OneAquaHealth

PROTECTING URBAN AQUATIC ECOSYSTEMS TO PROMOTE ONE HEALTH

Grant Agreement: 101086521

D5.3 Digital app and backoffice

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Research Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Document Information

Deliverable number:	**D5.3**
Deliverable title:	Digital app and backoffice
Deliverable version:	Draft version
Work Package number:	WP5
Work Package title:	AI supported Environmental surveillance system
Due date of delivery:	M18 (June 2024)
Actual date of delivery:	20.6.2024
Dissemination level:	Public (PU)
Type:	Demonstrator (DEM)
Author(s):	George Koutalieris, Paraskevi Rapti, Symeon Symeonidis, Giorgos Stamatelatos, Vassilis Perifanis (ENORA)
Contributor(s):	Maria João Feio, Sónia Serra, Ana Raquel Calapez, Janine P Silva (UC)
Reviewer(s):	Anne Moen, UiO
Project name:	Protecting Urban Aquatic Ecosystems to Promote One Health
Project Acronym:	OneAquaHealth
Project starting date:	01.01.2023
Project duration:	48 months
Rights:	OneAquaHealth Consortium

Document history

Version	Date	Beneficiary	Description
0.1	05.02.2024	ENORA	Set up of the document and Creation of Table of Contents
0.2	07.03.2024	ENORA/UC	Description of the Citizen Science App specs
0.4	08.04.2024	ENORA	Update of architecture and technical specifications
0.5	03.05.2024	ENORA	Documentation of the development and presentation of the software user interfaces
0.6	22.05.2024	UC	Comments and contributions in the Citizen Science section
0.7	02.06.2024	ENORA/SHINE	Alignment with citizen science design specs in WP4
0.8/0.9	12.06.2024	ENORA/UiO	Final version for peer review/revisions based on peer review comments
1.0	20.06.2024	ENORA/UC	Final version for submission

Executive Summary

D5.3 (Digital app and backoffice) is a demonstration deliverable that encapsulates the software development and documents the in-situ data backoffice and the Citizen Science app development. It is based on the developments of data definition and data collection methods for ecosystem health indicators at the OneAquaHealth research sites, as well as the requirements analysis and functional specifications developed by citizens and local alliances. In addition to documenting technological developments, this deliverable also provides a coherent software design framework that leads this technological development and the upcoming improvements planned in the new releases. The software development work will continue to evolve until M24 to capture end-user feedback and continuously improve during the integration in the OneAquaHealth Information Hub. All the software development updates of Task 5.2 will be reported to the D5.2 (Geospatial and satellite information platform vs2). The software development process is driven by a SCRUM software development methodology that links the user requirements (user stories) with the software components and applications.

Contents

Executive Summary....4
1 Introduction....8
1.1 Relation to other tasks and deliverables....8
1.2 Structure of the deliverable....9
2 Citizen Science Application....9
2.1.1 Analysis of requirements and needs....9
2.1.1.1. Key features....10
2.1.1.2. Backend implementation....12
2.1.1.3. Software components design....12
2.1.2 User walkthrough of the OAH Citizen Science App....15
2.1.3 Installation of the OAH Citizen Science App....23
3 Conceptual Architecture....28
3.1 Design and Architecture....28
3.1.1 Data Collection Layer....28
3.1.2 Data Processing and Analytics Engine Layer....28
3.1.3 OAH Visualization Layer....29
3.1.4 Interactive Endpoints....29
3.1.5 User Applications....29
3.1.6 Security and Support Layers....29
3.2 Data Management....29
3.3 Authentication and Authorisation....29
3.4 API....30
4 Data Backoffice Development....33
4.1 Architecture Design: Overview of the Data Backoffice Architecture....33
4.2 GEOSSIP: The OAH software platform....35
5 Software development methodology: SCRUM methods and tools....46
6 Next Steps....48
7 Conclusion....49
References....50

Figures

Figure 1 Citizen Science technology stack 13
Figure 2 Citizen Science - Submission Guide 16
Figure 3 Request for location 17
Figure 4 Citizen Science - Site selection 17
Figure 5 Citizen Science - Image and video upload 18
Figure 6 Citizen Science App - First question set 19
Figure 7 Citizen Science App - Second question set 19
Figure 8 Citizen Science App - Third question set 20
Figure 9 Citizen Science App - Overall assessment 20
Figure 10 Citizen Science App - Submission 21
Figure 11 Citizen Science App - List of Submissions 22
Figure 12 Citizen Science App - Submission Preview 22
Figure 13 Visit the Citizen Science app 23
Figure 14 Installation of the app 24
Figure 15 Acceptance for installation of Citizen Science App 25
Figure 16 Installation message 26
Figure 17OAH Citizen Science App installed on the mobile apps 27
Figure 18 OAH High-level architecture of applications 28
Figure 19 EO APIs 31
Figure 20 GEOSSIP APIs 32
Figure 21 The software architecture of the OneAquaHealth backend 33
Figure 22 GEOSSIP Home Page 36
Figure 23 Login Screen 37
Figure 24 Research Sites Map 38
Figure 25 List of Research sites and coordinates 39
Figure 26 Sample Expert Dashboard for Nutrients 40
Figure 27 Categories of Data Annotations types 41
Figure 28 Preview of Data Submissions 42
Figure 29 Data Submission Form 43
Figure 30 GEOSSIP - Satellite Imagery 44
Figure 31 GEOSSIP - EO Indicators 45
Figure 32 Scrum methodology in Jira 47
Figure 33 Jira tickets for the implementation 47

Acronyms & Abbreviations

Term	Description
API	Application Programming Interface
EO	Earth Observation
OAH	OneAquaHealth

1 Introduction

This deliverable, D5.3 *Digital app and backoffice*, aims to document the results of activities conducted in WP5 up to M18 (June 2024) of the OneAquaHealth. Specifically, it details the work completed in the following task:

- *Task 5.2 Data Backoffice and Citizen Science app* - This task involves designing and developing the Data Backoffice, which aggregates various data feeds from WP2, WP3 and WP4 regarding the OAH five research sites (Coimbra, Toulouse, Ghent, Benevento, and Oslo). Additionally, a Citizen Science app was developed based on user preferences and enabled user engagement in T4.3 by allowing user-generated content to be fed into the Data Backoffice.

The rest of the activities in WP5 have been carried out. They are related to *Task 5.1 Geospatial and Satellite Information Platform vs1*, which focuses on developing a web-based Geospatial and Satellite Information Hub (GEOSSIP) that provides access to in-situ data produced during WP2 and WP3, as well as earth observation data. These activities are documented in *D5.1 Geospatial and Satellite Information Platform vs1*, submitted concurrently with this document in June 2024 (M18).

The design of the Citizen Science App is based on the requirements analysis, personas and functional specifications outlined in deliverable D4.1, "Citizen science design, process and outcomes vs1" [2].

1.1 Relation to other tasks and deliverables

The relationships between this deliverable and other deliverables and tasks are outlined in Table 1 (inputs from other tasks/deliverables) and Table 2 (outputs to other tasks/deliverables) below.

It receives inputs from:

Table 1. D5.3 Input from other tasks and deliverables

Deliverable	Due Date	Input for D5.3
D2.1	M18	This deliverable specifies the Key ecosystem health and biological health indicators.
D3.1	M18	This deliverable links health indicators that may be associated with the ecological condition of urban streams. This work, however, is not yet fully developed because it involves extensive desk research on the impact of urban aquatic resources on the overall community and public health indicators.
D4.1	M18	The design of the Citizen Science App is based on the requirements analysis, personas and functional specifications outlined in deliverable D4.1, "Citizen science design, process and outcomes vs1".
D7.7	M18	The Data Management Plan has been updated based on the data model utilised in the OAH Backend.

It provides outputs to:

Table 2. D5.3 Output for other tasks and deliverables

Deliverable	Due Date	Output from D5.3
T2.4	M26-M38	To be considered in the upcoming iterations of this research documentation.
WP3		To be considered in the upcoming iterations of this research documentation.
WP4		To be considered in the upcoming iterations of this research documentation.
T5.3	M24-42	To be considered in the upcoming development of the AI models.
T6.2	M24-42	To be considered in the upcoming development of the OAH Information Hub Services.

1.2 Structure of the deliverable

The rest of this document is organised as follows. It is divided into the following sections: Section 2 explains the Citizen Science App and its key features. Section 3 presents the Conceptual architecture and design of the OneAquaHealth, while the following section 4 presents the Data Backoffice architecture and development. Finally, section 5 presents the software development methodology for the development, and sections 6 and 7 show the following steps and conclusions of the applications.

2 Citizen Science Application

The OAH Citizen Science App is a Progressive Web Application (PWA) designed to engage with the public in monitoring and assessing urban stream ecosystems. As an integral part of the OAH Information Hub, this app aims to raise awareness and educate the community regarding the health of urban streams. It is primarily intended for citizen scientists, environmental enthusiasts, researchers and educators.

The OAH Citizen Science App empowers individuals to participate actively in environmental monitoring, providing valuable data to help protect and restore urban stream ecosystems. Through this app, the OAH Information Hub seeks to build a knowledgeable and engaged community dedicated to preserving freshwater ecosystems, especially those located inside cities.

2.1.1 Analysis of requirements and needs

The OneAquaHealth app empowers citizens to participate in the monitoring and reporting of urban stream health. Through the app, users contribute to a growing database of environmental data, promoting sustainable practices and community involvement in ecological preservation.

Informative Section

- The app features an educational section detailing what constitutes a stream ecosystem (e.g., channels, banks, habitats, riparian vegetation) and what a healthy vs degraded ecosystem is. It includes an explanation of the scoring system used for assessing stream health, supplemented by visual aids and comparisons similar to those found in the Mosquito Alert app.

Scoring System

- A scoring system ranging from 1 to 3 quantifies stream degradation:
    - 1 Good quality (The ecosystem components are there: riparian vegetation, natural channel, good water quality, biodiversity).
    - 2 Moderate quality (Some alterations, still biodiverse, with vegetation in the margins, water looks good).
    - 3 Poor quality (Highly modified/artificialized, loss of riparian vegetation, loss of habitats, polluted).
- Detailed criteria for scoring areprovided for each assessment category.

Mapping and Data Visualisation

- Interactive maps will display monitored streams, highlighting their health classification and providing georeferenced data.
- Users can access their own data contributions, following the model established by the Mosquito Alert app.

Community Feed

- A live feed via Open Information Hub will showcase user contributions, displaying data from nearby streams (defined by a geographic buffer zone).

Registration and Use

- The app features a straightforward registration process, requiring a username, email, and screen name, following the standard procedure for similar applications. Anonymous entries are not considered at this stage of the application. However, we are considering this option especially in the case of privacy-preserving scenarios and when complying with regulations like the General Data Protection Regulation (GDPR) is necessary.
- An intuitive interface with large icons and clear, simple language caters to a diverse user base, including non-native language speakers.

Security

- Robust security measures are implemented to ensure user data protection and privacy.

Content Creation and Sharing

- Users are able to create and share georeferenced photos, encouraging outdoor exploration and environmental engagement via the Open Information Hub.

Mapping and Community Interaction

- A map visualisation of shared content will enhance learning about urban streams.
- Users can comment on shared posts, promoting community interaction via Open Information Hub.

2.1.1.1. Key features

1. User Authentication:

- Users can log in using their credentials (username and password) to access personalised features based on their role (general user, scientist, or administrator).

2. Introductory Guide:

- New users are presented with a beginner's guide that can be skipped. This guide explains the project's and app's aims, introduces important concepts about stream ecosystems and emphasises the significance of urban streams and their health. The user can return to the beginner guide at any time using the designated icon (see screenshots in Section 2.1.2).

3. Photo and Geotagging:

- Users can take photos and videos of streams and their features and upload them to the app. The app uses modern PWA features, such as the geolocation API, to map the user's exact location and the photos taken from their device. Users can adjust the location on the map before submitting.

4. Stream Assessment Guide:

- Detailed instructions guide users through the process of assessing the stream. Users are encouraged to take up to four photos from different perspectives and make a short optional video. This guide can also be skipped.

5. Stream Assessment Questionnaire:

- Users fill out a questionnaire about the stream's condition, including details about the channel, margins/riparian zone, and overall ecosystem health. Each section of the questionnaire can be skipped if desired.

6. Data Submission and Review:

- Users can submit their photos, videos and questionnaire responses. They can also view a list of submissions from other users for the same location, fostering a sense of community and collaboration.

7. User Interface and Experience:

- The app features an intuitive and engaging user interface designed for ease of use. It is mobile-responsive and accessible, ensuring that a broad user base can effectively engage with the app.

8. Backend Architecture:

- The app leverages PWA technology for a seamless user experience across devices. It integrates with the OAH Information Hub to ensure efficient data collection, storage and management.

9. Data Management and Privacy:

- In this version, the app adheres to common data privacy policies in order to ensure that user information is appropriately protected under a common privacy-preserving framework shared with the OAH Open Information Hub(OIH). Anonymous data contributions will also be considered in agreement with the ongoing OIH framework.

10. Testing and Quality Assurance:

- A comprehensive testing strategy includes unit tests, integration tests and user acceptance testing to ensure the app functions reliably and meets user expectations.

11. Deployment and Maintenance:

- Detailed deployment steps and maintenance plans are outlined to ensure the app remains up-to-date and bugs-free. Regular updates and user feedback are incorporated to improve the app continuously.

12. Future Enhancements:

- The app is designed with future enhancements in mind, allowing for new features and improvements based on user feedback and technological advancements.

2.1.1.2. Backend implementation

The backend architecture of the OAH Citizen Science App is designed to ensure scalability, security and seamless integration with the OAH Information Hub. The architecture leverages Progressive Web App (PWA) technology, providing users with a responsive and reliable experience across devices and platforms. Below is a detailed explanation of the backend components and their interactions.

