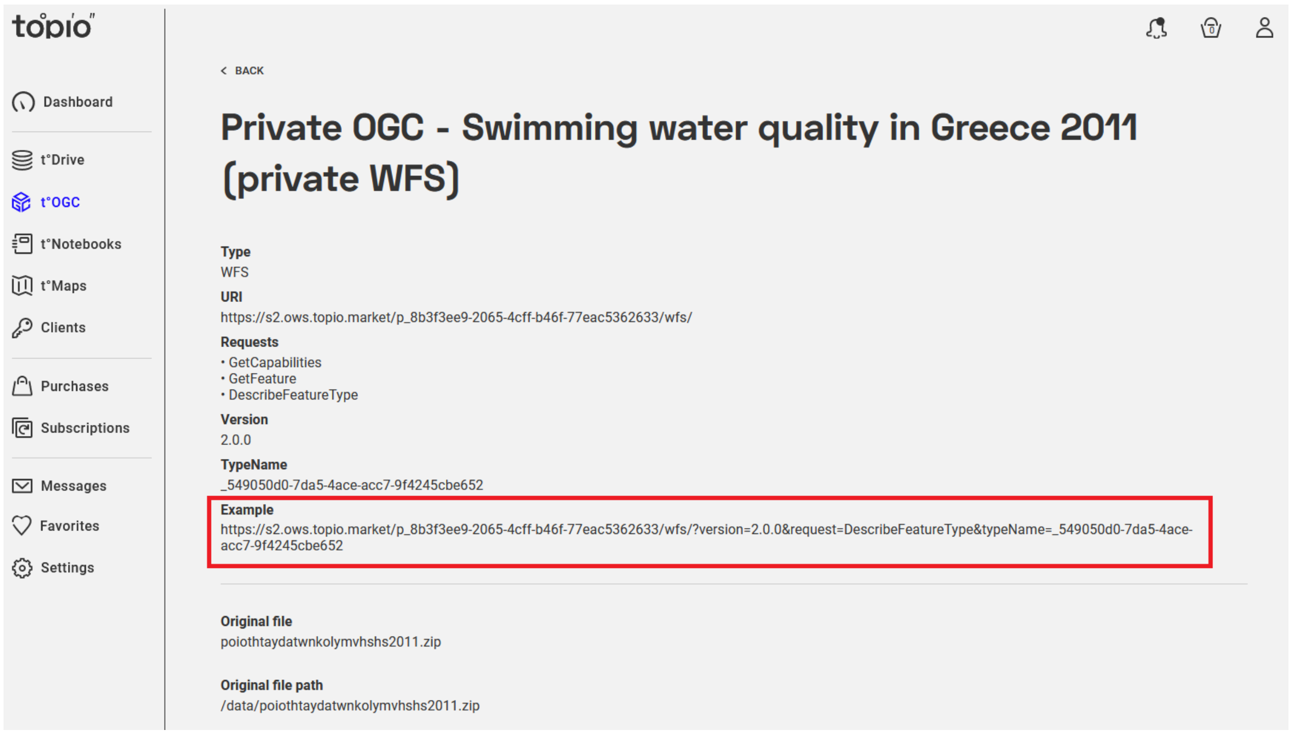


WMS kpatro@athenarc.gr
Lakes of Greece (private WMS) FROM 0.02€ per call
Version: 1.0 Last updated: 07 Dec 2022

WFS kpatro@athenarc.gr
Swimming water quality in Greece 2011 (private WFS) FROM 0.02€ per call
Version: 1.0 Last updated: 07 Dec 2022

Figure 82: Private OGC services

When the user clicks on one of these services in the list, she can view all its details, as shown in Figure 83. This provides the full information on the URI where this OGC service is accessible, the supported requests, the type and version of the services, as well as the name of the layer (type name), the original file name and its path on her Topio Drive storage. An example request is also given, so that the user can directly invoke it and verify the status of the service. Note that in order to use such a service, the user must create access credentials (as discussed in Section 2.7.10) and then provide them to the applications (e.g., Topio Notebooks) that will make API requests on behalf of the user.



Private OGC - Swimming water quality in Greece 2011 (private WFS)

Type
WFS

URI
https://s2.ows.topio.market/p_8b3f3ee9-2065-4cff-b46f-77eac5362633/wfs/

Requests
- GetCapabilities
- GetFeature
- DescribeFeatureType

Version
2.0.0

TypeName
_549050d0-7da5-4ace-acc7-9f4245cbe652

Example
https://s2.ows.topio.market/p_8b3f3ee9-2065-4cff-b46f-77eac5362633/wfs/?version=2.0.0&request=DescribeFeatureType&typeName=_549050d0-7da5-4ace-acc7-9f4245cbe652

Original file
poiothtaydatwnkolymvhshs2011.zip

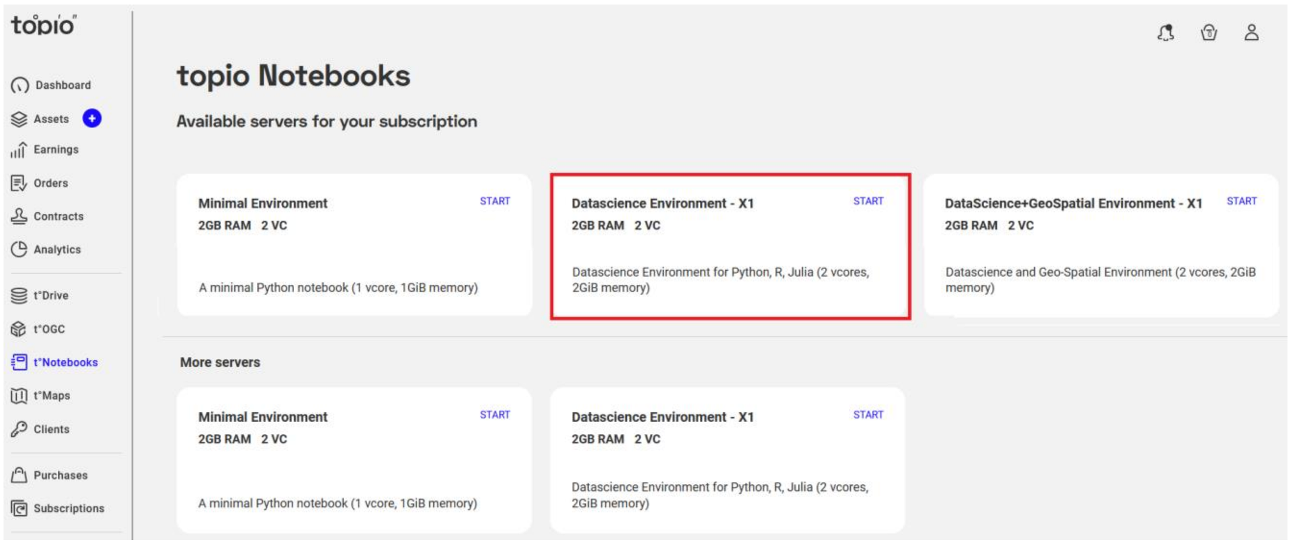
Original file path
/data/poiothtaydatwnkolymvhshs2011.zip

Figure 83: Details of a private OGC service (WFS)

2.7.8. Topio Notebooks

Topio Notebooks is an integrated Jupyter Hub deployment, provided as a value-added application to Topio users. Topio Notebooks is a streamlined Jupyter notebook environment focused on supporting geospatial data science. Users can enjoy ready-to-use geospatial processing and visualization libraries, integrated access to Topio data and Topio APIs, domain-specific templates, and runtime environments curated by and for geospatial data scientists. Topio Notebooks aims to be useful for non-experts, for all tasks related to geospatial analyses; from simple data cleaning and enrichment, to geocoding and trend detection, up to analyzing satellite imagery.

Through the Dashboard, the user can explore the offerings available to her subscription (Figure 84). The view is split into two panes: the top includes the servers included in the user's current subscription tier, while the bottom presents additional servers available at a higher tier. From this environment, the user can start/stop individual servers and open them, at which point the user is returned to the active notebook. The exact number of active images and limits (total active hours, total processing hours) is established by the user's subscription tier.



topio Notebooks

Available servers for your subscription

Minimal Environment START 2GB RAM 2 VC A minimal Python notebook (1 vcore, 1GiB memory)	Datascience Environment - X1 START 2GB RAM 2 VC Datascience Environment for Python, R, Julia (2 vcores, 2GiB memory)	DataScience+GeoSpatial Environment - X1 START 2GB RAM 2 VC Datascience and Geo-Spatial Environment (2 vcores, 2GiB memory)

More servers

Minimal Environment START 2GB RAM 2 VC A minimal Python notebook (1 vcore, 1GiB memory)	Datascience Environment - X1 START 2GB RAM 2 VC Datascience Environment for Python, R, Julia (2 vcores, 2GiB memory)

Figure 84: Topio Notebooks

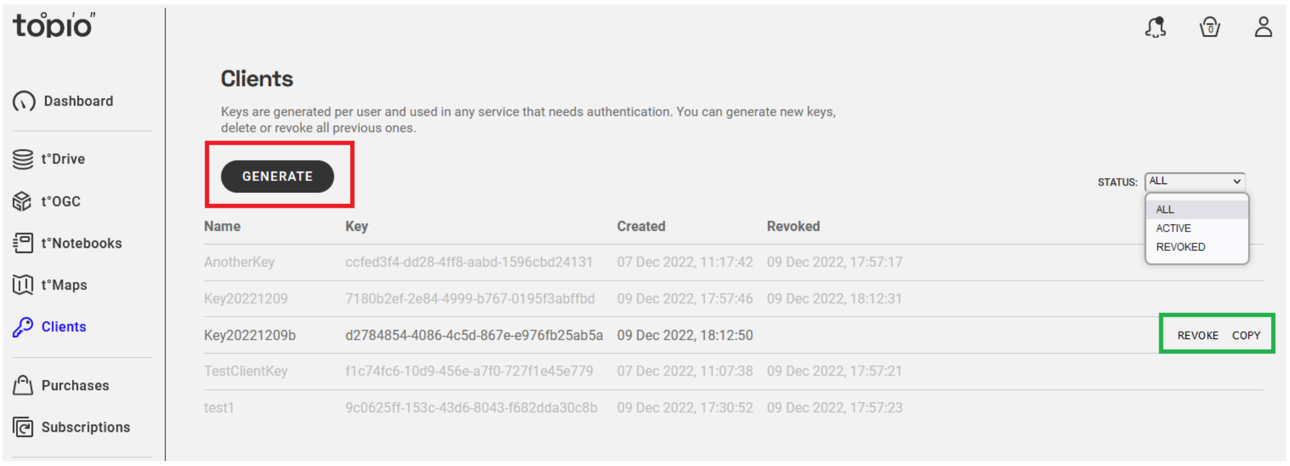
2.7.9. Topio Maps

Topio Maps is a comprehensive framework for creating, using, sharing, and integrating interactive maps in web and mobile applications. Topio Maps enables users to create custom maps using Topio data and services, beautiful base maps, ready to use styles. Users can access their own or purchased geospatial data to publish spatial content as web maps, dashboards or geostories.

2.7.10. Clients

The menu *Clients*, in the sidebar of the Dashboard, allows a user to generate client credentials and view existing ones. By clicking on the button "*Generate*", a user can create new credentials. This is actually a pair of two items, namely a *client ID* (listed under column *Key* in Figure 85) and a *client secret* that appears only once (when new credentials are generated). The user must keep credentials in a safe place as confidential data. These client credentials are then used to obtain access tokens from Topio Identity Server[61] using the OAuth-2.0 "client-credentials" flow. These access tokens can be employed by Topio applications (e.g., Topio Maps, Topio Notebooks) in order to make API requests to OGC services (Section 2.7.7) on behalf of the user. An access token represents the authorization of the client application to access specific parts of the user's data. Access tokens must also be kept confidential and can only be used over an HTTPS connection. They are short-lived tokens (expire after some days), are generated per user and can be used in any service of the platform that requires authentication. Users can generate new access tokens or revoke existing ones according to their needs.

[61] https://accounts.topio.market



Clients

Keys are generated per user and used in any service that needs authentication. You can generate new keys, delete or revoke all previous ones.

Name	Key	Created	Revoked
AnotherKey	ccfed3f4-dd28-4ff8-aabd-1596cbd24131	07 Dec 2022, 11:17:42	09 Dec 2022, 17:57:17
Key20221209	7180b2ef-2e84-4999-b767-0195f3abffbd	09 Dec 2022, 17:57:46	09 Dec 2022, 18:12:31
Key20221209b	d2784854-4086-4c5d-867e-e976fb25ab5a	09 Dec 2022, 18:12:50	
TestClientKey	f1c74fc6-10d9-456e-a7f0-727f1e45e779	07 Dec 2022, 11:07:38	09 Dec 2022, 17:57:21
test1	9c0625ff-153c-43d6-8043-f682dda30c8b	09 Dec 2022, 17:30:52	09 Dec 2022, 17:57:23

Figure 85: Client credentials for accessing Topio services

2.8. Supplier Dashboard

2.8.1. Overview

Once a supplier signs in the platform, the main dashboard page shows an updated overview of the latest developments regarding her assets. As exemplified in Figure 86, the supplier can be informed about her total profits, a breakdown of her assets (data files, APIs, etc.) offered through the marketplace, along with charts that reflect the trend of her earnings and evolution of her sales in various market segments. The supplier can choose the *time period* of reference for the displayed figures, e.g., last month, last year, etc. as indicated with the options in the red box. This page also includes the list of her most *popular assets* for a time period of her choice. Upon changing the time reference in a panel, the respective contents get updated accordingly. As illustrated with the options within the green box in Figure 86, for each of these panels in the dashboard, she can choose to download the respective statistics as a file, or even remove the panel from her dashboard.

On the sidebar menu of the dashboard, several options offer more detailed information to a supplier about the activity in her assets in the marketplace, as detailed next. Note that all this information is confidential and available only to the specific supplier concerning her own assets.

