# Co-designed Citizen Observatories Services for the EOS-Cloud

*H2020 programme: Research and Innovation action*

# Deliverable 4.2
# Experts portal for biodiversity data validation

**January 14th, Version 1**


Project funded by the European Commission within the Horizon 2020 Programme (2014-2020)

| | |
|---|---|
| Grant Agreement No. | 863463 |

| Type | | |
|---|---|---|
| R | Document, report excluding the periodic and final reports | |
| DEM | Demonstrator, pilot, prototype, plan designs | X |
| DEC | Websites, patents filing, press & media actions, videos, photos, etc. | |
| SOF | Software, technical diagram, etc. | |
| OTHER | Flyers, etc. | |

| Dissemination level | | |
|---|---|---|
| PU | Public, fully open. | X |
| CO | Confidential, restricted under conditions set out in Model Grant Agreement | |
| CI | Classified | |

**Revision history**

| R# | Date | Description/Reason of change | Deliverable contributors |
|---|---|---|---|
| R0.1 | 20211116 | First skeleton | Marisol Parrilla López (BINEO), Santiago Martínez de la Riva (BINEO), Robin Giles Ribera (BINEO) |
| R0.2 | 20211130 | Draft content | Marisol Parrilla López (BINEO) |
| R0.3 | 20220114 | Draft shared with Consortium partners - contributions & review | Robin Giles Ribera (BINEO), Santiago Martínez de la Riva (BINEO), Marisol Parrilla López (BINEO) |
| R1.1 | 20220120 | Internal review | Sasha Woods, Michael Dodd, Xavier Salvador |
| R1.2 | 20220125 | Updates | Robin Giles Ribera (BINEO), Santiago Martínez de la Riva (BINEO), Marisol Parrilla López (BINEO) |

**Authors**

Marisol Parrilla López, Robin Giles Ribera, Santiago Martínez de la Riva

**Citation**

*This document is public can be cited as: Cos4Cloud consortium D4.2 (2021) Parrilla, M. , Giles Ribera, R., Martínez de la Riva, S. Expert Portal for biodiversity data validation (D4.2). Project funded by the European Commission within the Horizon 2020 Programme GA #863463. Available in: http://www.cos4cloud-eosc.eu*

**License and attribution**

# Executive Summary

Based on the first deliverable D4.1 General purpose integration platform (in which we carry out the implementation of a general purpose architecture) and taking into account the feedback received from the experts in different co-design sessions, we present Cos4Bio, a service whose main mission is to create an ecosystem that experts in Biodiversity related to Citizen Science can use to carry out searches and downloads quickly and in a standardised way, generating data sets from different sources of information, such as citizen observatories.

The main objective throughout the development process in this second deliverable has been to reuse the base of the general purpose Experts Portal - thus demonstrating the validity of the work already carried out and the success of implementation - in an agile way to build a service focused on citizen science observations of biodiversity. With these foundations, we have built a service, which contains the following minimal functionalities conceptualized at the beginning of the Cos4Cloud project, and defined within the framework of the Minimum Viable Ecosystem (MVE):

- Login service
- Internationalization service allows users to change the platform into different languages.
- Observation search service:
    - By species
    - By place
- Results list
- Observation filtering service.
- Download service
- Presentation of the detail of an observation
- Feedback service.
- Identification service.

However, these requirements were merely a starting point, and throughout the process, we have taken on board the opinions of experts, received during the course of several co-design workshops, carried out in order to create a service adapted to user needs. In this way, following user requirements, we have included a set of new functionalities, adaptations and modifications that make the Portal of Experts for Biodiversity a service currently available in the EOSC Marketplace. https://marketplace.eosc-portal.eu/services/cos4bio.