1. Server-Side Components:

- Application Server: Hosts the backend application logic, persistence system, and user management. It is implemented using Spring Boot and the Hibernate ORM framework.

- Database: Stores user data, stream assessments, and media. Uses a relational database like PostgreSQL for structured data.

- Authentication Service: Manages user authentication and authorisation. The service is implemented using JWT-based industry standards.

- File Storage: Stores uploaded media files (photos and videos). Typically uses cloud storage solutions like AWS S3 or Google Cloud Storage.

- Geolocation Service: Processes and converts geotagged data into accurate map photos and assessments. It utilises APIs like Google Maps, OpenStreetMap and Leaflet.

- API Gateway: Manages API traffic and routes requests to appropriate services. It ensures secure communication between the front end and back end.

2. Integration with OAH Information Hub:

- Data Sync Service: Synchronizes data between the app and the OAH Information Hub. Ensures that data collected by the app is available on the broader platform, maintaining consistency and accessibility.

- Notification Service: Sends notifications to users regarding updates, reminders and alerts related to their stream assessments. Enhances user engagement and ensures timely communication of important information.

2.1.1.3. Software components design

The diagram in Figure 1 below presents the backend software system's key parts (components) and their interactions. The OAH Citizen Science App context involves specifying how each part of the backend architecture functions, how they communicate with each other, and how they collectively support the app's features and requirements. This design ensures that the system is modular, scalable and maintainable.

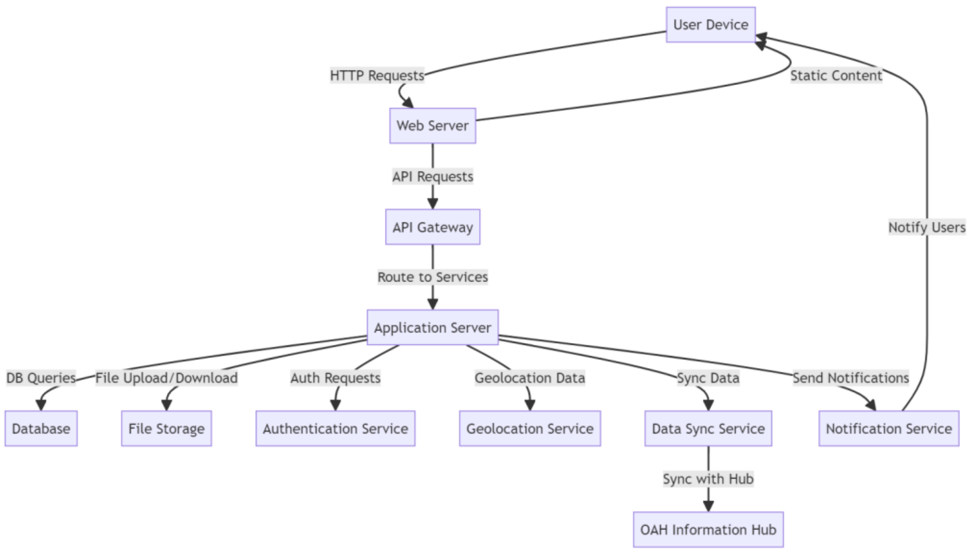


Figure 1 Citizen Science technology stack

Software components summary

User Device: The user's device (mobile, tablet, or desktop) interacts with the app through a web browser. The PWA ensures a seamless experience, providing offline capabilities and fast loading times.

Responsibility: Interface for the user to interact with the app.

Interactions: Sends HTTP requests to the web server, receives responses, and displays content. Handles offline storage and synchronisation.

Web Server: The web server handles incoming HTTP requests, serves static files (HTML, CSS, JS), and forwards API requests to the API Gateway.

Responsibility: Handles HTTP requests, serves static content, and forwards API requests.

Interactions: Communicates with the user device and the API Gateway.

Interface: HTTP endpoints for static files and API requests.

Application Server: The core backend logic resides here. It processes user requests, interacts with the database, handles authentication, processes geolocation data, manages file uploads and communicates with other services.

Responsibility: Contains the core business logic, processes user requests and interacts with databases and other services.

Interactions: Communicates with the database, file storage, authentication service, geolocation service and other internal services.

Interface: RESTful APIs for business operations internal service APIs.

Database: The relational and NoSQL databases store structured data (user profiles, stream assessments) and unstructured data (media files metadata).

Responsibility: Stores structured and unstructured data.

Interactions: Receives queries from the application server and returns data.

Interface: SQL/NoSQL queries, database connection protocols.

File Storage: Cloud storage solutions are used to store photos and videos uploaded by users, ensuring scalability and durability.

Responsibility: Stores media files (photos and videos).

Interactions: Handles file uploads and downloads from the application server.

Interface: APIs for file operations (upload, download, delete).

Authentication Service: This service manages user login, registration, and session management using secure authentication protocols.

Responsibility: Manages user authentication and authorisation.

Interactions: Validates user credentials, issues tokens, and verifies token validity for the application server.

Interface: OAuth2, JWT, authentication endpoints.

Geolocation Service: Processes geotagged data from user-submitted photos and maps them accurately on the app interface.

Responsibility: Processes and validates geotagged data.

Interactions: Receives geolocation data from the application server and returns processed location information.

Interface: Geolocation APIs.

API Gateway: Acts as a reverse proxy to manage API requests, route them to appropriate services, and ensure secure communication.

Responsibility: Manages API traffic, routes requests to appropriate services, and ensures security.

Interactions: Routes user requests to the application server and other services.

Interface: API endpoints for routing requests.

Data Sync Service: Ensures that data collected through the app is synchronised with the OAH Information Hub, facilitating broader data analysis and reporting.

Responsibility: Synchronises data between the app and the OAH Information Hub.

Interactions: Receives data from the application server and sends data to the OAH Information Hub.

Interface: Data synchronisation APIs.

Notification Service: Sends real-time notifications to users regarding updates, reminders, and alerts about their contributions and the overall project.

Responsibility: Sends notifications to users.

Interactions: Receives notification requests from the application server and sends notifications to user devices.

Interface: Notification APIs (e.g., push notifications, email).

2.1.2 User walkthrough of the OAH Citizen Science App

The application in its current version is presented in the sections below, and the link is: https://app.enora-oah.eu/citizens/submission.

This tab includes a section with guides, as shown in Figure 2, where we will demonstrate how a user can submit observations.

When the user selects the option "Citizen Science," a pop-up window will appear asking for their location, as presented in Figure 3.

CITISCI GEOSSIP New Citizens Science Submission enora_user1

Citizen Science Submit

Citizen Science List

Next →

Submission guide

What is an Urban Stream? – Streams in all landscapes!

When asked to visualize a running stream, you won't envision skyscrapers, office buildings, city buses, or stores. Nonetheless, there are urban streams coexisting with urban development, typically greatly influenced by the humans residing in their watershed. Streams are vital components of all landscapes, including forests, cities, and suburbs. Streams, stream banks, and the lowlands around them provide important habitat for animals, plants and many other organisms that share landscape with us. They are also part of a drainage network for rain and melting snow from our streets, parks, and yards.

Healthy stream ecosystems

Healthy streams must be surrounded by specific vegetation – riparian vegetation – capable of living with their roots in saturated or high humidity soil without rotting. Forested streams are full of life (from micro to macroscopic, terrestrial to aquatic) including mammals, birds, fish, amphibians, reptiles, terrestrial and aquatic invertebrates, fungi and high biomass of microorganisms constituting intricate food networks. Some of the aquatic organisms are very efficient in breaking down all vegetation litter that drop in the water or are pushed there by any surface runoff after rain, gradually cleaning the water of any organic litter coming from surroundings.

Ecosystem services

If well preserved streams are important biodiversity reservoirs, providing a myriad of ecosystem services to the human population such as better air quality, temperature extremes and floods mitigation, offering leisure and social areas, as well as outdoor laboratories for environmental education. Thus, these streams sustain **Ecosystem functions and habitats** (e.g., primary production, nutrients cycle, and decomposition), **provisioning services** (e.g., water and food supply for people and all other organisms), **regulating services** (e.g., flow, cycle, and purification of water, recycling soil sediments) and **cultural services** (e.g., recreation, leisure, and aesthetics, but also educational and spiritual services).

Important concepts

When looking **downstream** (to where the stream runs), the terrain alongside the bed (channel) of a stream in its left is defined as **left bank** of the stream, while the right is defined as the **right bank** of the stream (Fig.1A). In a **stream cross-section** we can measure the **width** of the stream channel (more or less perpendicular to the channel), and the **depth** of the stream that may vary along the cross-section of the stream but can be easily measured (Fig.1B).

Fig. 1A Fig. 1B

A healthy riverine ecosystem besides the river channel includes healthy **riparian zone** (full of a variety of typical vegetation of the region and adapted to the proximity of water, e.g., Fig2A) and areas of low-lying ground adjacent to streams – **floodplain** – formed mainly of nutrient-rich river sediments and subject to flooding after storms and heavy snowmelt (Fig.2B). The shape and nature of the floodplain may change over time, because a stream's main channel naturally shifts due to accretion and erosion, which affects when and where excess of water may initially overflow the streams's banks during a heavy rain flood event.

Fig. 2A Fig. 2B

Challenges of urban streams

Urban stream ecosystems are crucial for enhancing biodiversity and sustainability in cities by connecting people, animals and plants, and many other organisms. Yet, they face challenges such as the lack of space, vegetation loss, artificialization, and other urbanisation processes. This degradation can lead to numerous disservices to humans, including exposure to emerging pathogens, reduced disease resistance, and exacerbated climate change impacts, and other health concerns in cities.

Main differences between streams in forests and streams in urban areas

A big issue concerning urban streams is how impervious they are compared to streams that flow through more natural areas. The urban development around these streams increases the amount of impervious surfaces. These include surfaces such as concrete, asphalt, and many other man altered soil structures (Fig.3). These kinds of surfaces do not allow water to infiltrate as easily as forest ground or open fields. This becomes an issue when it rains or storm as the high number of impervious surfaces causes stormwater to quickly raise the water level of the river, raising concerns of flooding. It may than erodes stream channels and causes streams to be incised. So, unpreserved urban streams often suffer from **more flooding during rainstorms**. However, also water scarcity **during dry seasons, sometimes caused by water retention by dams or water abstraction, makes** the flow management and rehabilitation of urban streams challenging. Additionally, urban streams are often polluted by urban runoff and combined sewer outflows.

Swimming Pool
Roof
Roof
Patio
Deck
Driveway
Property Boundary
Private Walkway
Pervious
Impervious Surfaces

Fig. 2A

The project of OneAquaHealth

OneAquaHealth aims to improve urban freshwater ecosystems' sustainability promoting OneHealth and restoring aquatic ecosystem health, by identifying early warning indicators of degradation and enhancing environmental monitoring using AI-assisted tools. Hence, to improve urban aquatic site management through the development of digital tools like an Environmental Surveillance System, Decision & Support System, and Citizen Science App to raise awareness and to engage all relevant stakeholders and to jointly achieve thriving ecosystems and healthier communities for the future.

What is OneHealth?

OneHealth is a comprehensive strategy aimed at promoting sustainable and optimal health for humans, all other organisms, and ecosystems. Well-preserved urban natural areas support several ecosystem services and function as restorative environments capable of providing physical and mental wellbeing to the populations that enjoys them. The existence of these natural areas near home within cities, provides people stimulus to leave home, promoting social interaction, reducing stress and offering a diverse and pleasant environment full of new sensory experiences. Urban streams can there be viewed as wonderful laboratories to promote the collaborative work of many stakeholders to effectively apply the OneHealth perspective.

What can you do with us?

If you have an urban stream nearby, sometimes in hidden valleys between the hills of your city, even if heavily modified (with concrete walls looking like a duct), you can help us taking care of it, by looking into its channel, bank and margins/riparian zone and providing us relevant information and photos through this app. And ultimately, letting us know what's your overall assessment of the stream in the following categories:

- Good quality (the ecosystem components are present and look good: riparian vegetation, natural channel, good water quality, biodiversity)
- Moderate quality (there are some alterations, some biodiversity, with some vegetation in the margins, and water looks good ...)
- Poor quality (it is highly modified/artificial, with major losses of riparian vegetation or only invasive species, loss of habitats, the water is polluted)

Figure 2 Citizen Science - Submission Guide

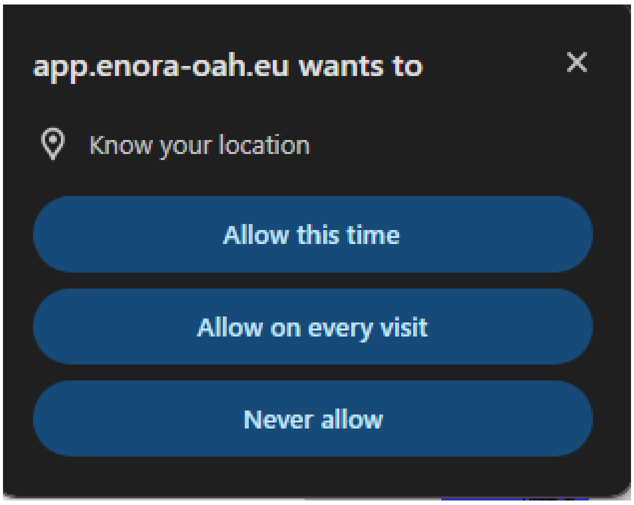


Figure 3 Request for location

After clicking "Next", a map displaying the five research cities and sites appears, allowing users to select their site of interest. The app automatically calculates the distance between the user's current location and all available sites. This feature aims to empower users with knowledge about their nearest OneAquaHealth endpoints. At this stage, the user cannot add his own sites because the CitizenScience App complements the research (in-situ and EO data analysis) that is in progress in selected locations and sampling sites. However, via the GEOSSIP application, scientists can add new areas that are offered for citizen information collection.