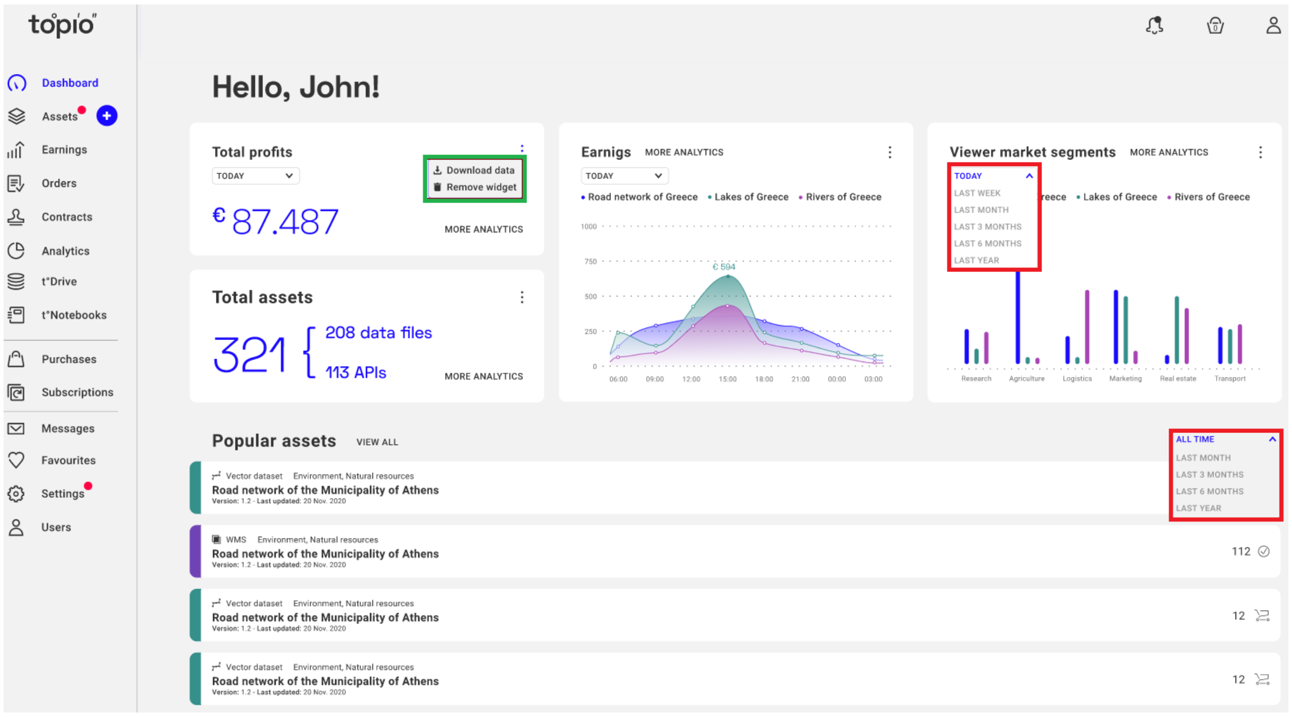
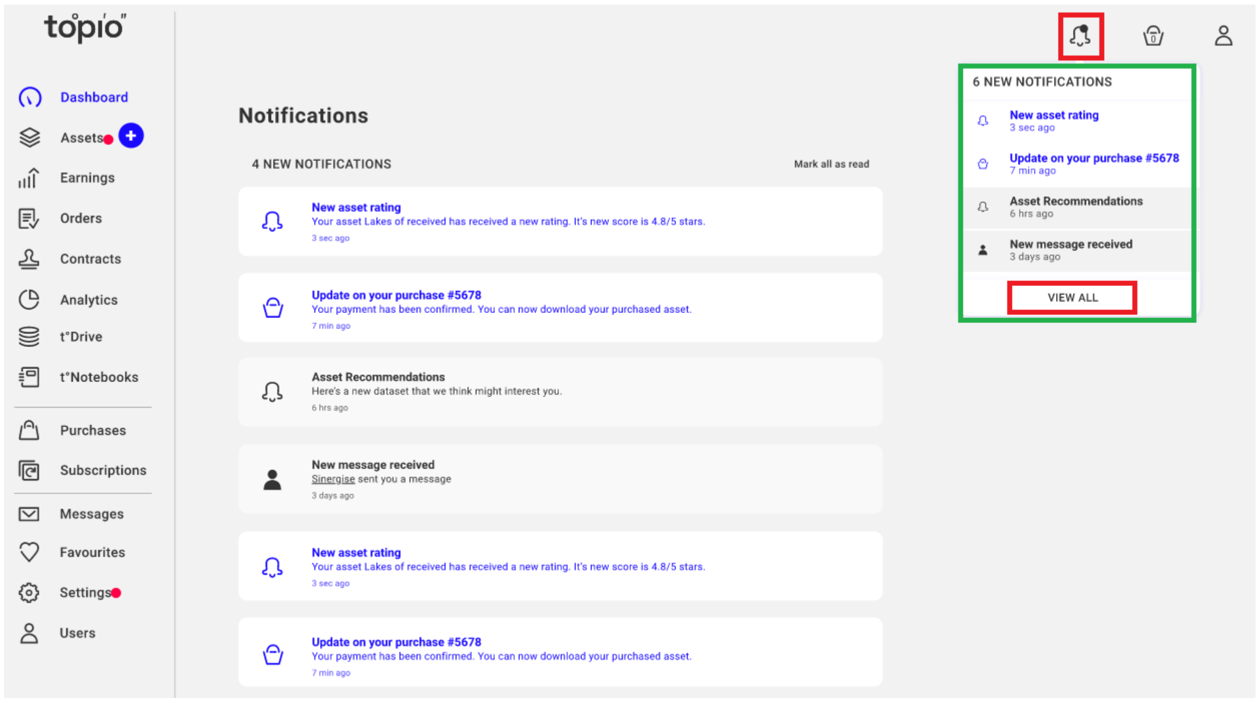


Figure 86: Supplier's dashboard

Once signed in, the supplier can use the icons shown in the top left corner of the dashboard and respectively check any notifications, her cart, or her platform account. Note that these icons are permanently visible as long as the supplier is signed in the platform, enabling her quick access to the respective functionality. More specifically:

- By clicking on the *notifications* icon (the bell icon shown within a green box in Figure 87), the supplier can get a *quick overview* of recent notifications issued by the platform regarding her account, as listed in the green box. By clicking on the "*View all*" button, she can display the full list, with those unread clearly marked (shown in blue color). Such notifications may concern messages (e.g., a new rating for one of her assets or recommendations), but also pending actions regarding an asset (e.g., steps to complete its publication in the marketplace or issues raised during its review by the Helpdesk). Such notifications may also concern purchases made through the platform, e.g., an order she has placed (acting as a user in the platform) for an asset available from another supplier.
- Clicking on the *cart* icon, the supplier can get an overview of the items currently in her cart; note that this concerns assets available from other suppliers. Of course, she can view all the contents of her cart, make modifications (add/remove assets) etc.



Notifications

4 NEW NOTIFICATIONS — Mark all as read

New asset rating
Your asset Lakes of received has received a new rating. It's new score is 4.8/5 stars.
3 sec ago

Update on your purchase #5678
Your payment has been confirmed. You can now download your purchased asset.
7 min ago

Asset Recommendations
Here's a new dataset that we think might interest you.
6 hrs ago

New message received
Sinergise sent you a message
3 days ago

New asset rating
Your asset Lakes of received has received a new rating. It's new score is 4.8/5 stars.
3 sec ago

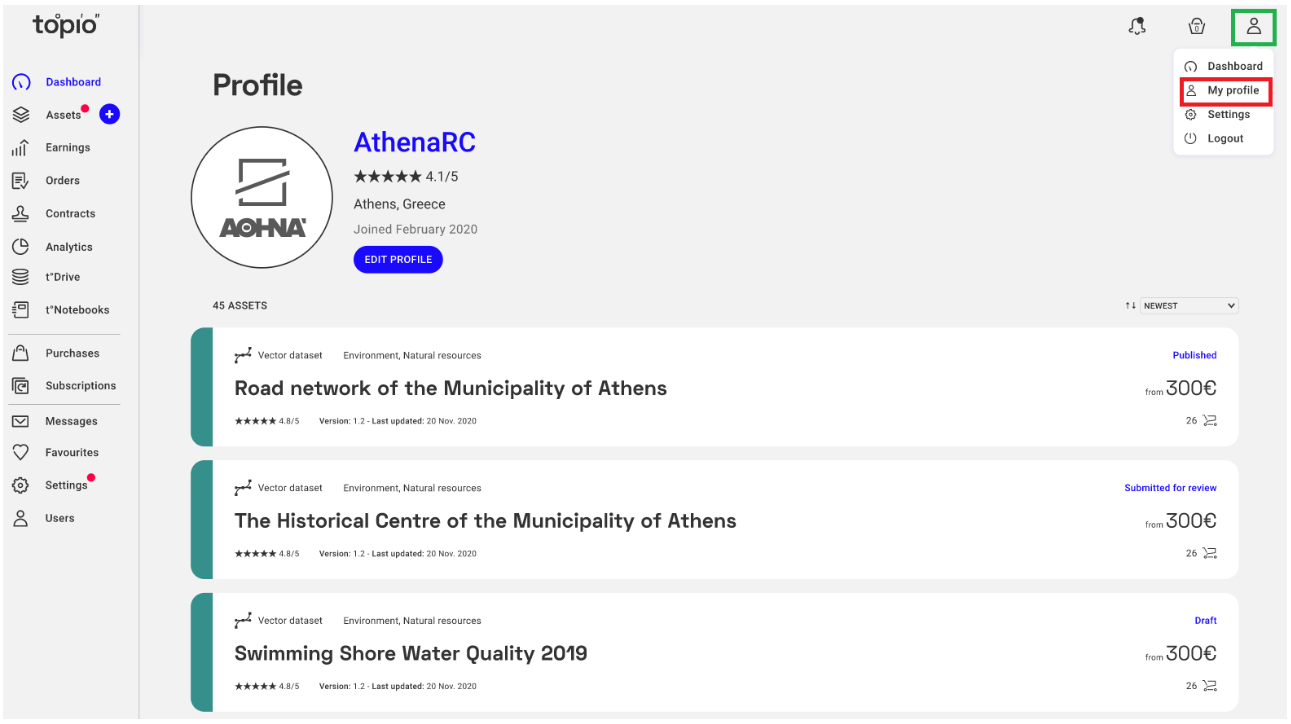
Update on your purchase #5678
Your payment has been confirmed. You can now download your purchased asset.
7 min ago

6 NEW NOTIFICATIONS

New asset rating — 3 sec ago
Update on your purchase #5678 — 7 min ago
Asset Recommendations — 6 hrs ago
New message received — 3 days ago

VIEW ALL

Figure 87: Notifications to a supplier



Profile

AthenaRC
★★★★★ 4.1/5
Athens, Greece
Joined February 2020
EDIT PROFILE

45 ASSETS — NEWEST

Vector dataset — Environment, Natural resources — Published
Road network of the Municipality of Athens
★★★★★ 4.8/5 — Version: 1.2 · Last updated: 20 Nov. 2020 — from 300€ — 26

Vector dataset — Environment, Natural resources — Submitted for review
The Historical Centre of the Municipality of Athens
★★★★★ 4.8/5 — Version: 1.2 · Last updated: 20 Nov. 2020 — from 300€ — 26

Vector dataset — Environment, Natural resources — Draft
Swimming Shore Water Quality 2019
★★★★★ 4.8/5 — Version: 1.2 · Last updated: 20 Nov. 2020 — from 300€ — 26

Figure 88: Supplier's profile

2.8.2. Assets

Clicking on the option *Assets* in the sidebar, a supplier gets two separate lists of her assets as shown in Figure 89:

- On top, there is a list with her *unpublished assets*: those that are still draft, submitted, rejected, currently under review by the Helpdesk, etc. The supplier can filter this list by the asset *status* (options indicated in the red box) and order the items by various *sorting criteria* (by title, status, date created or modified, as shown within the green box). The supplier can open a *Draft* asset, inspect or modify its settings (metadata, pricing models, delivery, etc.) using a wizard, and finally submit it for publication.
- Below, there is another list with *all published assets* currently offered by this supplier through the marketplace. The supplier can choose to view all her assets or specific categories (data files, based on Topio APIs or her own APIs), as denoted with the options in the orange box; for each category, the number of total assets per category is also shown. The supplier can view her published assets in ascending/descending order by several *sorting criteria* (title, date, type). Next to each asset in this list, the supplier can click on the *More* button (denoted with the dots icon) and directly perform specific actions on that asset (*Edit, Create WMS, Create WFS, Delete*) as indicated within the blue box. Also note that on top of this list,

In addition, the supplier can add a new asset to those offered in the marketplace. By either clicking on the + button in the sidebar or the button "*Add an Asset*", she can publish a new asset with the wizard. Of course, when she hovers the mouse over an asset, she can click on this item and open the *Asset View* page with the full information about this asset (Section 2.4).

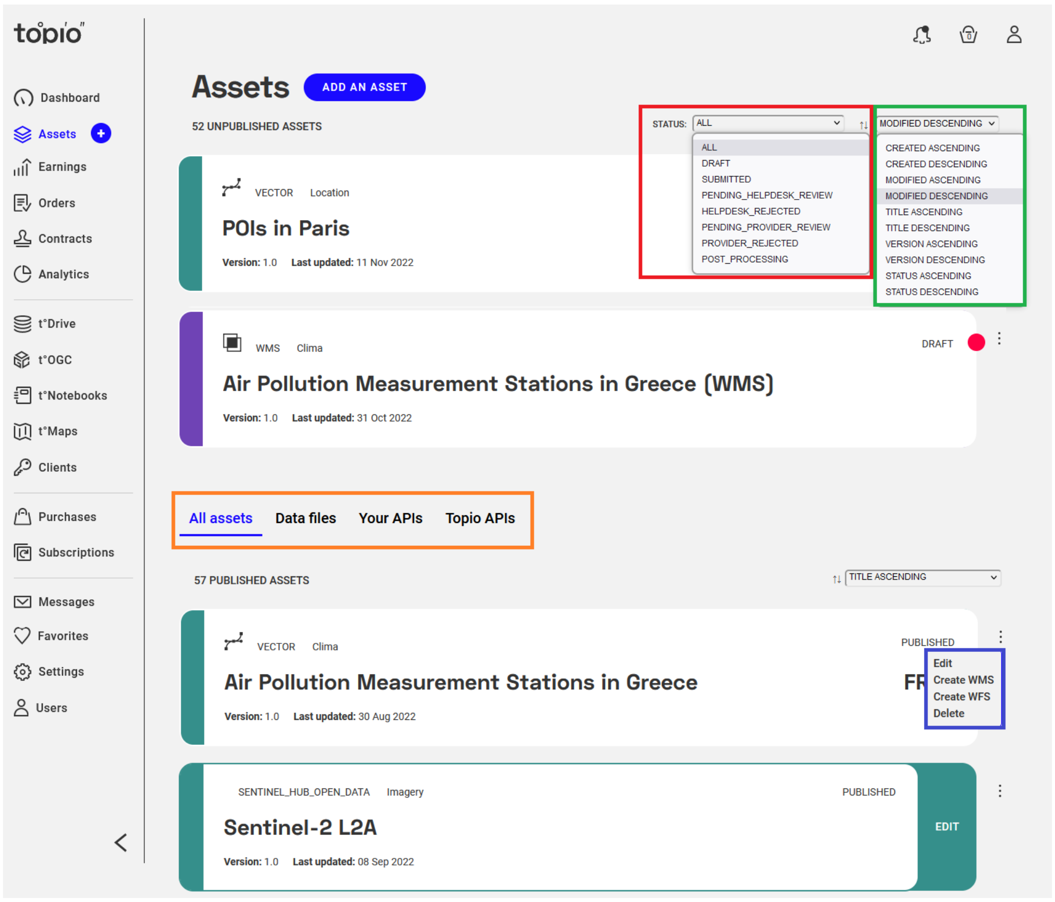
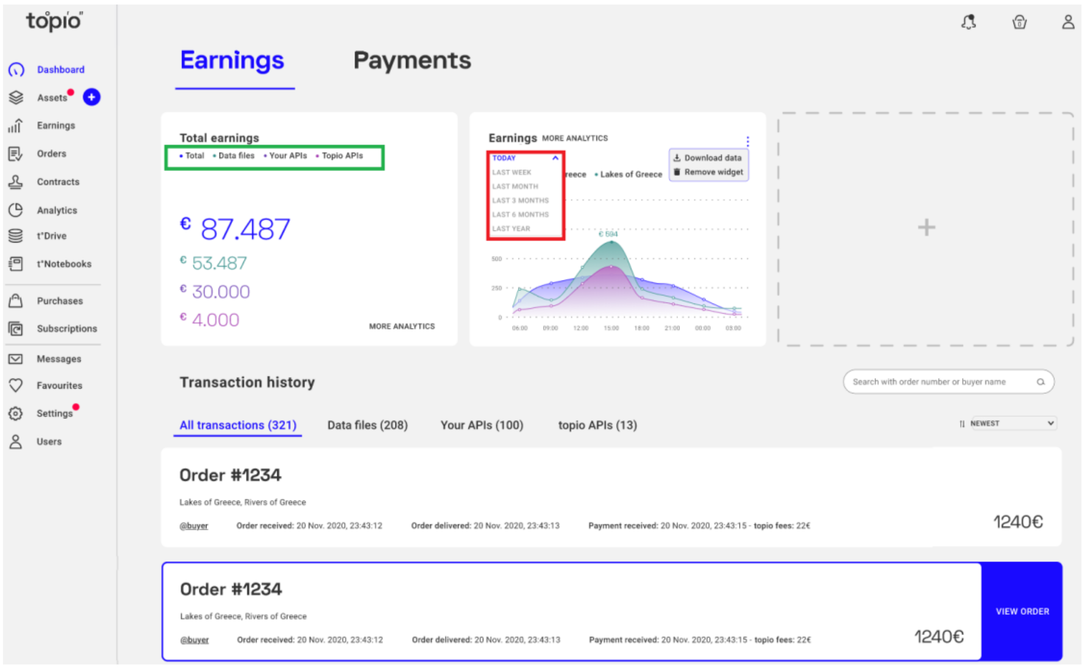