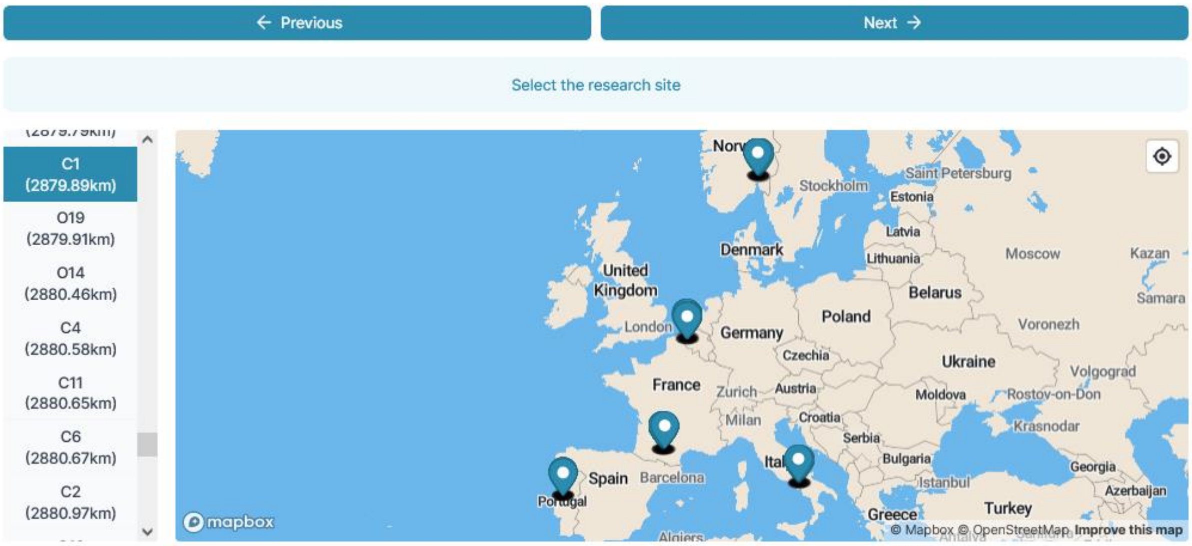


Figure 4 Citizen Science - Site selection

The user chooses the first option (Coimbra – C1) for demonstration purposes, as shown in Figure 4.

After selecting a specific site, the user can upload their images and a short video, as displayed in Figure 5, corresponding to upstream and downstream, as well as pictures with surrounding context and elements of biodiversity.

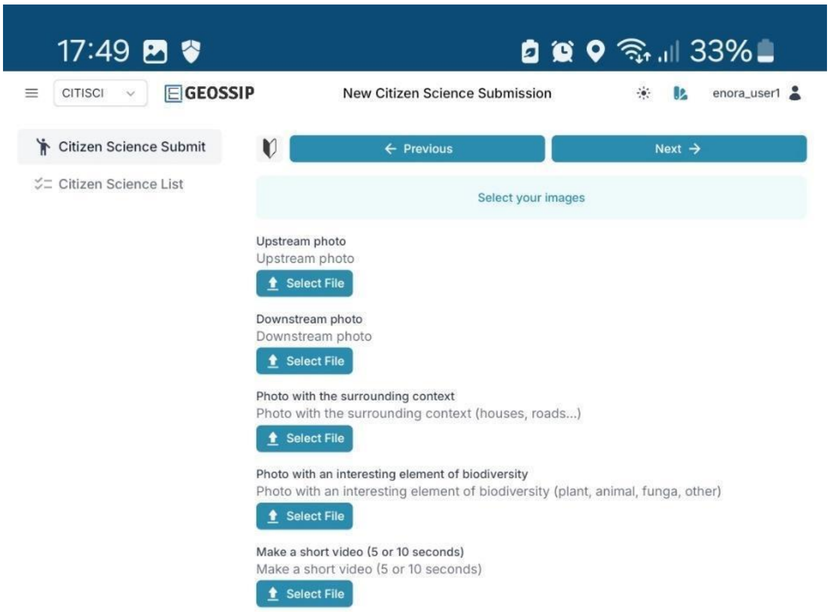


Figure 5 Citizen Science - Image and video upload

Following this, there are three steps of questions, as provided in Figures 6, 7 and 8, leading to the overall assessment of the stream.

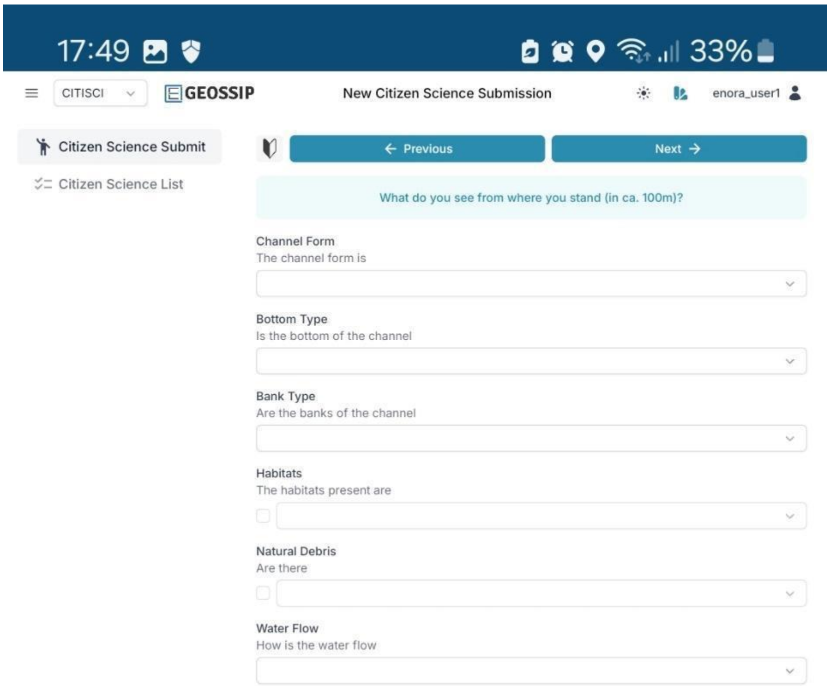


Figure 6 Citizen Science App - First question set

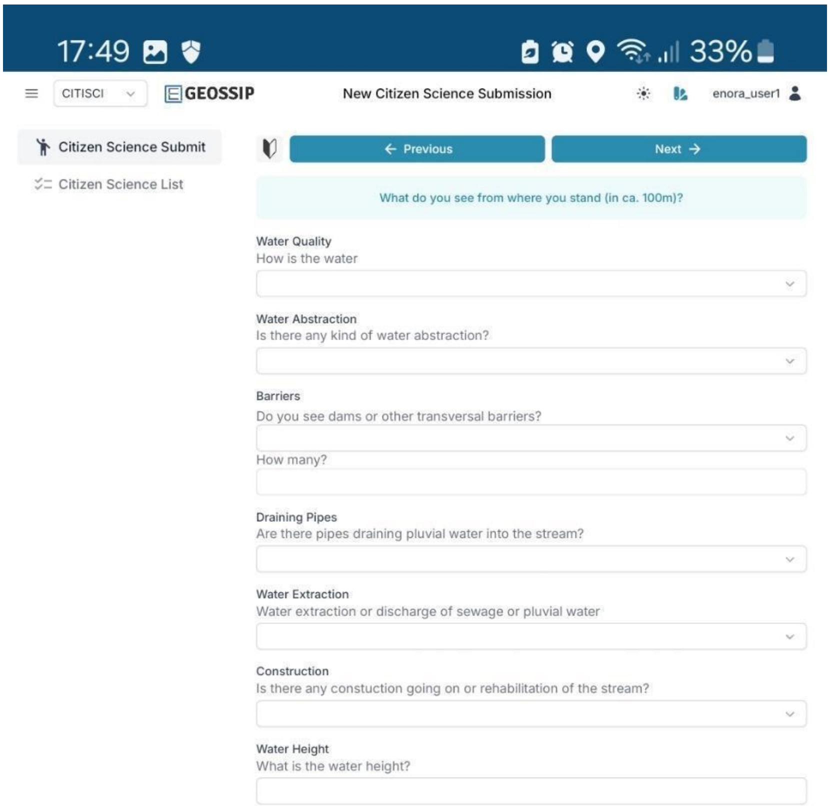


Figure 7 Citizen Science App - Second question set

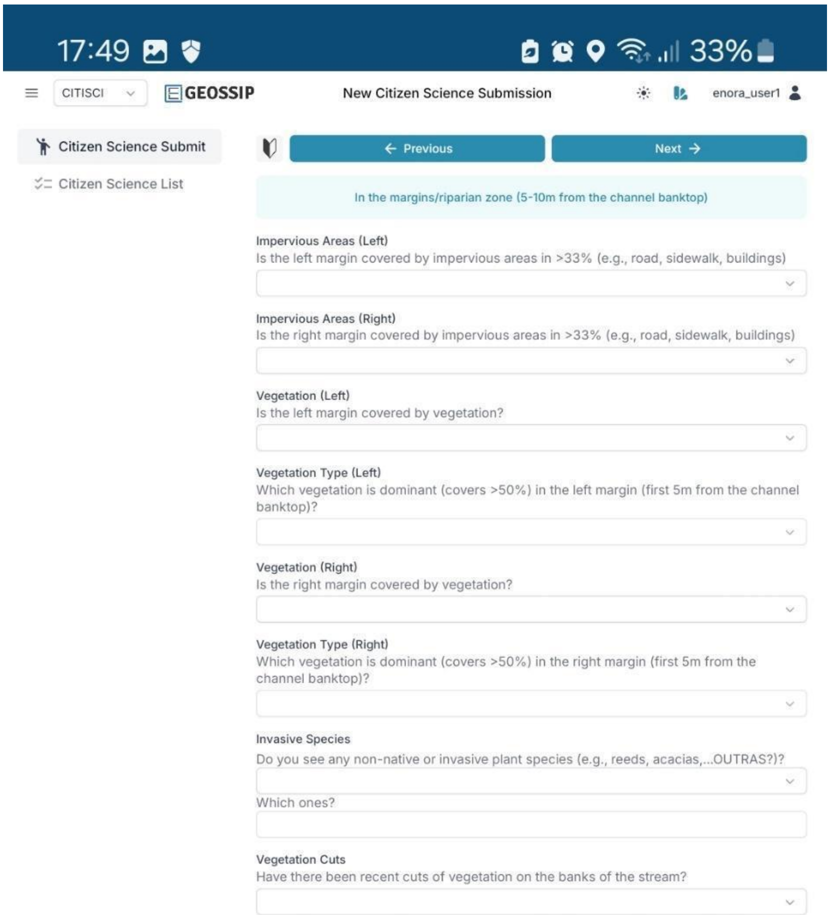


Figure 8 Citizen Science App - Third question set

After choosing the overall assessment (good, moderate or poor quality), as shown in Figure 9, and A summary of the submission is provided before submission.

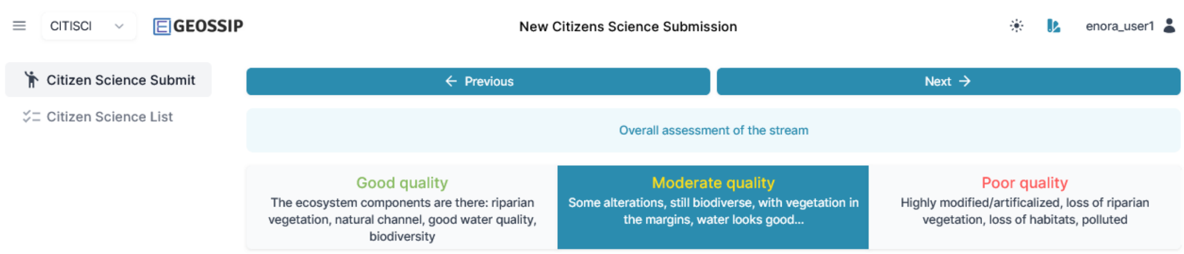
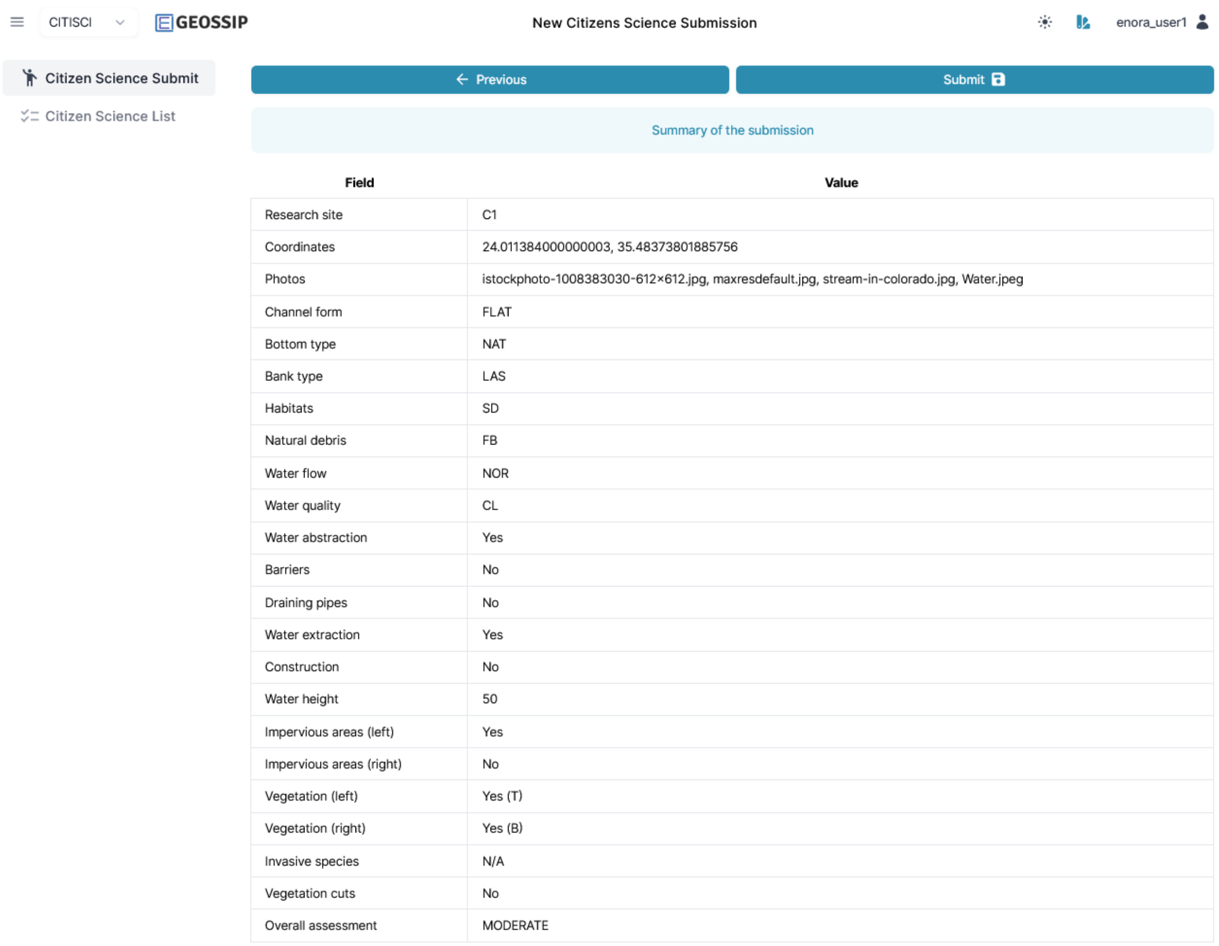


Figure 9 Citizen Science App - Overall assessment

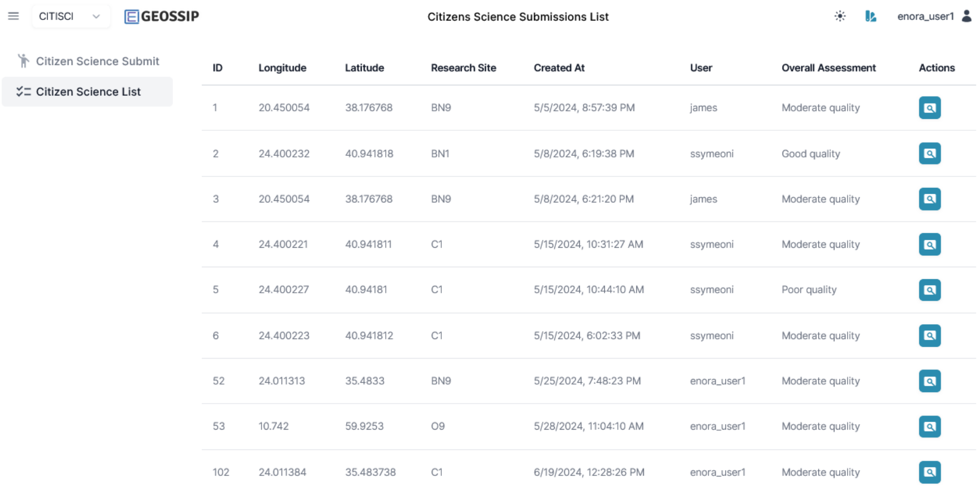
Finally, the user can complete the submission process, and the observation is submitted and stored, as presented in Figure 10.



Field	Value
Research site	C1
Coordinates	24.011384000000003, 35.48373801885756
Photos	istockphoto-1008383030-612×612.jpg, maxresdefault.jpg, stream-in-colorado.jpg, Water.jpeg
Channel form	FLAT
Bottom type	NAT
Bank type	LAS
Habitats	SD
Natural debris	FB
Water flow	NOR
Water quality	CL
Water abstraction	Yes
Barriers	No
Draining pipes	No
Water extraction	Yes
Construction	No
Water height	50
Impervious areas (left)	Yes
Impervious areas (right)	No
Vegetation (left)	Yes (T)
Vegetation (right)	Yes (B)
Invasive species	N/A
Vegetation cuts	No
Overall assessment	MODERATE

Figure 10 Citizen Science App - Submission

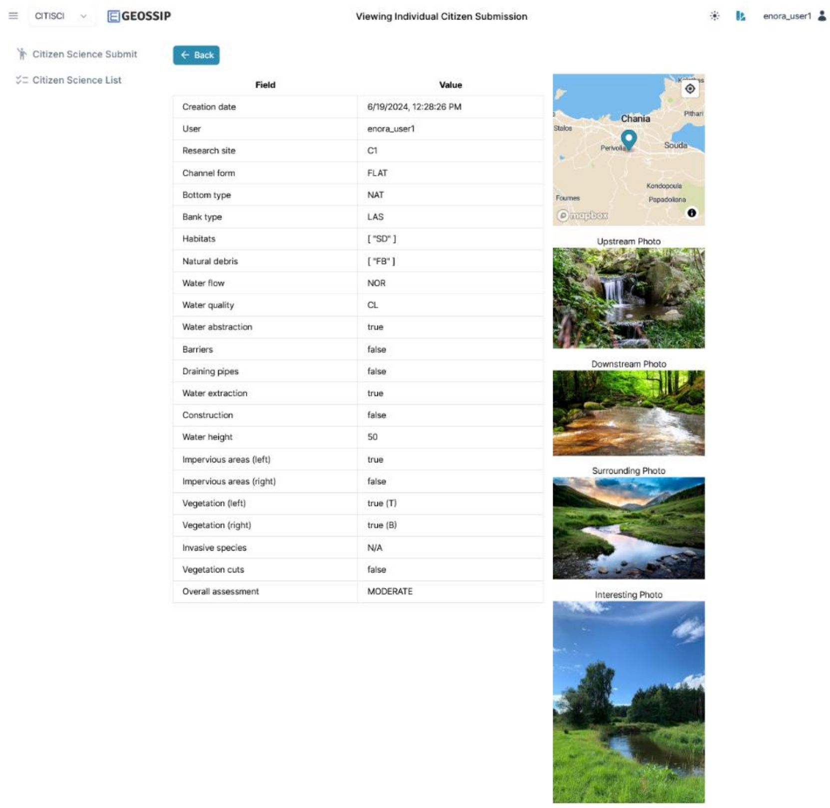
The list of the citizen science submissions with ID, Longitude, Latitude, Research Site, Date, User and Assessment are shown in Figure 11.



ID	Longitude	Latitude	Research Site	Created At	User	Overall Assessment	Actions
1	20.450054	38.176768	BN9	5/5/2024, 8:57:39 PM	james	Moderate quality	
2	24.400232	40.941818	BN1	5/8/2024, 6:19:38 PM	ssymeoni	Good quality	
3	20.450054	38.176768	BN9	5/8/2024, 6:21:20 PM	james	Moderate quality	
4	24.400221	40.941811	C1	5/15/2024, 10:31:27 AM	ssymeoni	Moderate quality	
5	24.400227	40.94181	C1	5/15/2024, 10:44:10 AM	ssymeoni	Poor quality	
6	24.400223	40.941812	C1	5/15/2024, 6:02:33 PM	ssymeoni	Moderate quality	
52	24.011313	35.4833	BN9	5/25/2024, 7:48:23 PM	enora_user1	Moderate quality	
53	10.742	59.9253	O9	5/28/2024, 11:04:10 AM	enora_user1	Moderate quality	
102	24.011384	35.483738	C1	6/19/2024, 12:28:26 PM	enora_user1	Moderate quality	

Figure 11 Citizen Science App - List of Submissions

By clicking on the "Actions" button of one element of the list, the following screen with the submission details appears, and the results are displayed in Figure 12.



Field	Value
Creation date	6/19/2024, 12:28:26 PM
User	enora_user1
Research site	C1
Channel form	FLAT
Bottom type	NAT
Bank type	LAS
Habitats	["SD"]
Natural debris	["FB"]
Water flow	NOR
Water quality	CL
Water abstraction	true
Barriers	false
Draining pipes	false
Water extraction	true
Construction	false
Water height	50
Impervious areas (left)	true
Impervious areas (right)	false
Vegetation (left)	true (T)
Vegetation (right)	true (B)
Invasive species	N/A
Vegetation cuts	false
Overall assessment	MODERATE

Figure 12 Citizen Science App - Submission Preview

2.1.3 Installation of the OAH Citizen Science App

The following Figures 13 to 17 show the OAH Citizen Science App installation steps on a mobile device.