Figure 89: Assets offered by a given supplier

2.8.3. Earnings/Payments

With the *Earnings* option in her dashboard, a supplier can obtain a detailed view of her revenues, examine how her turnover evolves over time, and inspect the complete history of transactions involving orders placed by customers of her own assets (Figure 90). Note that amounts shown in the *earnings* tab represent (i) orders that have been completed and their *payments* received, as well as (ii) amounts due to be received by the platform. For instance, revenues from API assets based on number of requests may be collected and reimbursed periodically (e.g., every month) or when they exceed a significant amount (e.g., 100 euros) to avoid bank surcharges. In contrast, the *payments* tab shows only amounts actually reimbursed, as well as any due payments.

Regarding *earnings* (Figure 90), the supplier may choose to view the total amount or specify a particular type of assets (e.g., Data files, APIs) as indicated in the green box; this view is then refreshed accordingly. Also, the supplier can specify the *time period* of reference (indicative options shown in the red box) for calculating the related trends and statistics. By clicking on button "*More Analytics*", she can get a more detailed analysis of her earnings as discussed next in Section 2.8.6. In the *transaction history*, she gets a list of all transactions concerning her assets; the supplier can also search over her list of transactions, by specifying some criteria (e.g., keywords, price range) on the search bar on top of the list, which is refreshed accordingly. Upon clicking a particular item in the list, the order details are displayed as in Figure 93.

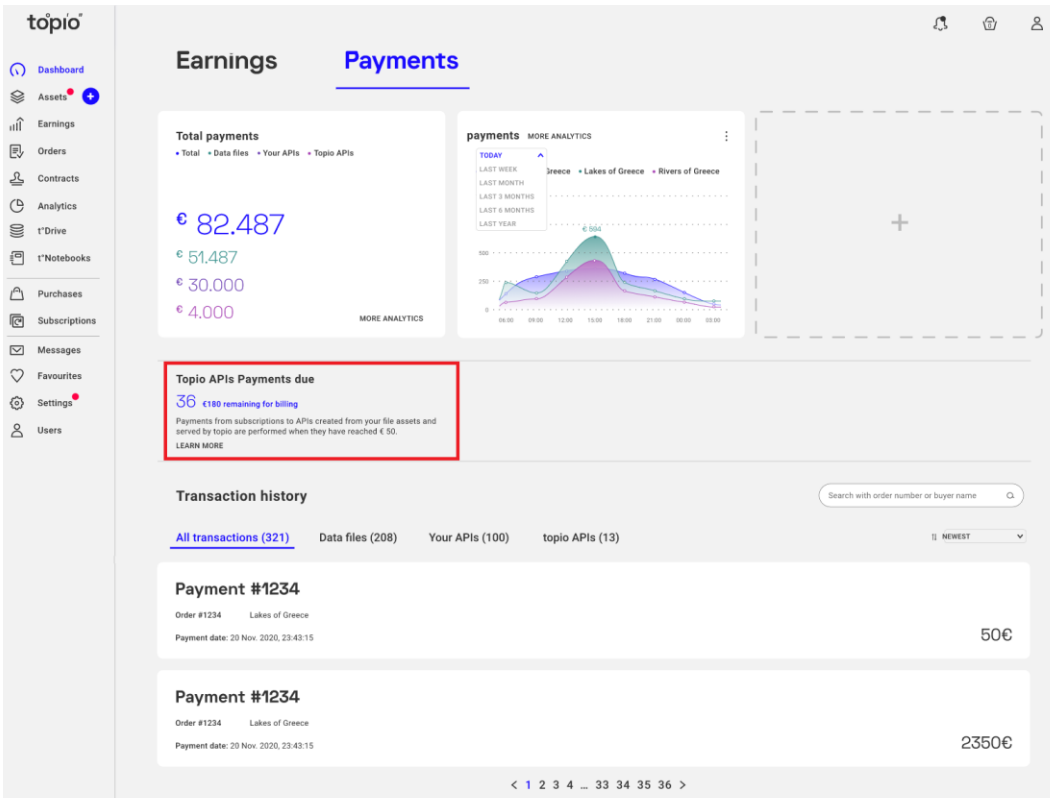


Order #1234
Lakes of Greece, Rivers of Greece
@buyer Order received: 20 Nov. 2020, 23:43:12 Order delivered: 20 Nov. 2020, 23:43:13 Payment received: 20 Nov. 2020, 23:43:15 - topio fees: 22€ 1240€

Order #1234
Lakes of Greece, Rivers of Greece
@buyer Order received: 20 Nov. 2020, 23:43:12 Order delivered: 20 Nov. 2020, 23:43:13 Payment received: 20 Nov. 2020, 23:43:15 - topio fees: 22€ 1240€

Figure 90: Earnings and transaction history of a supplier

If the supplier clicks on the *Payments* tab, a similar view is shown regarding cleared payments. The same functionality offered for *Earnings* is also available in the case of payments, with one extra feature (shown in the red box), which indicates the *due payments*. E.g., this may concern payments from subscriptions to APIs based on this supplier's assets. Displayed information includes the number of such owing payments, the total amount, as well as any related notifications (e.g., when the payment can be effectuated).



Total payments
• Total • Data files • Your APIs • Topio APIs

€ 82.487
€ 51.487
€ 30.000
€ 4.000
MORE ANALYTICS

payments MORE ANALYTICS
TODAY
LAST WEEK
LAST MONTH
LAST 3 MONTHS
LAST 6 MONTHS
LAST YEAR
• Lakes of Greece • Rivers of Greece

Topio APIs Payments due
36 €180 remaining for billing
Payments from subscriptions to APIs created from your file assets and served by topio are performed when they have reached € 50.
LEARN MORE

Transaction history
Search with order number or buyer name

All transactions (321) Data files (208) Your APIs (100) topio APIs (13)
NEWEST

Payment #1234
Order #1234 Lakes of Greece
Payment date: 20 Nov. 2020, 23:43:15
50€

Payment #1234
Order #1234 Lakes of Greece
Payment date: 20 Nov. 2020, 23:43:15
2350€

< 1 2 3 4 … 33 34 35 36 >

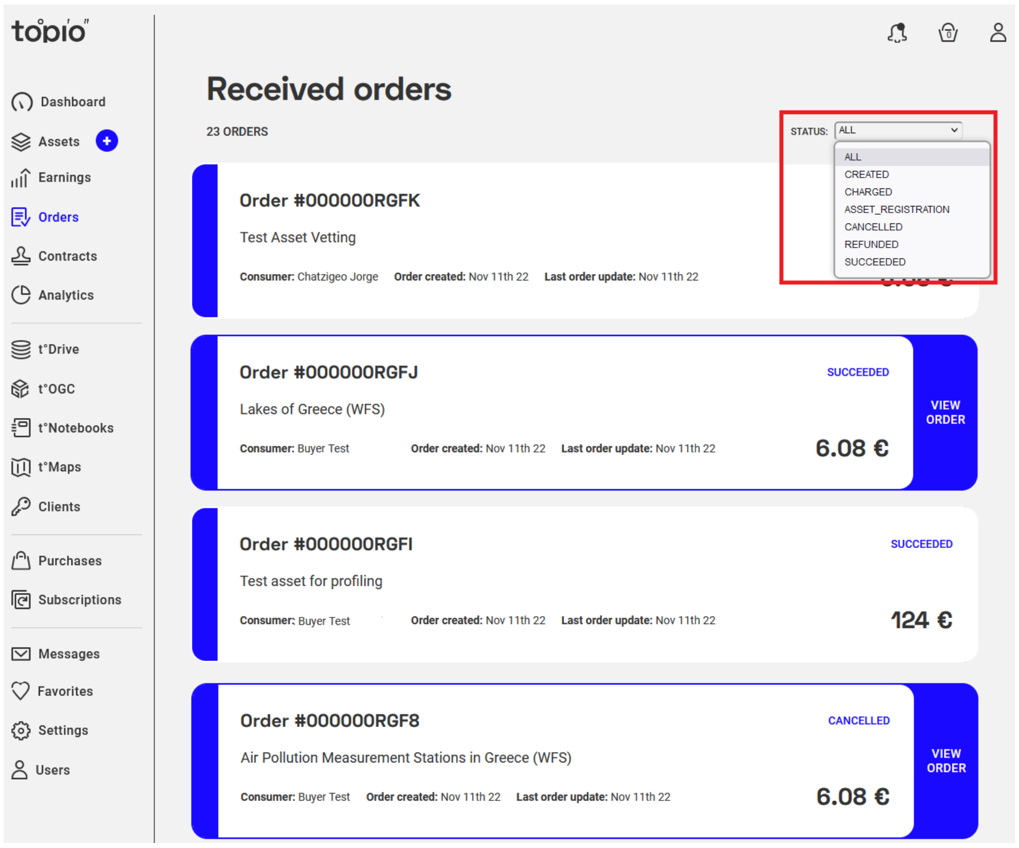
Figure 91: Payments of a supplier

2.8.4. Orders

Option *Orders* in the sidebar of her dashboard gives the supplier the complete list of all the orders placed for her assets (Figure 92). Thus, she can filter these orders by their *status* as shown with the options in the red box: e.g., whether they are received, cancelled, under processing for payment, or succeeded, and the list gets updated accordingly. By default, the supplier can view all orders by reverse chronological order.

Note that some orders may still *require some action* from the supplier; such orders are listed on the top of the list with a red notification next to them as a reminder that something is pending about them. For instance, this may indicate that the order must be approved by the supplier because she had activated the vetting option for this asset (as shown in Figure 71). Subsequently, some orders may be awaiting approval or may be even rejected by the seller.

For each order, certain *actions* can be performed. The supplier can mark an order as complete (thus acknowledging that the assets has been delivered and payment is received), cancel an order (e.g., if the sale is disputed), or even delete it from its records.


topio
Dashboard
Assets
Earnings
Orders
Contracts
Analytics
t°Drive
t°OGC
t°Notebooks
t°Maps
Clients
Purchases
Subscriptions
Messages
Favorites
Settings
Users

Received orders

23 ORDERS

STATUS: ALL

ALL
CREATED
CHARGED
ASSET_REGISTRATION
CANCELLED
REFUNDED
SUCCEEDED

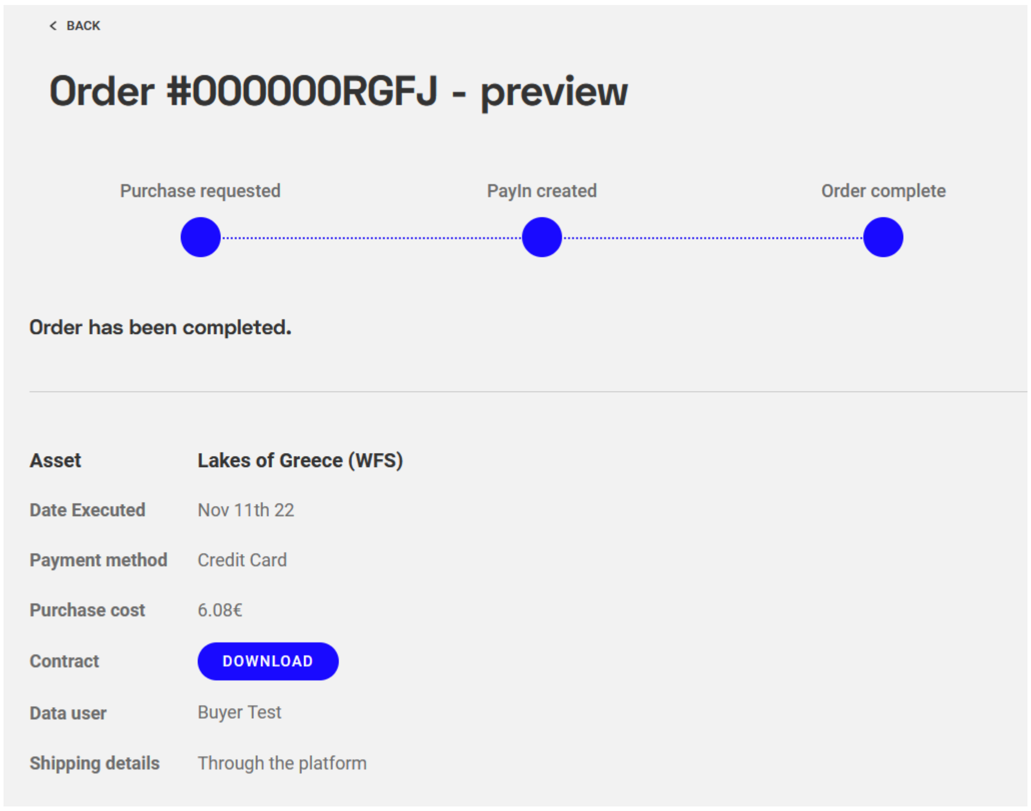
Order #000000RGFK
Test Asset Vetting
Consumer: Chatzigeo Jorge **Order created:** Nov 11th 22 **Last order update:** Nov 11th 22

Order #000000RGFJ SUCCEEDED
Lakes of Greece (WFS)
Consumer: Buyer Test **Order created:** Nov 11th 22 **Last order update:** Nov 11th 22
6.08 €
VIEW ORDER

Order #000000RGFI SUCCEEDED
Test asset for profiling
Consumer: Buyer Test **Order created:** Nov 11th 22 **Last order update:** Nov 11th 22
124 €

Order #000000RGF8 CANCELLED
Air Pollution Measurement Stations in Greece (WFS)
Consumer: Buyer Test **Order created:** Nov 11th 22 **Last order update:** Nov 11th 22
6.08 €
VIEW ORDER

Figure 92: Orders received by a supplier


< BACK

Order #000000RGFJ - preview

Purchase requested — PayIn created — Order complete

Order has been completed.

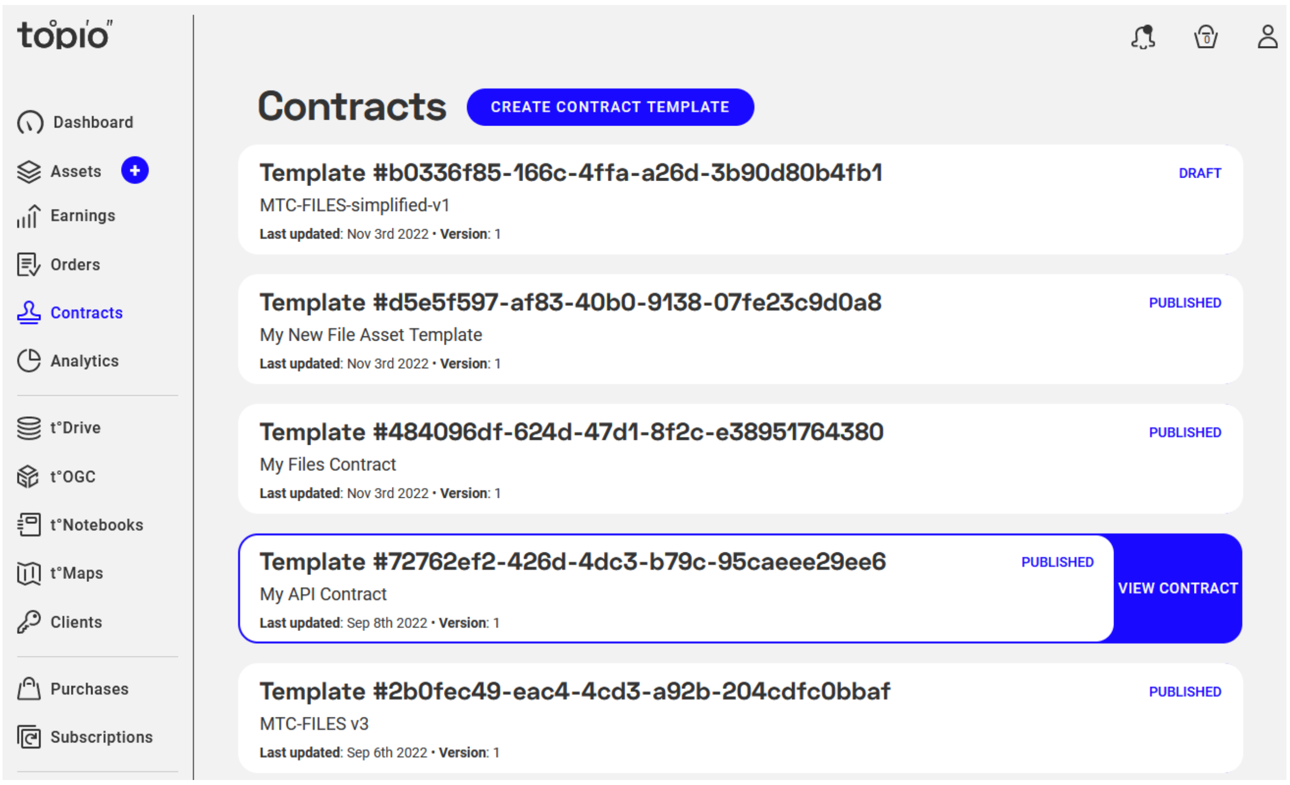
Asset	Lakes of Greece (WFS)
Date Executed	Nov 11th 22
Payment method	Credit Card
Purchase cost	6.08€
Contract	DOWNLOAD
Data user	Buyer Test
Shipping details	Through the platform

Figure 93: Monitoring the lifecycle of an order

By clicking on a particular order in this list, the supplier can monitor its status all along its lifecycle. As shown in Figure 93, she can inspect all stages of an order, examining all its details from the moment it was placed (customer, asset, price, date, etc.) to payment, and up to its completion. The status of this order changes after each step and this is also reflected in the list.

2.8.5. Contract

The *Contracts* option in the dashboard offers a supplier with an overview of the templates specified for licensing and selling her assets through the marketplace. This list (shown in Figure 94) does not refer to particular assets offered by this supplier, but to templates prepared for various types of assets and associated with them at the time of their publication (Section **0**). The list shows all templates specified by this supplier along with their *status* (Published, Inactive, Draft). Choosing a particular contract template from the list, the supplier can modify its status (e.g., deactivate it) and even start editing it. If she chooses to *deactivate* a contract associated with published assets, a warning appears (Figure 95) asking for her confirmation, since the status of those assets will change to *Unpublished* and will become invisible to visitors of the marketplace until they get associated with another contract template for future purchases. Besides, *edits* may involve any section of the contract terms and conditions, in the same flow discussed next for creating new templates.



Contracts

Template #b0336f85-166c-4ffa-a26d-3b90d80b4fb1 — DRAFT
MTC-FILES-simplified-v1
Last updated: Nov 3rd 2022 • **Version**: 1

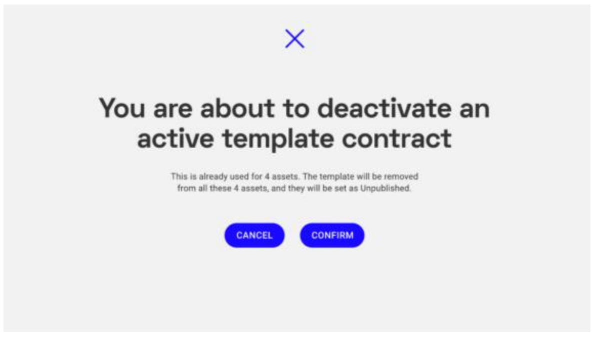
Template #d5e5f597-af83-40b0-9138-07fe23c9d0a8 — PUBLISHED
My New File Asset Template
Last updated: Nov 3rd 2022 • **Version**: 1

Template #484096df-624d-47d1-8f2c-e38951764380 — PUBLISHED
My Files Contract
Last updated: Nov 3rd 2022 • **Version**: 1

Template #72762ef2-426d-4dc3-b79c-95caeee29ee6 — PUBLISHED
My API Contract
Last updated: Sep 8th 2022 • **Version**: 1

Template #2b0fec49-eac4-4cd3-a92b-204cdfc0bbaf — PUBLISHED
MTC-FILES v3
Last updated: Sep 6th 2022 • **Version**: 1

Figure 94: Contract templates specified by the supplier


You are about to deactivate an active template contract

This is already used for 4 assets. The template will be removed from all these 4 assets, and they will be set as Unpublished.

Figure 95: Warning regarding deactivation of a contract template

2.8.6. Analytics

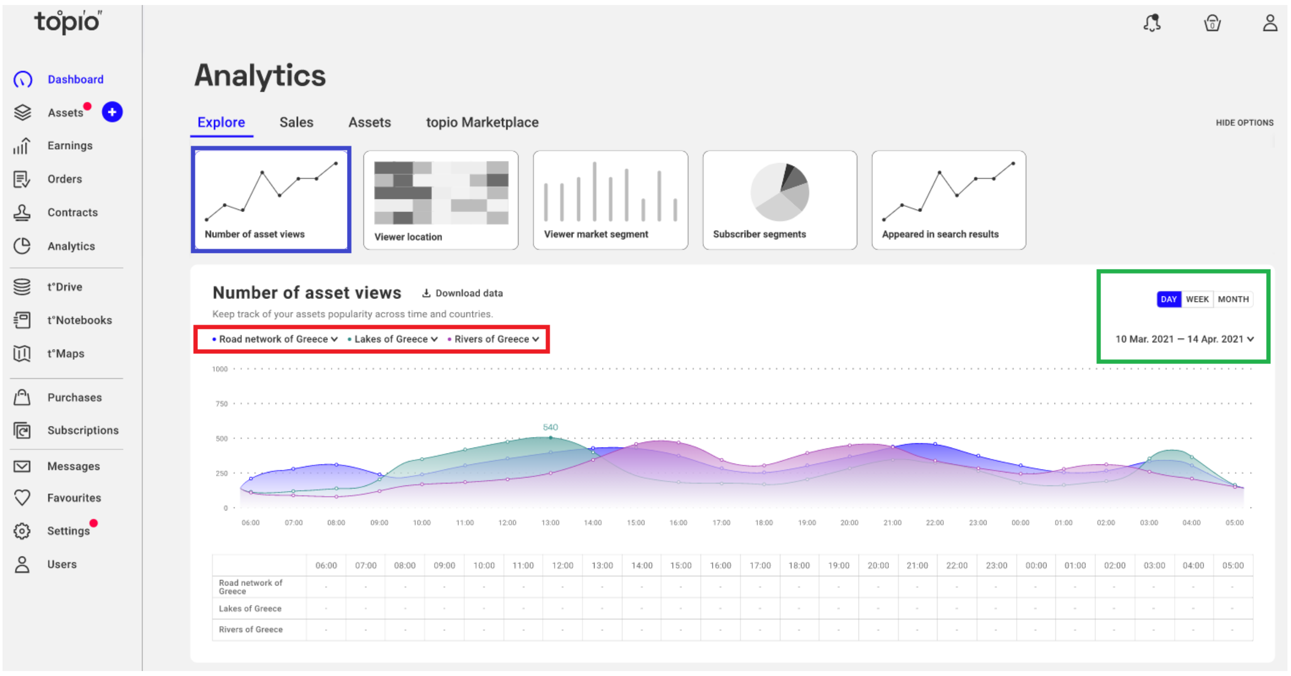
The *Analytics* option in the dashboard provides a wide range of statistics, charts, and diagrams (including line or bar plots, pie charts, heatmaps, and choropleth maps), along with several *filtering* and *aggregation* options that enable the owner to assess the trend of her assets and sales in the marketplace (e.g., per application domain, time period, country, etc.).

Filters can be applied to such analytics and may involve:

- *Asset(s) of interest*: the supplier can specify one or some of her own assets in order to compare their popularity amongst users or their actual sales in the marketplace. In most cases, the supplier can pick up to 3 assets; specifically for asset insights, she can compute analytics involving up to 10 of her assets.
- *Asset type*: this is mostly used in breaking down earnings per type (*data file, API, total*).

Aggregates can be computed for asset views and sales amounts on three possible *dimensions*:

- *Temporal*: the supplier can specify the time granularity of interest (day, week, or month);
- *Spatial*: the supplier can specify whether she wants a breakdown of the aggregated values by EU country;
- *Segment*: the supplier can display aggregates per market segment, i.e., broader application domains (like geomarketing, real estate, transport, environment, government, healthcare, etc.) where geospatial data assets are typically used.



Number of asset views

Keep track of your assets popularity across time and countries.

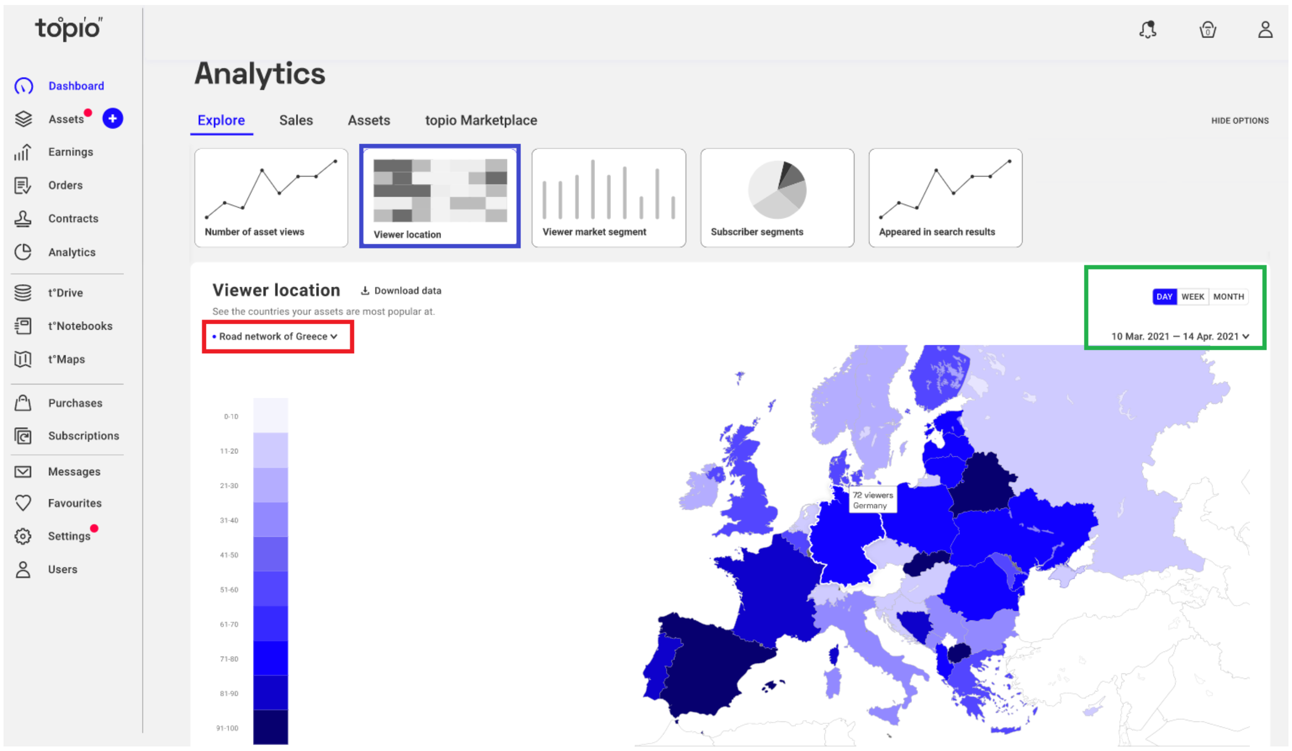
	06:00	07:00	08:00	09:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00	00:00	01:00	02:00	03:00	04:00	05:00
Road network of Greece	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Lakes of Greece	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Rivers of Greece	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Figure 96: Analytics regarding number of asset views

Overall, several *categories* of analytics are available, as shown with the tabs in Figure 96:

- *Explore*: Visitor statistics for *search requests* and *asset views* involving products offered by this supplier are aggregated in order to offer insight about their appeal to potential customers. As illustrated in Figure 96, the supplier can pick up to three of her assets (enclosed in the red box) and compare their popularity amongst visitors of the marketplace over a period of interest (specified in the green box). Such aggregates are plotted as line charts, but they are also available in tabular format and can be downloaded.
- *Sales*: This offers various visualizations on statistics regarding the number of transactions for file assets, number of subscriptions to APIs, actual earnings per asset type, breakdown of sales by customer location or market segment, etc. In addition, the supplier can examine the overall trend of her sales per asset type and by time of interest or country.
- *Assets*: This tab offers to the supplier in-depth insight of trends in popularity and sales of specific assets. Indeed, she can inspect how (up to 10 of) her assets trend among search results returned to visitors in the marketplace, and also examine the number, country, and market segment of visitors who have actually viewed their respective Asset View page (Section **2.4**). For example, she can visualize in a choropleth map the number of viewers per country for a particular asset of interest (Figure 97) or the market segment of viewers for some of her assets (Figure 98). Moreover, she can view charts and heatmaps regarding sales of chosen asset(s) across time.
- *Topio Marketplace*: This tab allows the supplier to assess the overall visibility and appeal of her assets across the marketplace. Of course, she can visually display her top viewed assets, those that most frequently appear among search results, and examine the terms mostly involved in visitors' search requests. She can also inspect her overall performance in the marketplace in terms of: number of visitors to her assets, number of subscribers to her APIs,

transactions, value of file assets, etc. She can also plot heatmaps concerning the geographical coverage of her assets, the location of viewers, and the location of her customers.