Figure 13 Visit the Citizen Science app

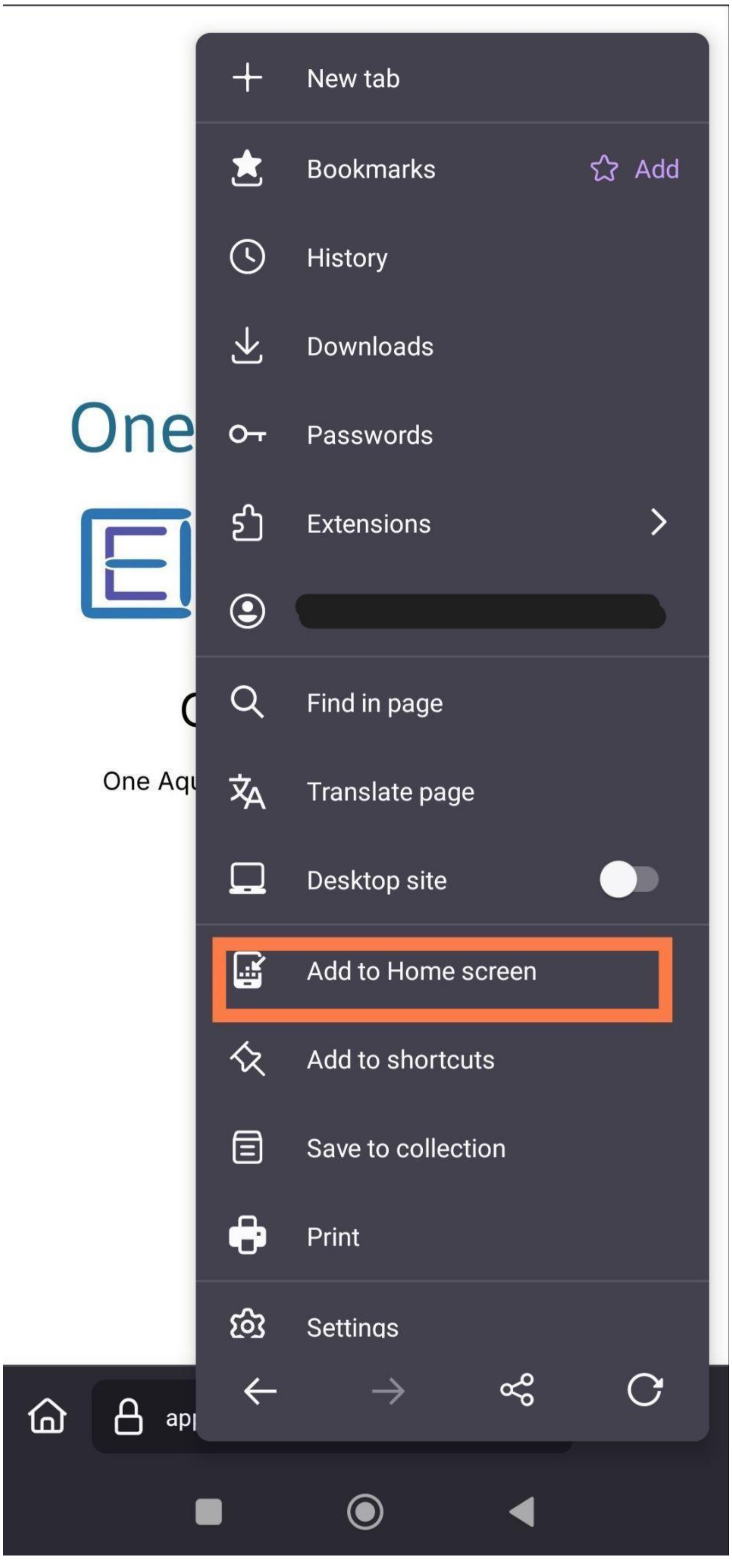


Figure 14 Installation of the app

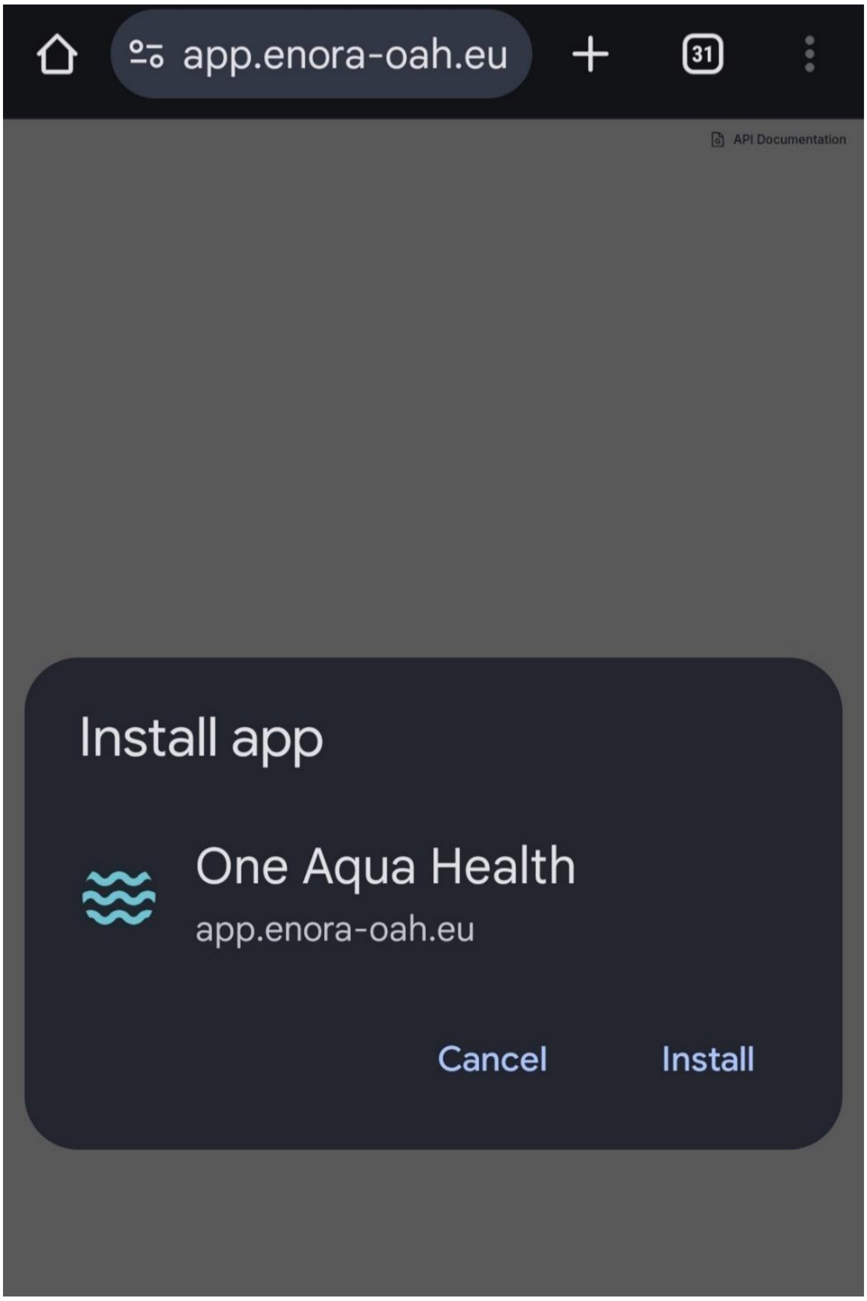


Figure 15 Acceptance for installation of Citizen Science App

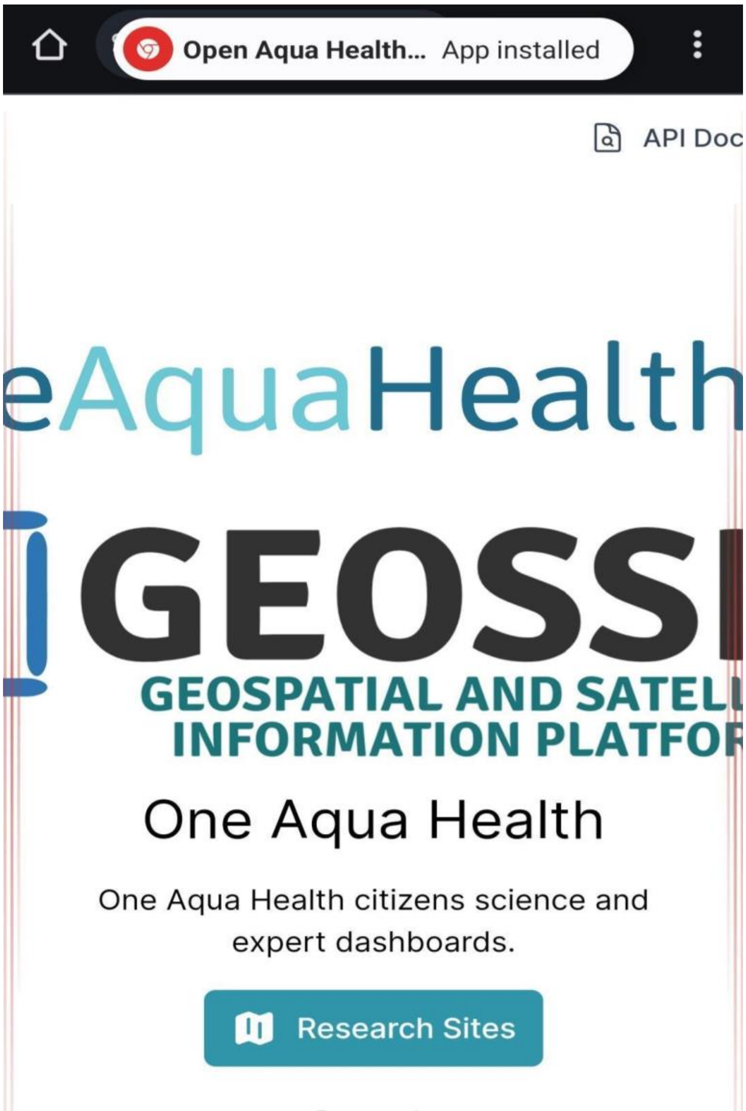


Figure 16 Installation message

Figure 17OAH Citizen Science App installed on the mobile apps

3 Conceptual Architecture

The conceptual architecture and design of the OneAquaHealth (OAH) are designed to manage and present geospatial and environmental data. This section outlines the architecture's layers, detailing the components. It also explains the role of interactive endpoints and the integration of various user-facing applications within the platform. The aim is to provide solutions for accessing and processing data, enhancing citizen science initiatives, and supporting experts.

3.1 Design and Architecture

The following Figure 18 presents the proposed High-level architecture as presented in *D6.1 OneAquaHealth Open Information Hub vs1* in Month12.

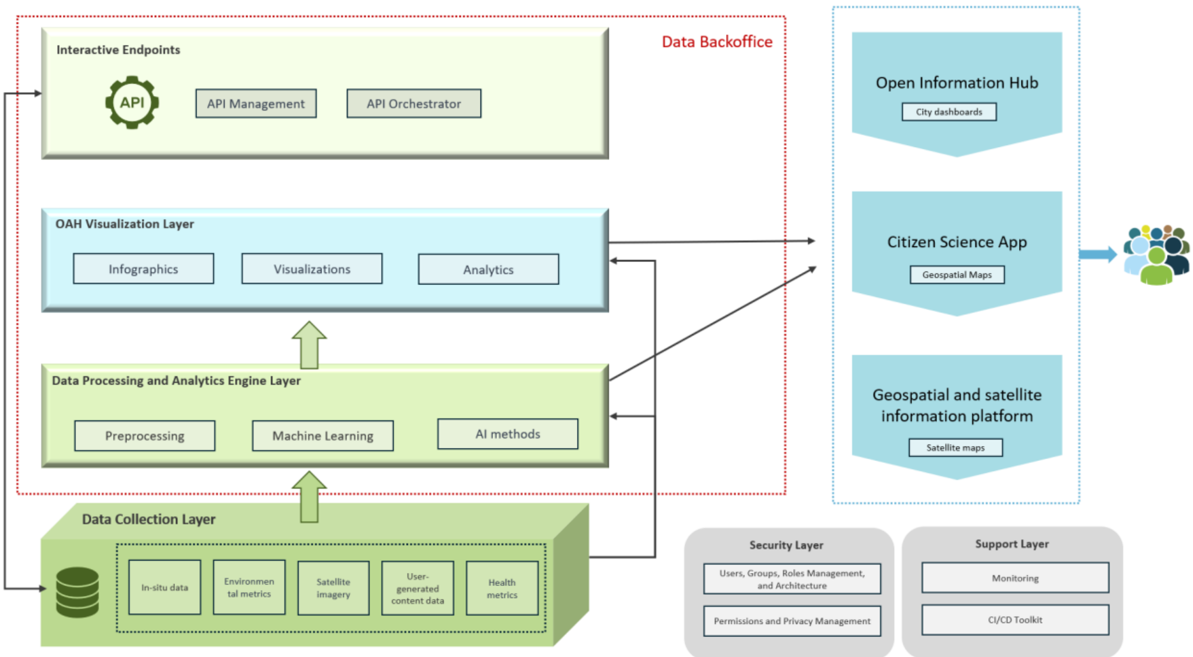


Figure 18 OAH High-level architecture of applications

The OAH architecture is structured into several interconnected layers, each responsible for distinct functionalities.

3.1.1 Data Collection Layer

This layer aggregates diverse data sources, including in-situ data, environmental metrics, satellite imagery, user-generated content, and health observations. It serves as the repository for all raw data inputs, ensuring comprehensive data coverage for subsequent processing and analysis.

3.1.2 Data Processing and Analytics Engine Layer

As mentioned above, the data collection layer handles the preprocessing of raw data, applies machine learning algorithms, and utilises AI methods to extract meaningful insights. It transforms raw data into analysis-ready formats, enabling efficient data manipulation and interpretation.

3.1.3 OAH Visualization Layer

This layer provides tools for creating infographics, visualisations, and analytics. It leverages the processed data to generate intuitive and informative visual representations, facilitating data-driven decision-making and user engagement.

3.1.4 Interactive Endpoints

Comprising the API Management and API Orchestrator components, this layer ensures seamless interaction between the platform's backend processes and user applications. It enables external applications to access and manipulate data through well-defined API endpoints, supporting custom integrations and advanced functionalities.

3.1.5 User Applications

The architecture connects several user-facing applications, including the Open Information Hub, Citizen Science App, and the Geospatial and Satellite Information Platform. These applications provide interfaces for different user groups, allowing them to access city dashboards, geospatial maps, and satellite imagery.

3.1.6 Security and Support Layers

These auxiliary layers ensure the platform's security and operational efficiency. The Security Layer manages user authentication, roles, permissions, and privacy settings, while the Support Layer oversees system monitoring and the CI/CD toolkit for continuous integration and deployment.

3.2 Data Management

The data management of the GEOSSIP platform is based on Deliverables D7.6 (Data Management Plan vs1) and D7.7 (Data Management Plan vs2), respectively, and focuses on systematic data handling and organisation to support efficient processing, analysis, and retrieval. Data from diverse sources, including in-situ measurements, satellite/Earth Observation(EO) analytics, user-generated content, and health indicators, are aggregated in the Data Collection Layer. These raw data are then categorised and stored in a secure and scalable repository. Data preprocessing steps include filtering, cleansing, validation, and normalisation to ensure consistency and quality across datasets. Once processed, the data is stored in various formats (e.g., structured databases, JSON files) depending on its intended use, whether for analytics, visualisation, or API access. Data organisation follows a layered model where raw, processed, and analysis-ready data are logically separated to maintain data integrity and facilitate version control. This structured approach ensures that data is easily accessible for machine learning algorithms, visualisations, and end-user applications while maintaining compliance with data privacy and security regulations.

3.3 Authentication and Authorisation

The GEOSSIP platform incorporates a multi-layered authentication and authorisation mechanism to secure data access and ensure user privacy. The platform utilises OAuth 2.0 with JSON Web Tokens (JWT) for secure access delegation, allowing users to authenticate. Role-Based Access Control (RBAC) is implemented to manage authorisation, where user roles (e.g., admin, researcher, citizen) dictate the permissions and access levels. During registration, users create an account by providing the necessary details. Access controls are enforced at multiple layers, including API endpoints and data repositories, to ensure users can only interact with data and functionalities relevant to their roles.

3.4 API

The GEOSSIP platform's API layer serves as the primary interface for data access, interaction, and integration with external applications. APIs are designed using RESTful architecture, providing standardised endpoints for different functionalities, including data retrieval, submission, and processing requests. The creation of these APIs involves defining explicit data schemas and models that support diverse data types, from spatial datasets to health indicators. Key features include pagination, filtering, and query customisation to optimise data fetching. Authentication for API access is enforced through API keys and OAuth/JWT tokens, ensuring that only authorised users or applications can interact with the platform.

Additionally, the API layer includes an orchestrator component that manages request routing, load balancing, and response caching, enhancing the overall efficiency and reliability of data exchange. The complete API documentation guides developers in seamlessly integrating their applications with the platform. Figures 19 and 20 present the Swagger documentation of EO and GEOSSIP APIs.