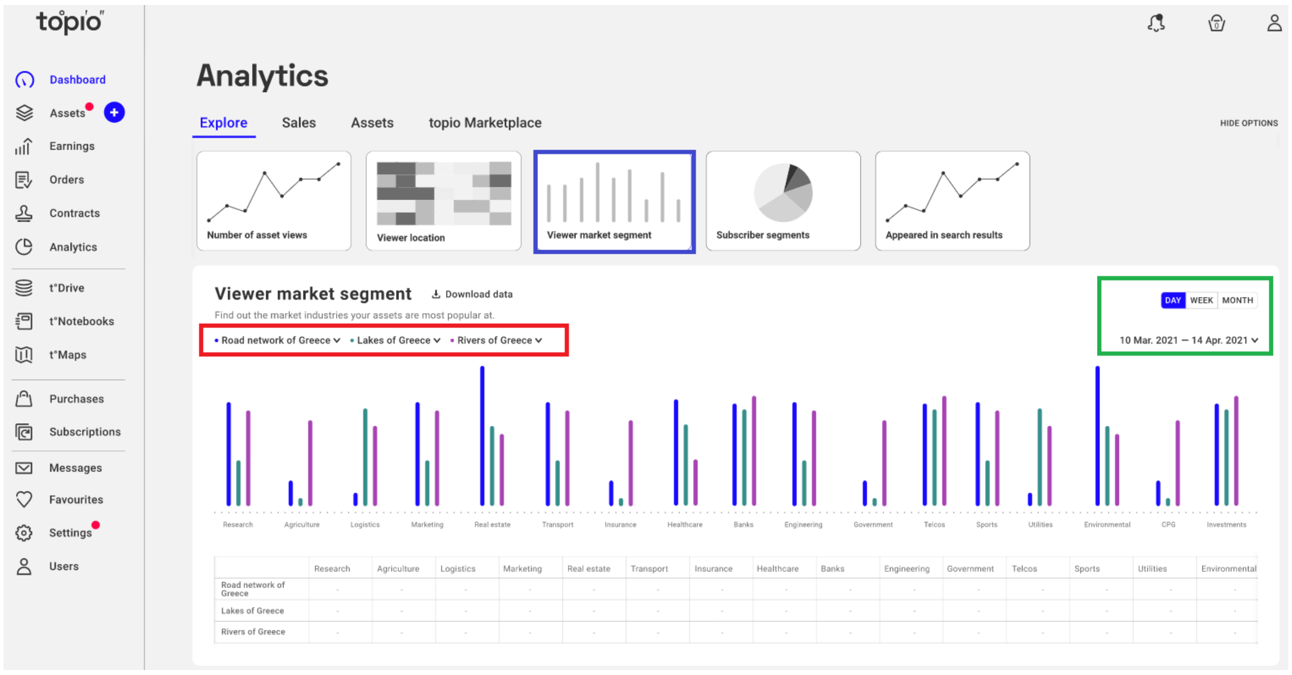


Viewer location

See the countries your assets are most popular at.

• Road network of Greece

Figure 97: Analytics on viewer locations for a specific asset



Viewer market segment

Find out the market industries your assets are most popular at.

• Road network of Greece • Lakes of Greece • Rivers of Greece

	Research	Agriculture	Logistics	Marketing	Real estate	Transport	Insurance	Healthcare	Banks	Engineering	Government	Telcos	Sports	Utilities	Environmental
Road network of Greece	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Lakes of Greece	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Rivers of Greece	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Figure 98: Market segmentation of viewers of specific assets offered by a supplier

2.8.7. Purchases

Since a supplier may himself be also a customer of data assets offered by other suppliers, this option works exactly as in the case of users (see Section 2.7.1).

2.8.8. Subscriptions

Since a supplier may himself be also a customer of API assets offered by other suppliers, this option works exactly as in the case of users (see Section 2.7.2).

2.8.9. Messages

The Messages option offers the same functionality as in the case of users (see Section 2.7.3).

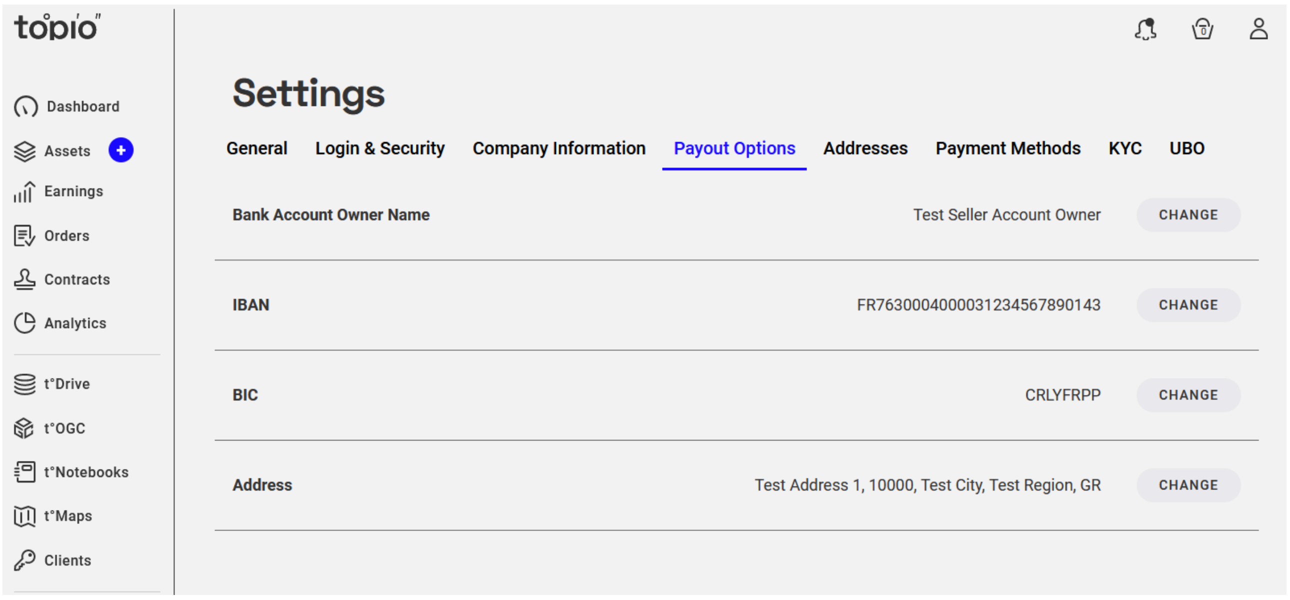
2.8.10. Favorites

This option is the same to that offered to users (see Section 2.7.4).

2.8.11. Settings

The Settings option in the dashboard of an asset supplier mostly supports the same functionality as in the case of registered consumers. However, three extra operations are available to suppliers:

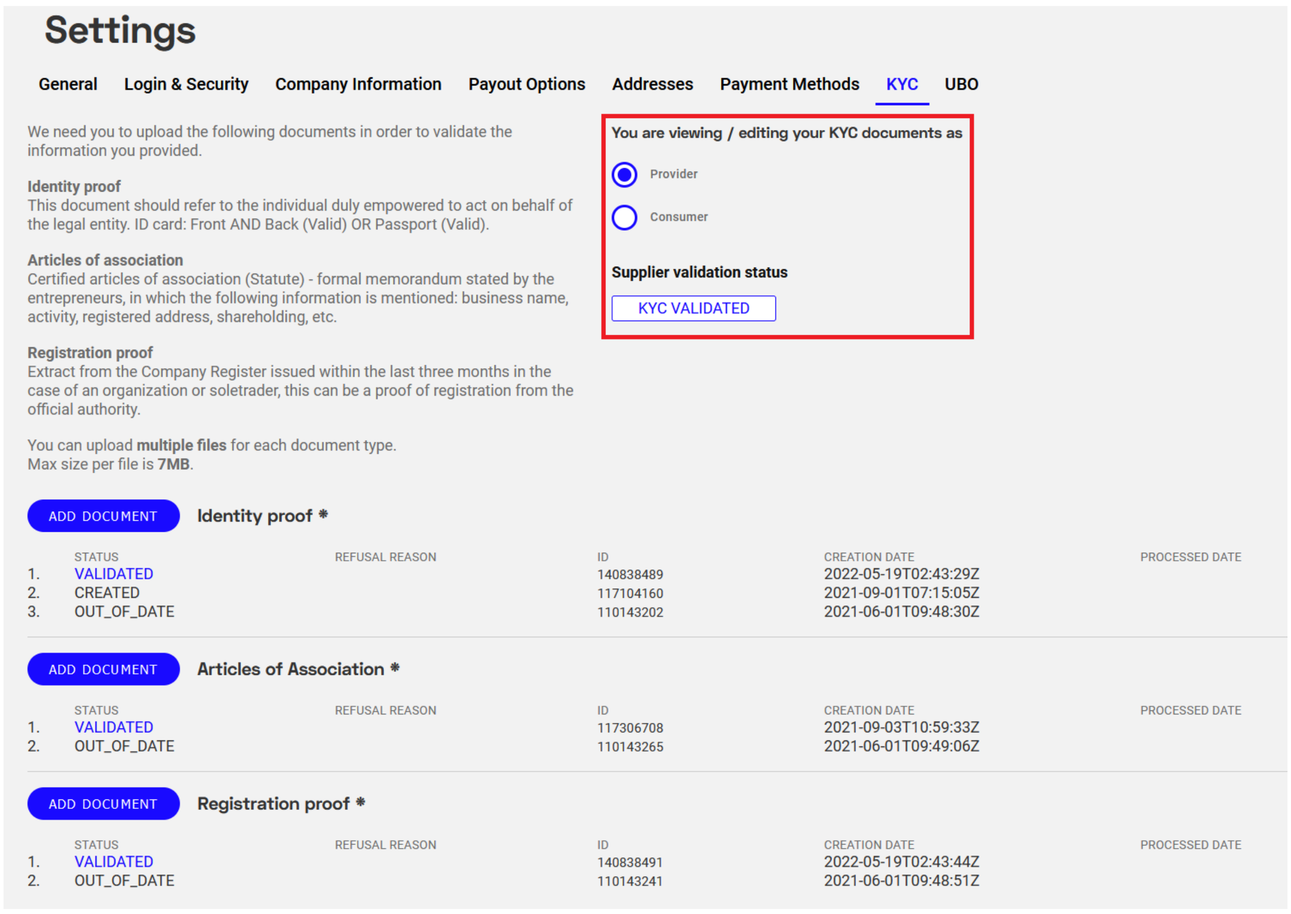
- A panel regarding *payout options* (Figure 99) offers information about the bank account (IBAN) where the supplier will be accepting payments. Through this dashboard page, suppliers can modify their preferred payout method (e.g., remove an account no longer active and set another IBAN account). Note that the contents of this dashboard page reflect the choices the supplier itself made when registered (see Section 2.2.3).



Bank Account Owner Name	Test Seller Account Owner
IBAN	FR7630004000031234567890143
BIC	CRLYFRPP
Address	Test Address 1, 10000, Test City, Test Region, GR

Figure 99: Supplier settings

- *KYC*. Before becoming a supplier in the platform, a user must successfully pass through *Know Your Customer* (KYC) process in order to verify her identity. This can be done either upon registration or before she starts trading her first asset in the platform. If a supplier is not KYC validated yet, a red notification appears on this tab in the settings. As illustrated in Figure 100, the supplier needs to upload several documents in order for the platform to validate that all provided information is correct and that she can perform transactions in the platform. These documents include *Identity proofs* (e.g., ID card or passport), *Articles of association* (i.e., specifying the company status, its registered address, etc.), and *proof of Registration* to an official authority. The supplier can upload each of these documents for validation by the platform; maximum size per document is 7MB. The list shows the status on each required document, which can be one of the following: *upload, validation asked, succeeded*, or *refused*. Note that possibly multiple items per category of documents may need be submitted, e.g., if previously uploaded document has been refused. Unless all required documents have been successfully validated, the supplier validation status remains "*Not KYC validated*" and the user is notified about the missing or additionally required documents. If all required documents are submitted and found valid, the status changes to "*KYC validated*" as indicated within the red box.



We need you to upload the following documents in order to validate the information you provided.

Identity proof
This document should refer to the individual duly empowered to act on behalf of the legal entity. ID card: Front AND Back (Valid) OR Passport (Valid).

Articles of association
Certified articles of association (Statute) - formal memorandum stated by the entrepreneurs, in which the following information is mentioned: business name, activity, registered address, shareholding, etc.

Registration proof
Extract from the Company Register issued within the last three months in the case of an organization or soletrader, this can be a proof of registration from the official authority.

You can upload **multiple files** for each document type.
Max size per file is **7MB**.