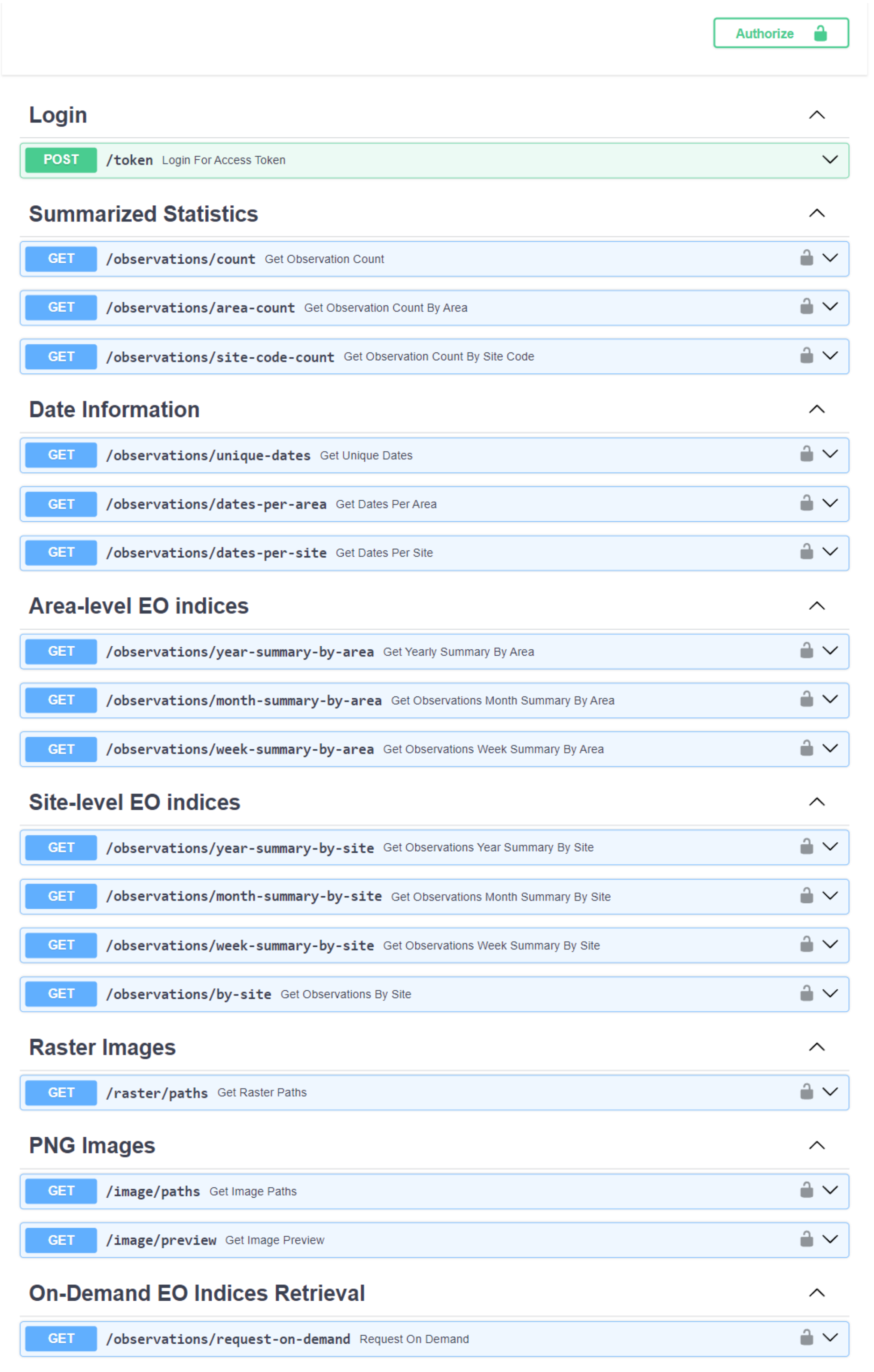


Figure 19 EO APIs

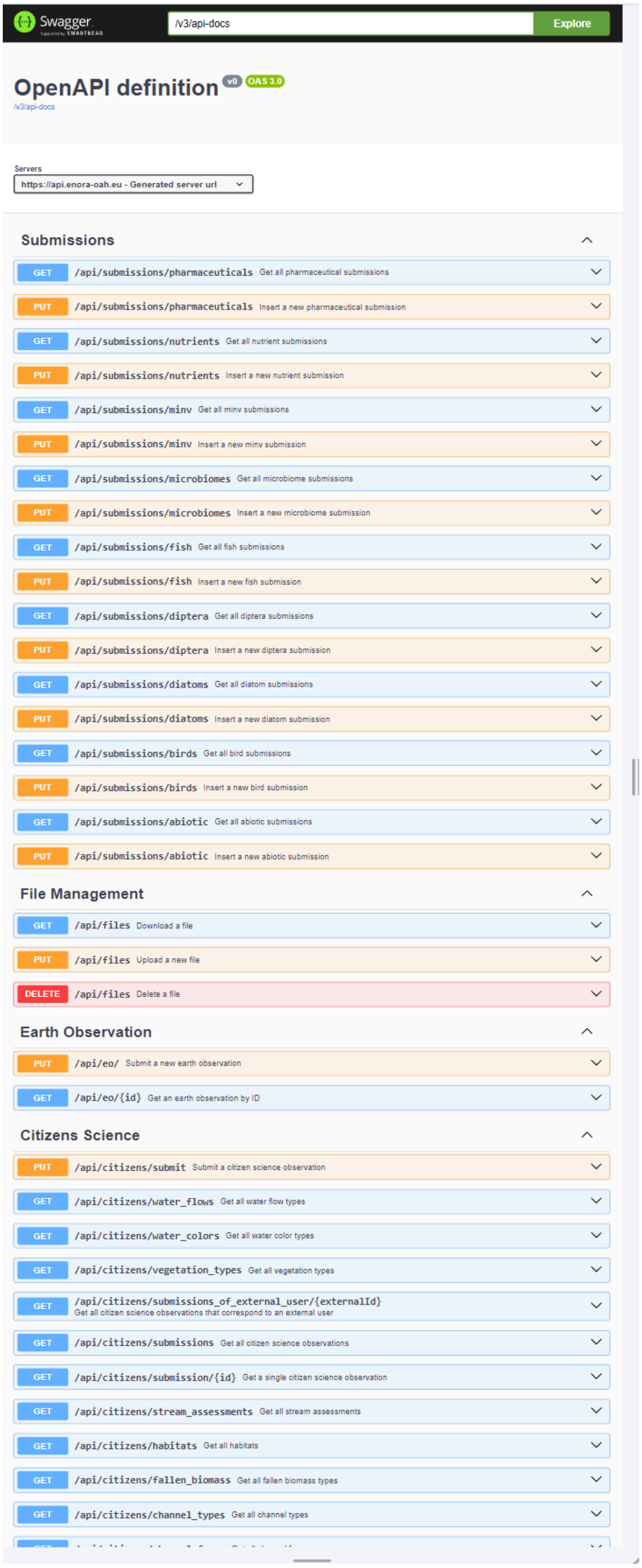


Figure 20 GEOSSIP APIs

4 Data Backoffice Development

4.1 Architecture Design: Overview of the Data Backoffice Architecture

The Data Backoffice architecture for the software is designed to be robust, scalable and flexible. It integrates diverse data sources and systems, ensuring seamless data persistence, processing and visualisation. This architecture supports the platform's mission to monitor and improve aquatic ecosystems and human health through comprehensive data integration and analysis.

Key components include the Persistence Layer, which stores data in relational databases, NoSQL databases and filesystems. This component also integrates external data from open APIs, EU Data Spaces and related scientific datasets. The Controller Layer orchestrates data flow with an ML Backend for machine learning processes and an Integration Backend for data consistency and transformation. The View Layer provides user interfaces for data visualisation and interaction, including a front end and headless routes for API access.

In addition, the architecture integrates with external systems such as the Earth Observation system for environmental data, OAH User Persistence for secure user management and the OAH Information Hub, which supports a Citizen Science App and interactive dashboards. This flexible design ensures effective data management and user engagement, contributing to the platform's goals of enhancing aquatic health monitoring and analysis, as shown in Figure 21.

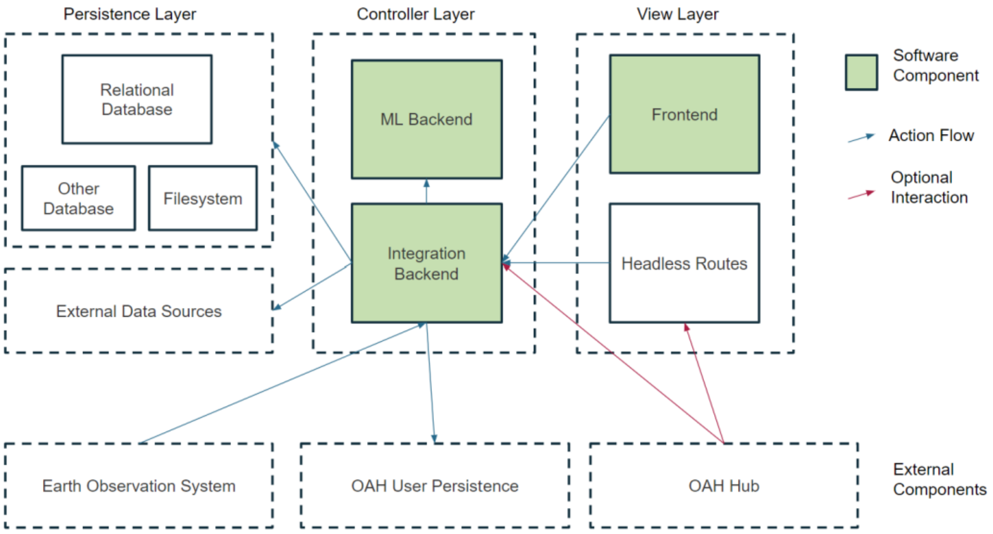


Figure 21 The software architecture of the OneAquaHealth backend

The architectural components and their interactions are summarised below.

1. Persistence Layer

The Persistence Layer is responsible for storing and managing all the system's required data. It consists of:

- Relational Database: Used for structured data storage, supporting complex queries and transactions. Typical databases include MySQL, PostgreSQL and Oracle.
- Other Databases: Include NoSQL databases (e.g., MongoDB, Cassandra) for unstructured or semi-structured data, providing flexibility and scalability.

- Filesystem: Used for storing large files, logs and other binary data which might not fit well in traditional databases.
- Additionally, the Persistence Layer integrates data from external sources:
- Open Data APIs: Publicly available APIs that provide health data relevant to aquatic ecosystems.
- EU Data Spaces: Initiatives by the European Union that provide large-scale, standardised datasets.
- Publicly Available Scientific Datasets: Datasets published by research institutions and organisations that are relevant to the platform's objectives.

2. Controller Layer

The Controller Layer orchestrates the data flow between the Persistence and View Layer. It comprises:

- Integration Backend: Handles the integration of various internal and external data sources and services. This includes API management, data exchange, data transformation, session management, authentication, authorisation and ensuring the consistency and reliability of the integrated data.
- ML Backend: Machine Learning models and algorithms that process data to generate predictions, insights, and visualisations.

3. View Layer

The View Layer is responsible for presenting data to the end-users. It includes:

- Front End: The user interface (UI) for interacting with the data provides tools for visualisation, exploration and manipulation.
- Headless Routes: API endpoints that support a headless architecture, allowing data to be accessed and manipulated by different front-end applications and services.

Integration Points

1. Geospatial and satellite information platform

The architecture integrates with the Earth Observation system via an API. This system provides crucial environmental data fed into the OAH platform for analysis and visualisation. The OAH Geospatial and satellite information platform is presented in more detail in deliverable D5.1.

2. OAH Information Hub

The Information Hub serves as the central repository for information dissemination. It supports:

- Citizen Science App: A mobile or web application that allows citizens to contribute data and engage with the platform.
- Dashboards: Interactive dashboards that visualise data and predictions, providing insights into aquatic health at OAH sites.

Detailed Architecture Flow

1. Data Ingestion:

- Data is ingested from diverse sources, including relational databases, NoSQL databases, filesystems and external APIs.

- Data from open data APIs, EU Data Spaces and scientific datasets are fetched and stored in appropriate storage systems.

2. Data Processing:

- The ML Backend processes the ingested data, applying machine learning algorithms to generate insights and predictions.
- The Integration Backend ensures that data from different sources is harmonised and ready for further use.

3. Data Storage:

- Processed data is stored back in the Persistence Layer, making it available for retrieval by the View Layer.
- The Relational Database handles structured data, while other databases and the filesystem manage unstructured data.

4. Data Visualisation and Interaction:

- The Front End retrieves data via the headless routes the Controller Layer provides.
- Users interact with the data through the front-end interfaces, visualising it using dashboards and contributing through the Citizen Science App.

5. External System Integration:

- The Earth Observation system's data is integrated via APIs, enriching the platform's dataset.
- The OAH User Persistence system ensures secure user access and management.

4.2 GEOSSIP: The OAH software platform

In the following sections, we present the GEOSSIP Software Platform that has been developed to meet the requirements of the backend architecture and serve as a single entry point to demonstrate all the OAH's software capabilities. The platform provides access to:

- Earth observation data is available for all OAH research sites based on the specifications described in D5.1.
- Data management for handling in-situ data from research sites.
- Visualisation of all information sources depicted on map layers.

The application in its current version is presented in the sections below, and the link is: https://app.enora-oah.eu/sites.

Login Screen: Initially, the user is presented with the main page, and by clicking on "Research Sites", they are directed to the login form. The login screen is the gateway to the GEOSSIP Software Platform, providing users with access to the system. Registered users can log in via JSON Web Tokens (JWT). Figures 22 and 23 present these screens.

Figure 22 GEOSSIP Home Page

Figure 23 Login Screen

Research Sites Map: Upon login, the user is presented with a map displaying the five main cities involved in the project, as shown in Figure 24. Users can see the available OneAquaHealth research sites when they zoom in on a specific city.

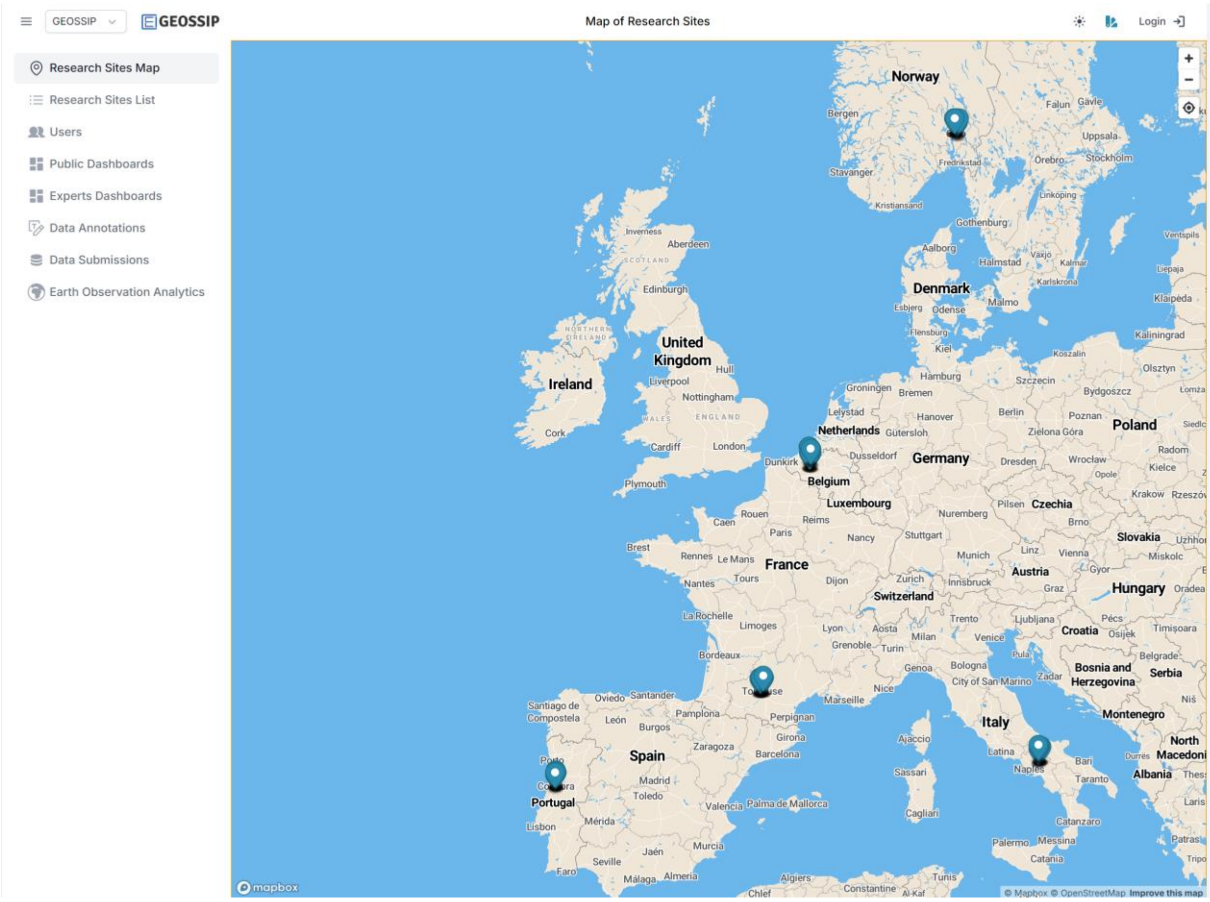


Figure 24 Research Sites Map

Research Sites List: The available sites are also listed in a tabular format, providing exact coordinates for each location, as presented in Figure 25.

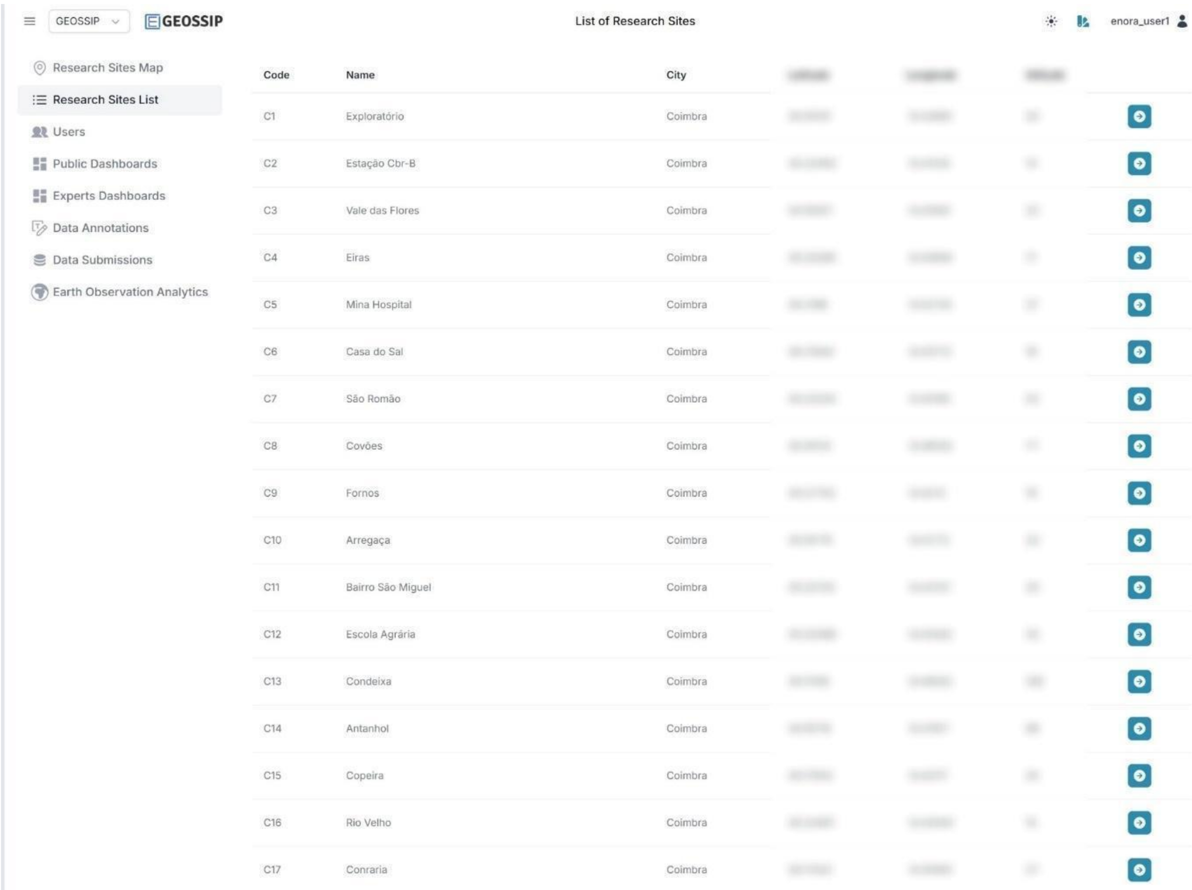


Figure 25 List of Research sites and coordinates

Sample of Expert Dashboard: The user can also interact with the demo dashboard, as presented in Figure 26, where they can select the city and site of interest (e.g., Coimbra - C1), as well as the date and quantity of interest. This interaction results in a tabular representation and a graphical visualisation for the specified city, site, date and quantity.

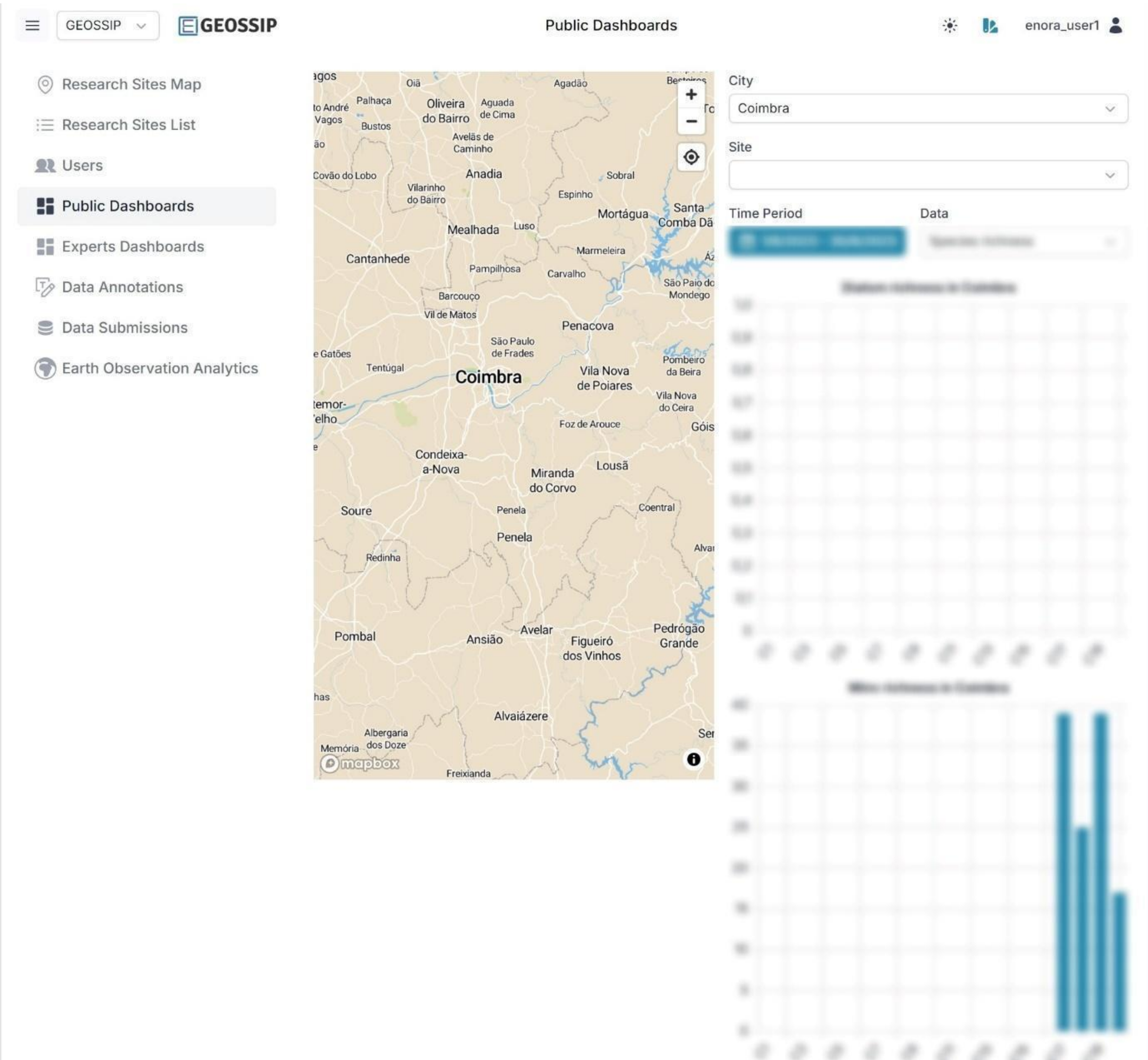


Figure 26 Sample Expert Dashboard for Nutrients

Data Annotations: There are four environmental types of lists: Biotics, Nutrients, Pharmaceuticals and Hydromorphology), as shown in Figure 27. These are the currently available data from WP2.

Figure 27 Categories of Data Annotations types

Data Submissions: When clicking on Data Submissions, as presented in Figure 28 and Figure 29, the user can see the data that has already been submitted to the platform, with each row representing an individual submission. Users can add observations for any research city and site at a specific time, corresponding to their observations. This applies to any of the categories (Biotics, Nutrients, Pharmaceuticals and Hydromorphology), each with its particular columns and categories to be completed. A successful submission is stored in the backend database.

Figure 28 Preview of Data Submissions

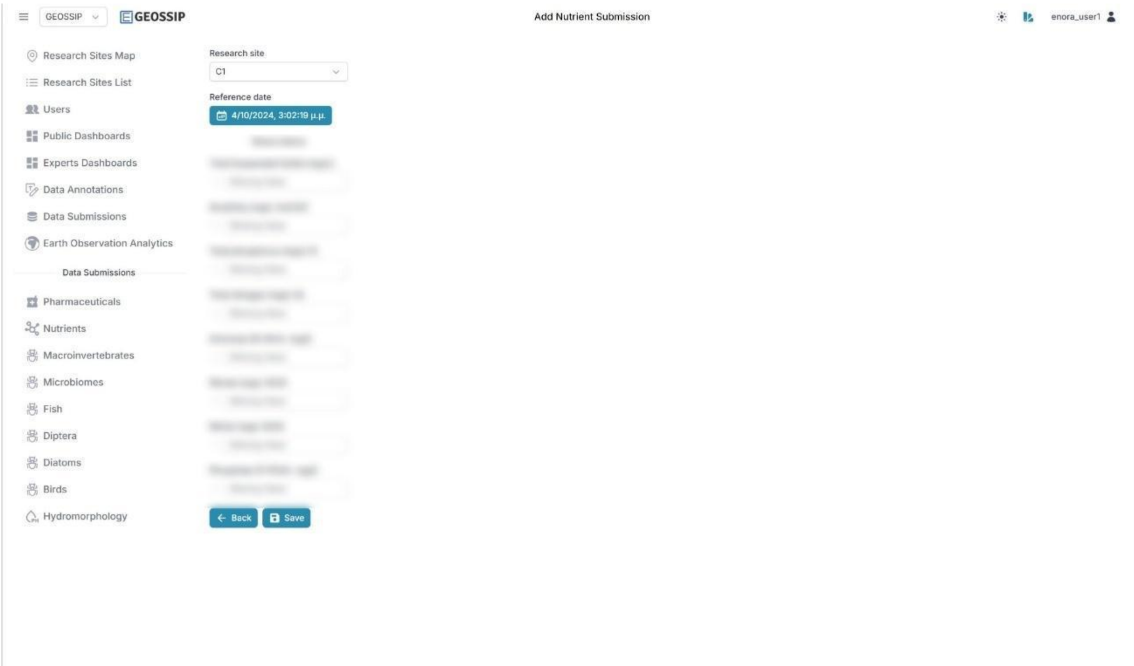


Figure 29 Data Submission Form

Figures 30 and 31 present the Earth Observation section, which shows the satellite imagery and the Earth Observation Indicators.

In Satellite imagery, the users can select an area of interest, apply an index from available indices, such as the Blue Wide Dynamic Range Vegetation Index or Normalised Difference Vegetation Index, and choose a specific date or range of dates to generate satellite observations. The detailed observation data table provides vital information, which is then visualised through the map, helping monitor changes over time.

In the Earth Observation Indicators module, the map displays the user's selected area of interest, while the sidebar allows the user to choose a region, research site, index type, and date to retrieve analytics. After selecting these options, the user can click 'Get Earth Observation Analytics' to view the data. The Observation Data Table below displays key metrics, and the opacity slider enables the user to compare different data layers. Using the 'Show Earth Observations Graph' option, the user can visualise trends, analyse changes over time, review analytics for the chosen index, and download the data to a file.