You are viewing / editing your KYC documents as

[✓ Provider] Consumer

Supplier validation status

KYC VALIDATED

Identity proof *

	STATUS	REFUSAL REASON	ID	CREATION DATE	PROCESSED DATE
1.	VALIDATED		140838489	2022-05-19T02:43:29Z	
2.	CREATED		117104160	2021-09-01T07:15:05Z	
3.	OUT_OF_DATE		110143202	2021-06-01T09:48:30Z	

Articles of Association *

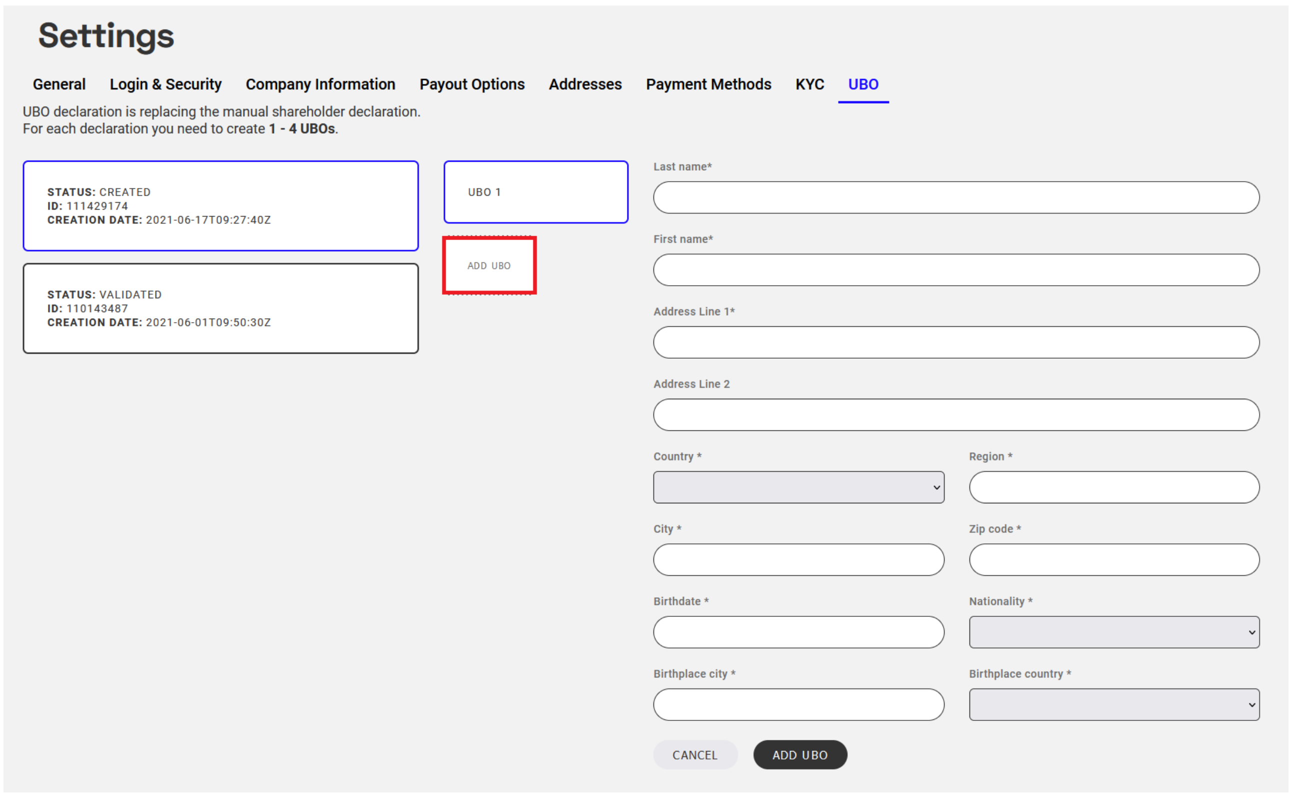
	STATUS	REFUSAL REASON	ID	CREATION DATE	PROCESSED DATE
1.	VALIDATED		117306708	2021-09-03T10:59:33Z	
2.	OUT_OF_DATE		110143265	2021-06-01T09:49:06Z	

Registration proof *

	STATUS	REFUSAL REASON	ID	CREATION DATE	PROCESSED DATE
1.	VALIDATED		140838491	2022-05-19T02:43:44Z	
2.	OUT_OF_DATE		110143241	2021-06-01T09:48:51Z	

Figure 100: KYC settings

- *UBO*. To avoid money laundering and terrorist financing, the platform requires information about the *Ultimate Beneficial Owner* (UBO) of the supplier company. UBO is an individual person who may not be recorded as a shareholder, but who ultimately owns the company, has actual power and authority to direct the company, and get profits. In the marketplace, UBO declaration replaces the manual shareholder declaration. To create such a declaration, the supplier must press the button "*Add declaration*" and then specify the required details for *at least one and up to four UBOs*, as illustrated in Figure 101. If there is an issue with the details submitted for any of the UBOs, the status of the declaration is marked as "*Incomplete*" and the specific UBO is shown in red. The supplier can then open the entry form for this UBO and edit the missing or incorrect values. If the status indication changes to "*Validated*", the process is successful; otherwise, it is marked as incomplete with the red notification in the *Settings*.


Settings

General | Login & Security | Company Information | Payout Options | Addresses | Payment Methods | KYC | UBO

UBO declaration is replacing the manual shareholder declaration.
For each declaration you need to create **1 - 4 UBOs**.

STATUS: CREATED
ID: 111429174
CREATION DATE: 2021-06-17T09:27:40Z

STATUS: VALIDATED
ID: 110143487
CREATION DATE: 2021-06-01T09:50:30Z

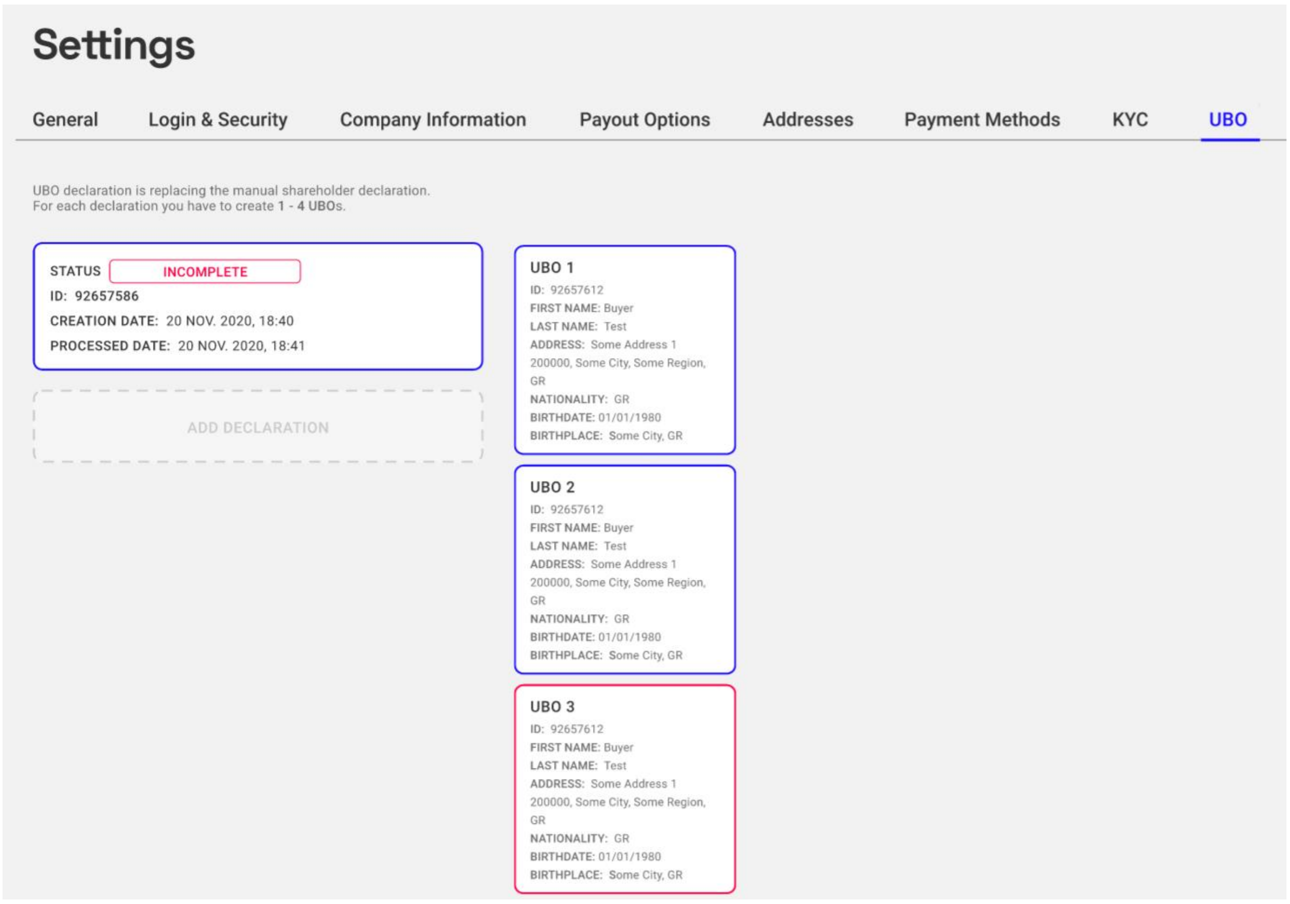
UBO 1

ADD UBO

Last name*
First name*
Address Line 1*
Address Line 2
Country *
Region *
City *
Zip code *
Birthdate *
Nationality *
Birthplace city *
Birthplace country *

CANCEL | ADD UBO

Figure 101: Adding UBO information


Settings

General | Login & Security | Company Information | Payout Options | Addresses | Payment Methods | KYC | UBO

UBO declaration is replacing the manual shareholder declaration.
For each declaration you have to create 1 - 4 UBOs.

STATUS INCOMPLETE
ID: 92657586
CREATION DATE: 20 NOV. 2020, 18:40
PROCESSED DATE: 20 NOV. 2020, 18:41

ADD DECLARATION

UBO 1
ID: 92657612
FIRST NAME: Buyer
LAST NAME: Test
ADDRESS: Some Address 1
200000, Some City, Some Region, GR
NATIONALITY: GR
BIRTHDATE: 01/01/1980
BIRTHPLACE: Some City, GR

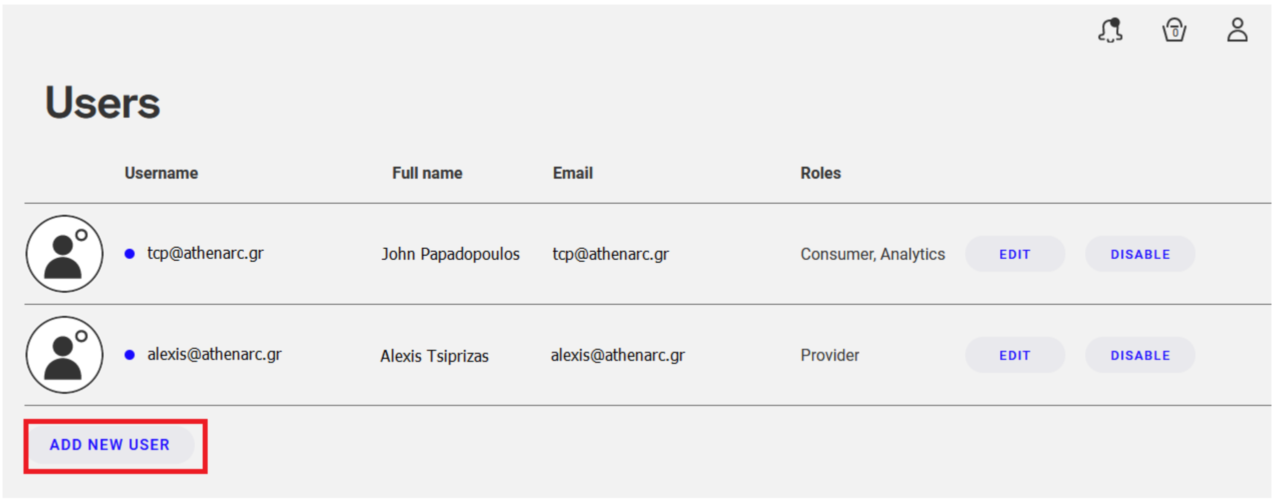
UBO 2
ID: 92657612
FIRST NAME: Buyer
LAST NAME: Test
ADDRESS: Some Address 1
200000, Some City, Some Region, GR
NATIONALITY: GR
BIRTHDATE: 01/01/1980
BIRTHPLACE: Some City, GR

UBO 3
ID: 92657612
FIRST NAME: Buyer
LAST NAME: Test
ADDRESS: Some Address 1
200000, Some City, Some Region, GR
NATIONALITY: GR
BIRTHDATE: 01/01/1980
BIRTHPLACE: Some City, GR

Figure 102: Status of UBO information

2.8.12. Users

Thanks to this option, a supplier can act as administrator of all the user accounts associated with her organization and manage their roles in the marketplace (*consumer*, *provider*, *analytics*) regarding her assets. Note that a user can be assigned multiple roles. As shown in Figure 103, the supplier can create new user accounts and associate them with her organization.


Users

Username	Full name	Email	Roles		
tcp@athenarc.gr	John Papadopoulos	tcp@athenarc.gr	Consumer, Analytics	EDIT	DISABLE
alexis@athenarc.gr	Alexis Tsiprizas	alexis@athenarc.gr	Provider	EDIT	DISABLE

ADD NEW USER

Figure 103: List of user accounts per organization

3. Annex

3.1. Workflows

In this Annex, we present the BPMN diagrams for the workflows implemented and deployed in the OpertusMundi platform. All workflows and their illustrations are available at our project repository[62]. As elaborated in Section 1.1, workflows are executed using the Camunda BPM platform and implement microservice orchestration. More broadly in the OpertusMundi platform, all tasks involving complex microservice interactions, user input and actions based on temporal events, are implemented as workflows.

3.1.1. Publish Asset

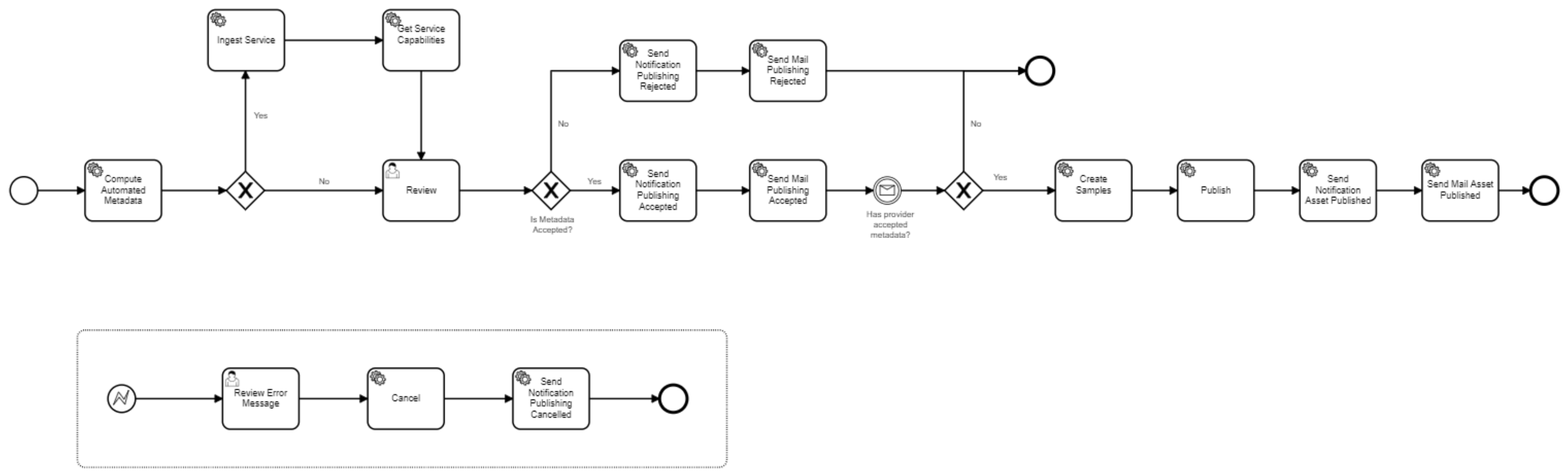
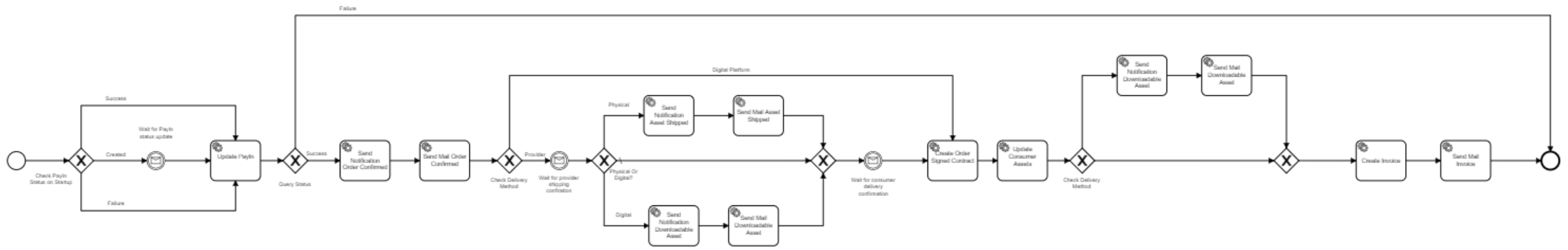

Ingest Service
Get Service Capabilities
Yes
Compute Automated Metadata
No
Review
Send Notification Publishing Rejected
Send Mail Publishing Rejected
No
Yes
Is Metadata Accepted?
Send Notification Publishing Accepted
Send Mail Publishing Accepted
Has provider accepted metadata?
No
Yes
Create Samples
Publish
Send Notification Asset Published
Send Mail Asset Published
Review Error Message
Cancel
Send Notification Publishing Cancelled

Figure 104: Publish asset workflow

[62] https://github.com/OpertusMundi/workflows

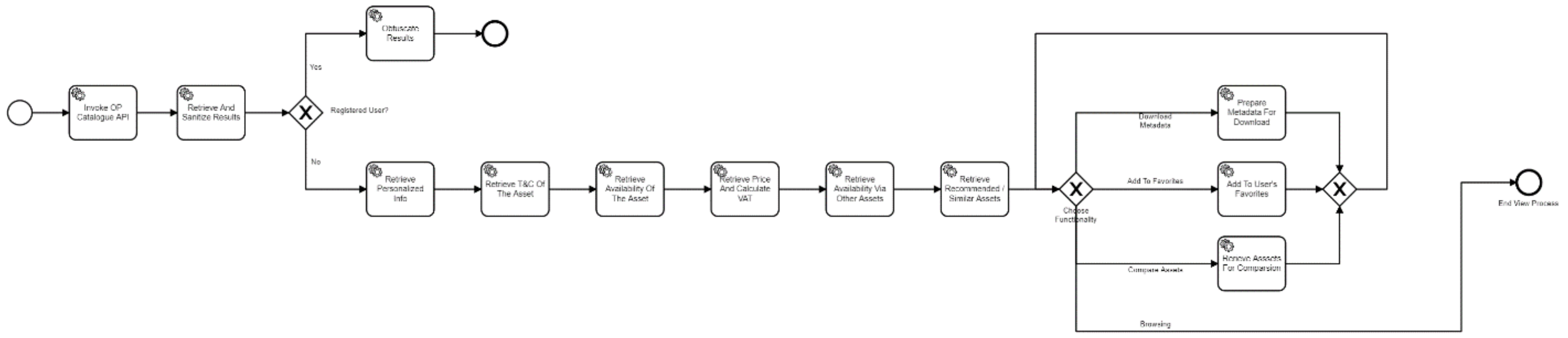
3.1.2. Purchase Asset with PayIn



Failure
Digital Platform
Success
Wait for PayIn status update
Created
Failure
Check PayIn Status on Startup
Update PayIn
Success
Query Status
Send Notification Order Confirmed
Send Mail Order Confirmed
Check Delivery Method
Provider
Wait for provider shipping confirmation
Physical
Physical Or Digital?
Digital
Send Notification Asset Shipped
Send Mail Asset Shipped
Send Notification Downloadable Asset
Send Mail Downloadable Asset
Wait for consumer delivery confirmation
Create Order Signed Contract
Update Consumer Assets
Check Delivery Method
Send Notification Downloadable Asset
Send Mail Downloadable Asset
Create Invoice
Send Mail Invoice

Figure 105: Purchase asset with PayIn

3.1.3. Purchase Asset without PayIn



Invoke OP Catalogue API
Retrieve And Sanitize Results
Registered User?
Yes
No
Obfuscate Results
Retrieve Personalized Info
Retrieve T&C Of The Asset
Retrieve Availability Of The Asset
Retrieve Price And Calculate VAT
Retrieve Availability Via Other Assets
Retrieve Recommended / Similar Assets
Choose Functionality
Download Metadata
Add To Favorites
Compare Assets
Browsing
Prepare Metadata For Download
Add To User's Favorites
Remove Assets For Comparison
End View Process

Figure 106: Purchase asset without PayIn workflow

3.1.4. Activate Supplier Account

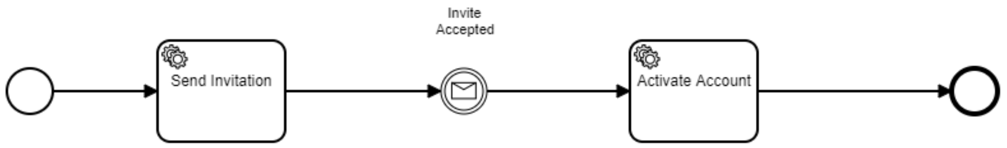


Figure 107: Activate supplier account workflow

3.1.5. Consumer Subscription Billing Payoff

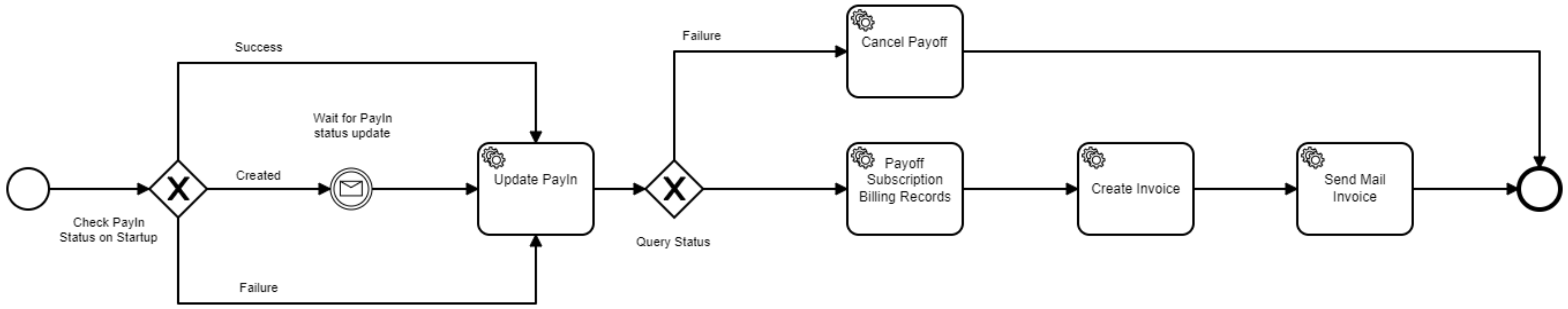


Figure 108: Consumer subscription billing payoff workflow

3.1.6. Sign Up Consumer

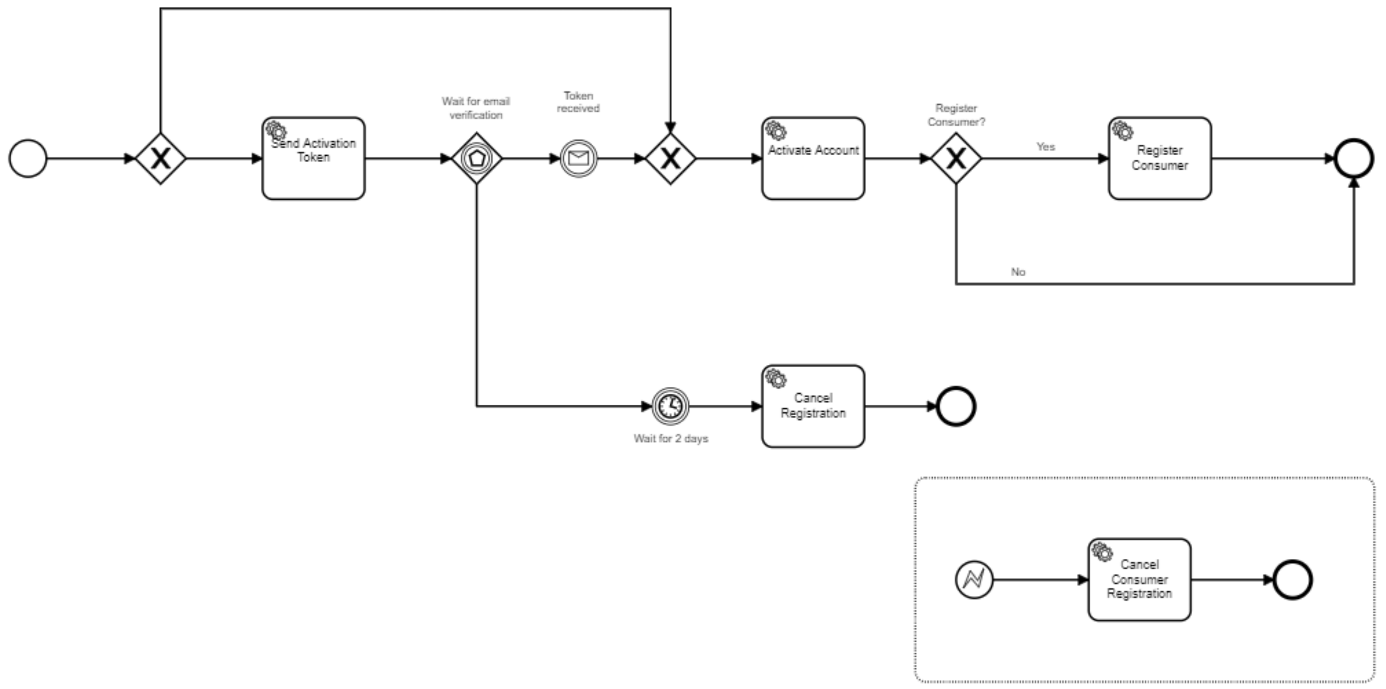


Figure 109: Sign up consumer workflow

3.1.7. Copy Resource to Drive

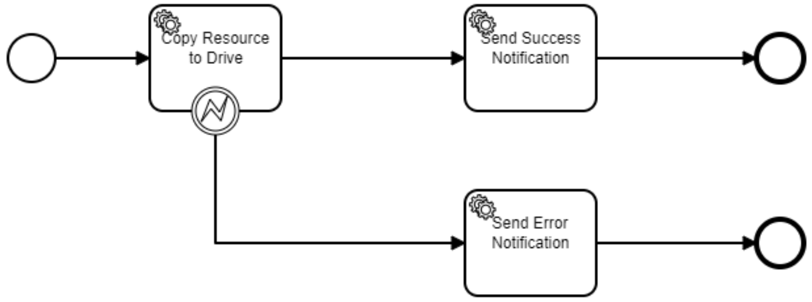


Figure 110: Copy resource to drive workflow

3.1.8. Harvest Catalogue

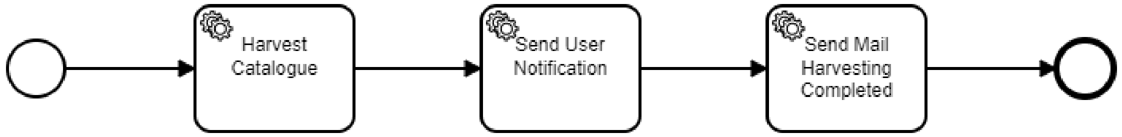


Figure 111: Harvest catalogue workflow

3.1.9. Provider Payout

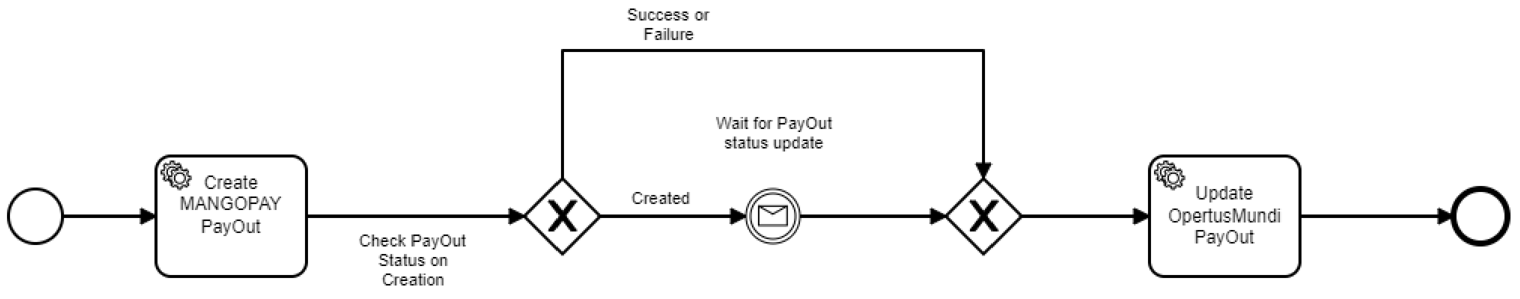

Success or Failure
Create MANGOPAY PayOut
Check PayOut Status on Creation
Created
Wait for PayOut status update
Update OpertusMundi PayOut