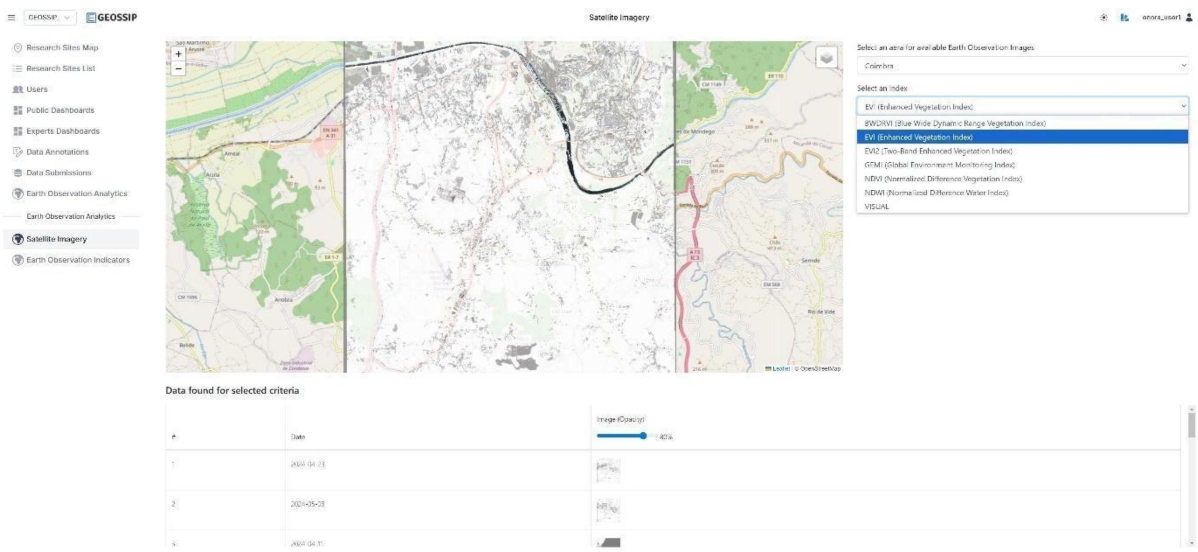


Figure 30 GEOSSIP - Satellite Imagery

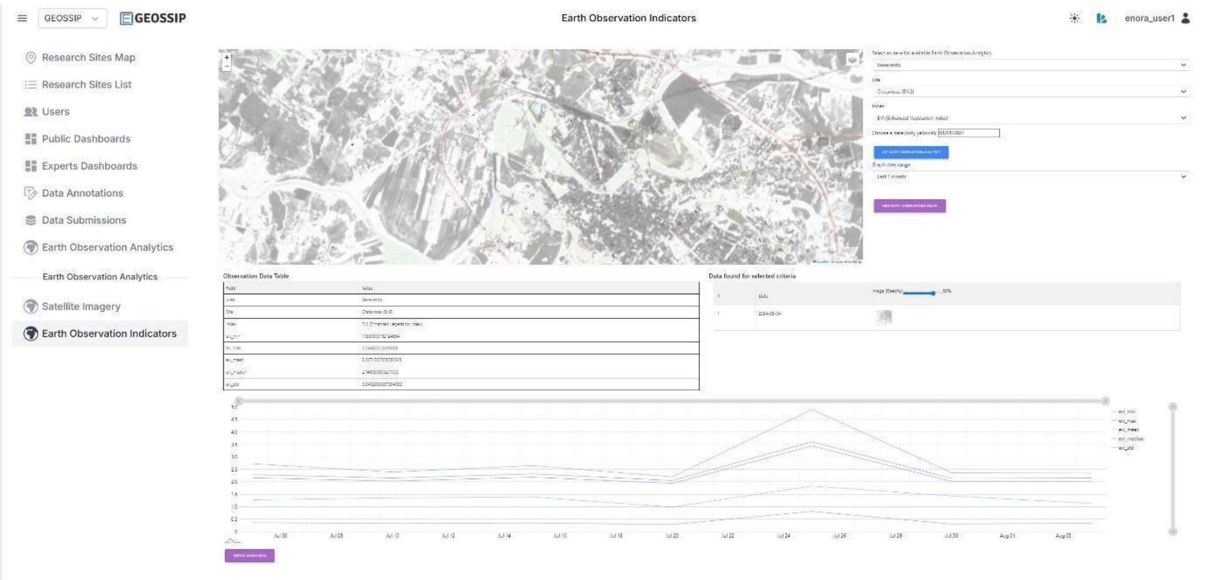


Figure 31 GEOSSIP - EO Indicators

5 Software development methodology: SCRUM methods and tools

In developing the Data Backoffice and the Citizen Science app for OneAquaHealth, we adopted the SCRUM methodology to ensure a flexible and efficient workflow. SCRUM, an agile framework, facilitates iterative progress through time-boxed sprints, typically lasting two to four weeks. Each sprint encompasses planning, execution, review, and retrospective phases, fostering continuous improvement and adaptation. This methodology enabled our team to manage complex tasks, quickly respond to changes, and integrate feedback from stakeholders and end-users. By emphasising collaboration, SCRUM facilitated regular communication among developers, researchers, and users, ensuring the final product met the diverse needs of our research sites and scientists.

To support SCRUM practices, we utilised a suite of tools designed to enhance productivity and transparency. Jira was employed for sprint planning, task tracking, and issue management, clearly representing progress and bottlenecks. Additionally, BitBucket was used for version control, enabling seamless code collaboration and integration. These tools collectively supported the SCRUM framework, ensuring our development process was agile and aligned with project goals.

User stories drove the implementation of the Data Backoffice and Citizen Science app to ensure that the development process remained user-centric and aligned with the needs of our stakeholders. Each user story, derived from the requirements gathered during user engagement activities, encapsulated specific functionalities and user expectations. These user stories were translated into detailed tickets within our project management tool, Jira, which outlined the tasks, acceptance criteria, and subtasks necessary for implementation. An example of this procedure is shown in Figures 32 and 33.

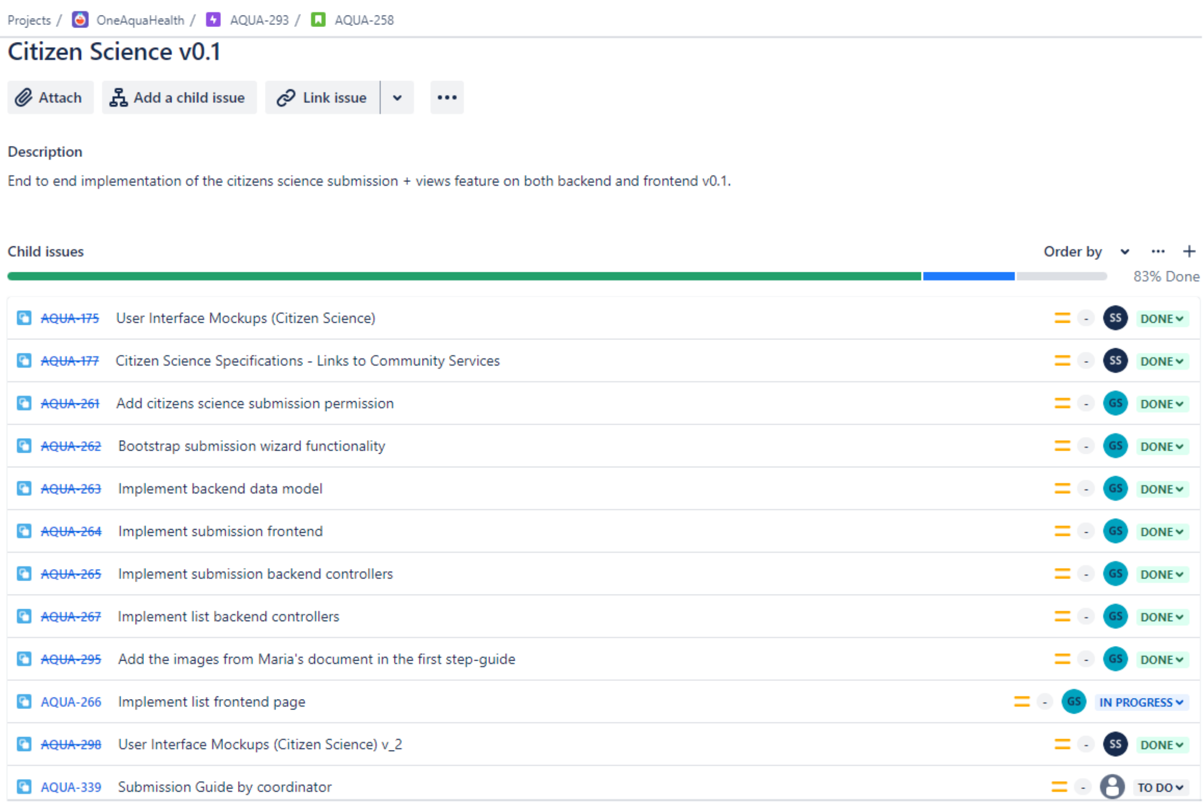


Figure 32 Scrum methodology in Jira

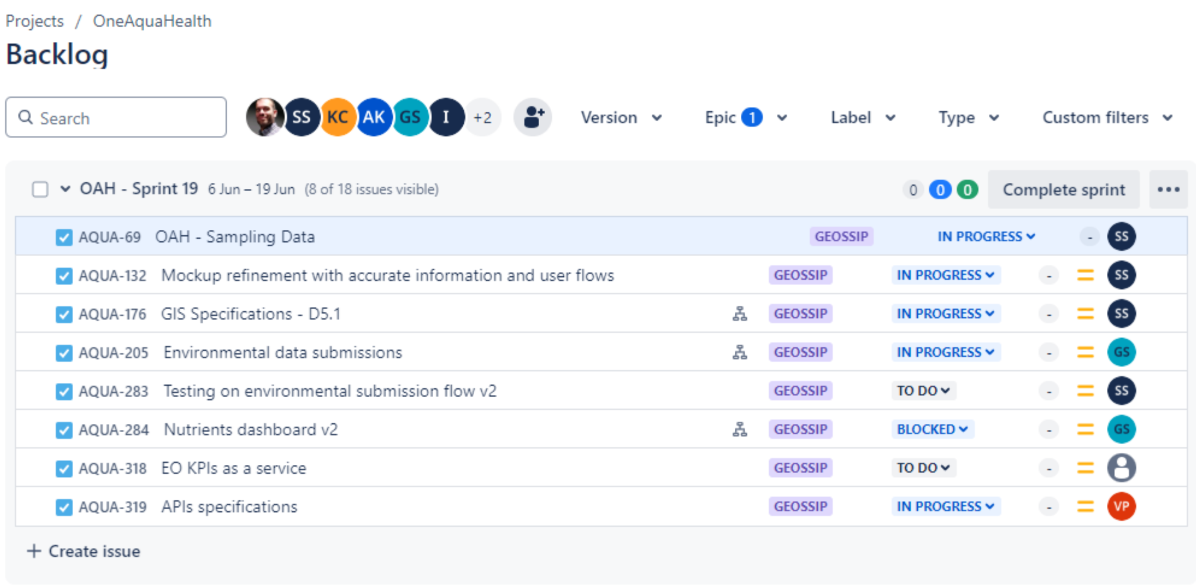


Figure 33 Jira tickets for the implementation

This approach provided a clear roadmap for developers, facilitated tracking and managing progress throughout the development process, and followed the AENEA methodology proposed by OAH partners [1]. Focusing on user stories and systematically addressing the associated tickets ensured that the final deliverables effectively met user needs and enhanced their engagement with the OneAquaHealth project.

6 Next Steps

Moving forward, the OAH partners will focus on several key enhancements and integrations to further advance the capabilities of the digital app and back-office systems. The primary tasks include the integration of APIs for satellite data processing and index calculation, which will significantly enhance the data collection and analysis capabilities. Additionally, the administration of research sites will be streamlined to improve operational efficiency. A user authentication API will be developed for seamless integration with the OneAquaHealth Information Hub, ensuring secure and efficient user management.

Enhancing the user experience is a priority, and efforts will be directed towards improving the user interface to make it more intuitive and engaging. Annotation taxonomies will be validated to ensure accuracy and reliability in data classification. Administration screens for managing data taxonomies and filtering screens for detailed data submissions will be developed to facilitate better data management and analysis.

The public and expert dashboards will be refined to cater to the specific needs of scientists, policymakers, and citizens, providing tailored insights and functionalities. A submission guide will be created to assist citizens in data submission, including an option to decline participation, ensuring user autonomy and consent.

Lastly, the API specifications in JSON format will be finalised in collaboration with SYNYO, ensuring standardisation and interoperability across the system.

All these new improvements and any related software updates in the OAH Data Backoffice and Citizen Science App will be described in Deliverable D5.2 - Geospatial and satellite information platform vs2 in Month 24.

These steps will collectively ensure that the software evolves to meet user needs and project goals, leading to a robust and user-centric OneAquaHealth Information Hub.

7 Conclusion

Deliverable D5.3 (Digital app and back-office) has incorporated and implemented the data management application, the Citizen Science App and the integration APIs that provide ecosystem health monitoring as specified by the OneAquaHealth initiative. The software demonstration is based on a rigid specification approach that has been presented in this deliverable. Through the alliance of citizen science and sophisticated information-gathering and processing methods, this deliverable documents the development achievements but also sets up the base for upcoming development in WP5 and WP6. These improvements will facilitate better data management, user engagement, and operational efficiency, ultimately contributing to the overarching goal of sustainable ecosystem health management. In conclusion, the D5.3 deliverable encapsulates the current state of technological development and lays a foundation for ongoing improvements. By addressing the outlined next steps, the OneAquaHealth project is well-positioned to achieve its mission of integrating advanced technological solutions with citizen science, fostering a participatory approach to ecosystem health monitoring. Improvements and related software updates in the OAH Data Backoffice and Citizen Science App will be detailed in Deliverable D5.2 - Geospatial and Satellite Information Platform v2, due in Month 24.

References

[1] Koutalieris, G., Symeonidis, S., Kapsomenaki, I., Feio, M. J., Esposito, L., Benis, A., ... & Tamburis, O. (2023, October). Enhancing Urban Environmental Sustainability Through Unified Stakeholders Needs Co-Creation Process (AENEA). In 2023 IEEE International Conference on Metrology for eXtended Reality, Artificial Intelligence and Neural Engineering (MetroXRAINE) (pp. 899-904). IEEE.

[2] D4.1 "Citizen science design, process and outcomes vs1". OneAquaHealth project deliverable