Figure 112: Provider payout workflow

3.1.10. Provider Send KYC Notification

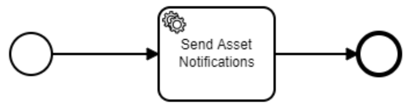


Figure 113: Provider send KYC notification workflow

3.1.11. Provider Update Default Contracts

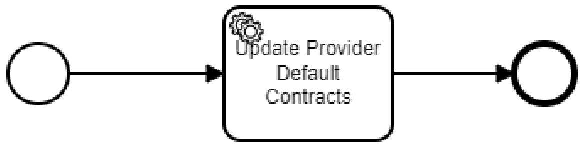


Figure 114: Provider update default contracts workflow

3.1.12. Provider Upload Contract

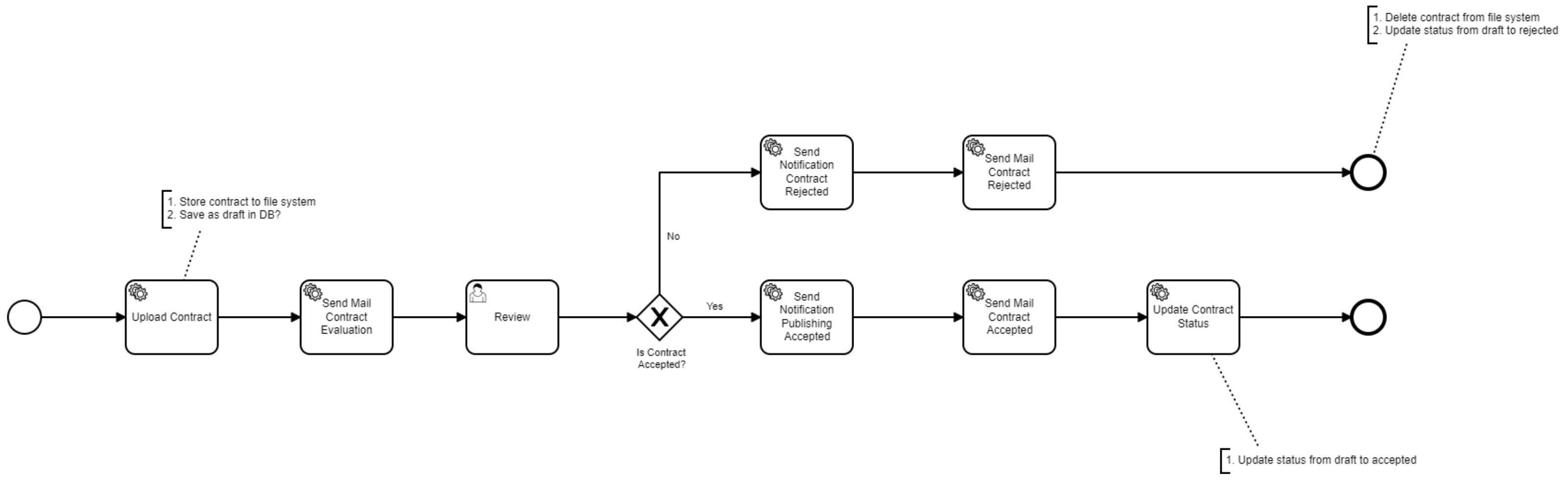

1. Delete contract from file system
2. Update status from draft to rejected
1. Store contract to file system
2. Save as draft in DB?
Upload Contract
Send Mail Contract Evaluation
Review
Is Contract Accepted?
No
Yes
Send Notification Contract Rejected
Send Mail Contract Rejected
Send Notification Publishing Accepted
Send Mail Contract Accepted
Update Contract Status
1. Update status from draft to accepted

Figure 115: Provider upload contract workflow

3.1.13. Publisher Remove Asset

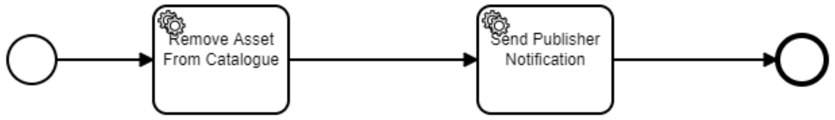


Figure 116: Publisher remove asset workflow

3.1.14. Register Consumer

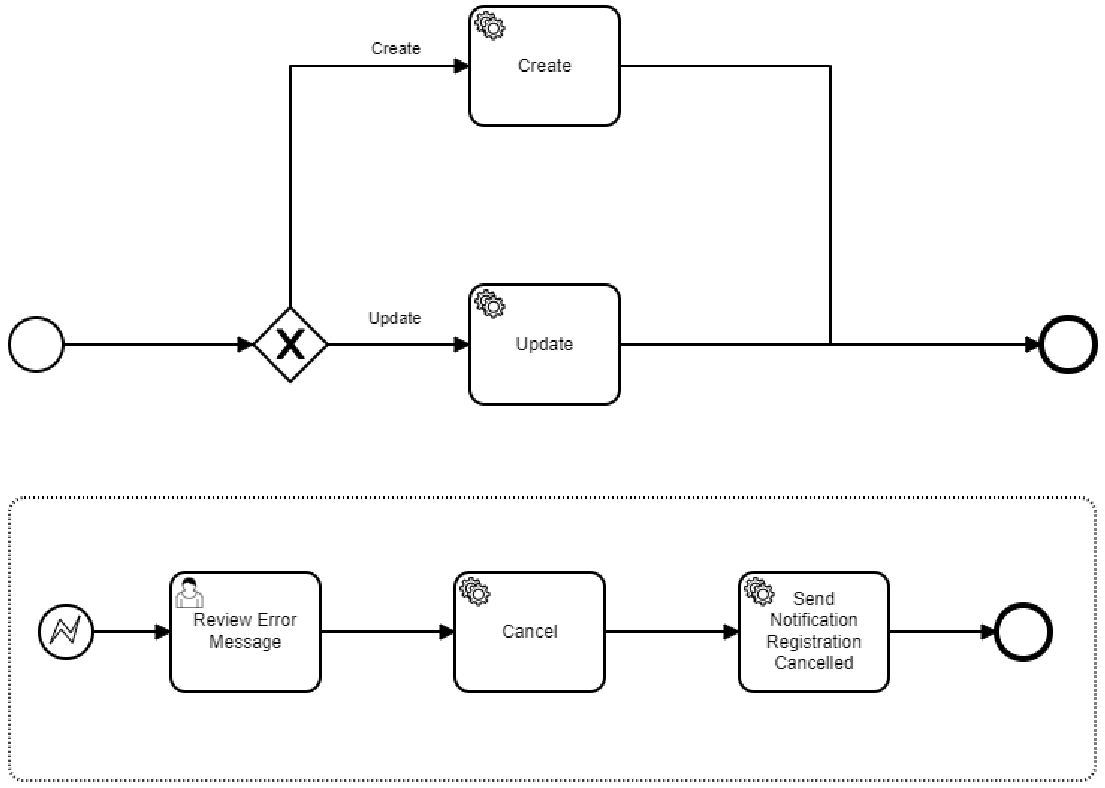


Figure 117: Register consumer workflow

3.1.15. Register Provider

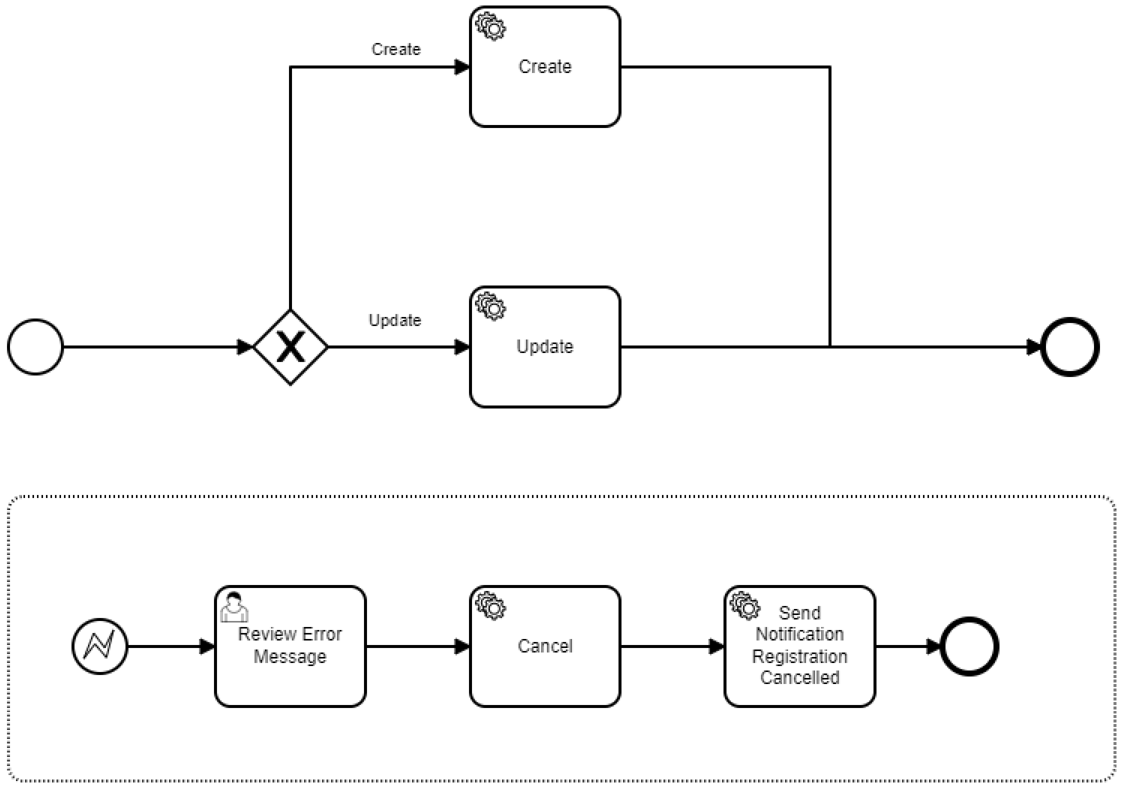


Figure 118: Register provider workflow

3.1.16. Subscription Billing

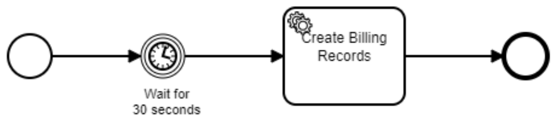


Figure 119: Subscription billing workflow

3.1.17. System Maintenance Delete All User Data

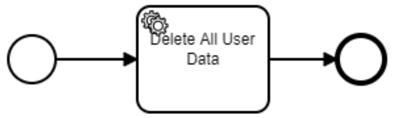


Figure 120: System maintenance delete all user data workflow

3.1.18. System Maintenance

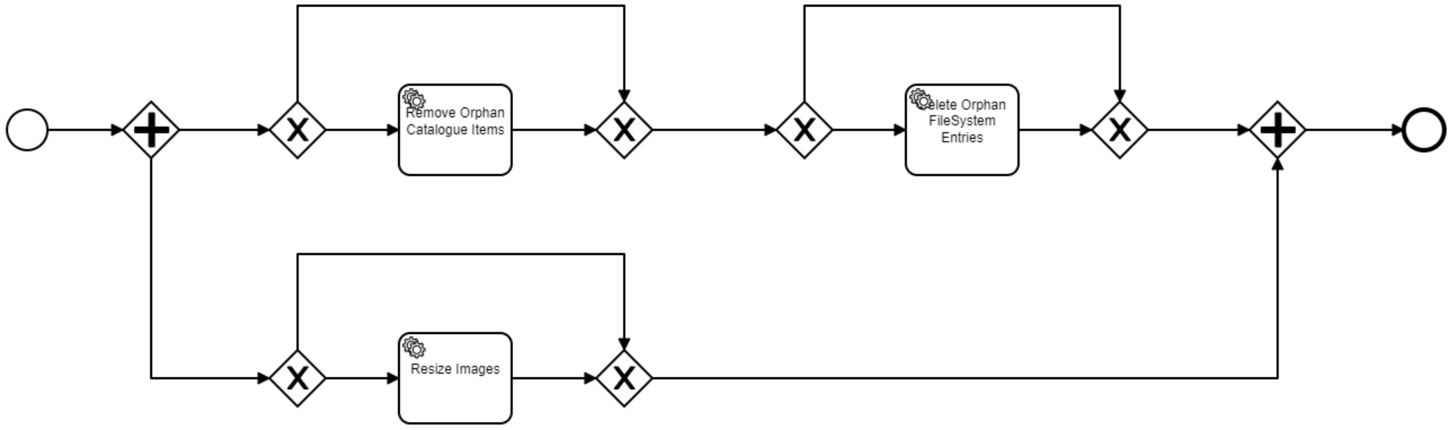

Remove Orphan Catalogue Items
Delete Orphan FileSystem Entries
Resize Images

Figure 121: System maintenance workflow

3.1.19. User Publish Service

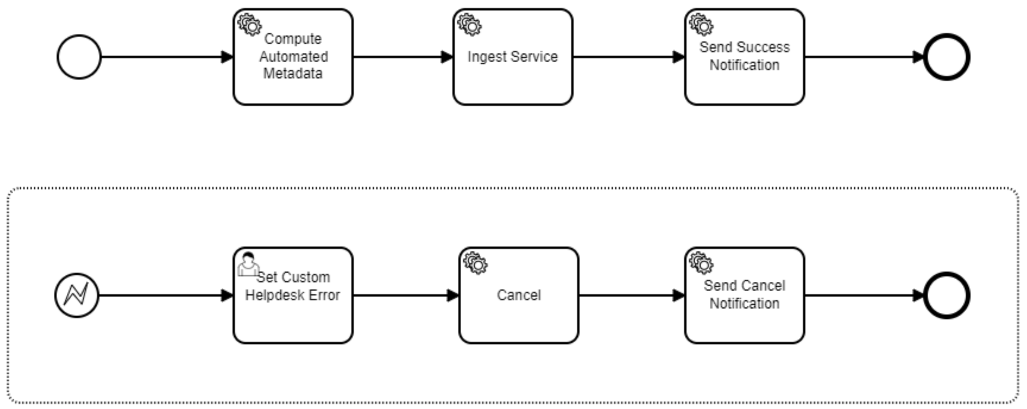

Compute Automated Metadata
Ingest Service
Send Success Notification
Set Custom Helpdesk Error
Cancel
Send Cancel Notification

Figure 122: User publish service workflow

3.1.20. User Remove Service

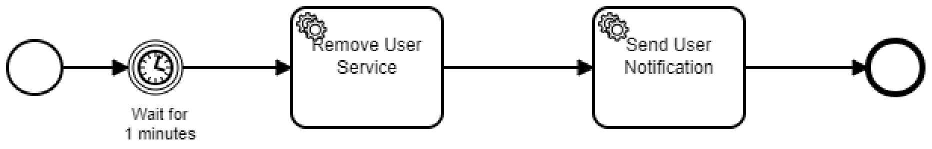


Figure 123: User remove service workflow

3.2. Software Versions

In this section, we enumerate the current versions for all major software components used for the system's deployment.

Ubuntu:

- Distributor ID: Ubuntu
- Description: Ubuntu 16.04.4 LTS
- Release: 16.04
- Codename: xenial

KeepAlived: 1:1.2.24

Docker Client (Community): 19.03.13

Docker API: 1.40

Docker Server Engine (Community): 19.03.13

Docker Registry: 2.7

Kubernetes Client: 1.22.2

Kubernetes Server: 1.22.2

Helm: 3.9.4

Nginx: 1.19.9

Nginx-Ingress: 1.0.4

Drone CI Server/Runner: 1.10

PostgreSQL: 10.15

PostGIS: POSTGIS="3.0.3 0" [EXTENSION] PGSQL="100" GEOS="3.8.1-CAPI-1.13.3" PROJ="7.0.1" LIBXML="2.9.10" LIBJSON="0.14" LIBPROTOBUF="1.3.3" WAGYU="0.4.3 (Internal)" TOPOLOGY

PgPool: 4.2

Node.JS: 10.16

Java: OpenJDK 17.0.5 Temurin

Maven: 3.6.3

Spring Boot: 2.7.3

Quarkus: 2.7.5

Python: 3.8

GDAL library: 3.0.2

Rsyslog: 8.2012.0