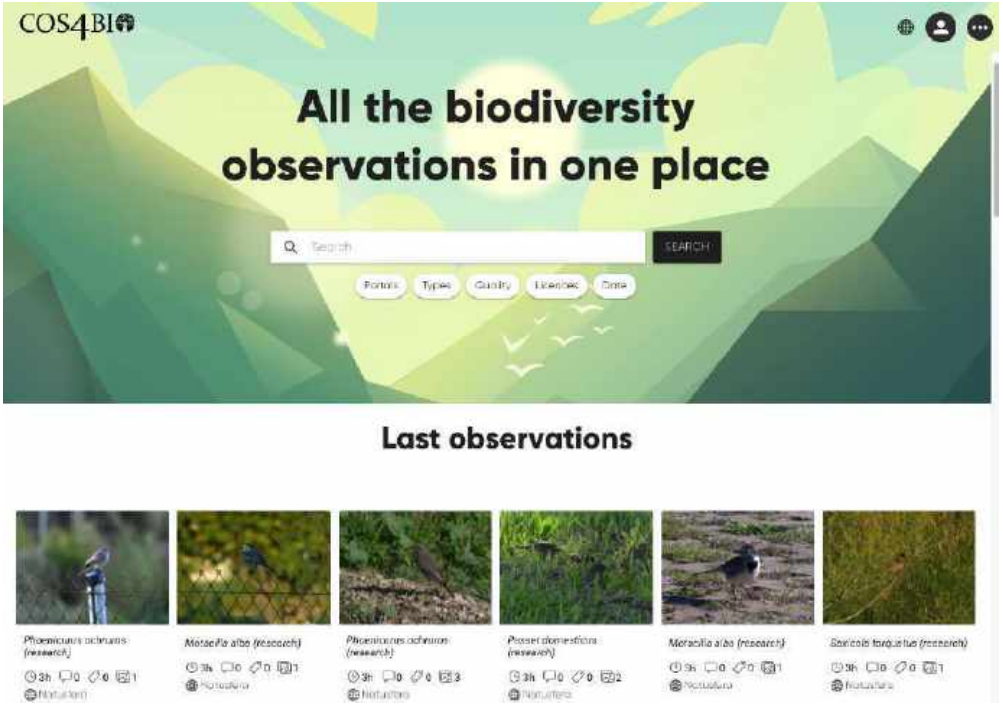
Additionally, throughout this process, we have developed our own interoperability layer, based on the Darwin Core terms standard, which is: easy to integrate by future citizen observatories, close to the context of biodiversity, widely used by the community, complying with standardization rules, and allowing us to create a service compatible with FAIR rules, which are essential to be able to publish our service in the EOSC.

**Conceptualization of the name**

The name 'Cos4Bio' arises from various concepts that are intrinsic to the service itself, such as integration, cooperation, community, life, biodiversity. These concepts are represented through a tree.

COS4BIO

A plant needs sunlight, carbon dioxide, minerals, and water to make its food through photosynthesis. The mixture of water with mineral salts is called raw sap. In this line, we could establish the parallelism in which the raw sap represents all the observations made by the citizens. The water and minerals flow to the trunk that represents the interoperability layer thanks to the different citizen observatories sharing their observations representing the roots of the plant that allow getting all of this information to mix them. In order for the plant ( or Cos4Bio service) to make its food, the raw sap is transported down the stem to the leaves. Upon reaching the leaves (the experts), the raw sap mixes with the carbon dioxide that the leaves take in from the air (that represent the identifications and comments contributed for each expert that helps to Cos4Bio identifies the scientific name of each observation and create this new community in continuous retro-feeding), becoming finally elaborated sap, which is the food of the plants.



# Index

## List of figures and tables

# 1. Methodological approaches

During this second deliverable and to implement Cos4Bio, we have followed the same procedures that we carried out for the implementation of the General Purpose Platform, which we detail in point 2. Analysis.

However, we have made an additional effort to take requirements from the experts through the co-design meetings. The objective from the beginning has been to make a platform for biodiversity experts tailored to their needs. For this reason, more meetings have been held in which new procedures have been used that allow experts to provide information in an easy and more precise way, and developers to better capture the ideas provided for their subsequent implementation.

Following this new process and always respecting the Agile Methodology, we generated a Decision Backlog table that would allow us to list and analyze all the suggestions, study their feasibility, prioritize them, and include them in the planning.

Many of the suggestions could imply significant changes in the initial architecture that was raised in the General purpose platform, so each of them was treated as a User Story in an independent sprint, which allowed us to see the implications that incorporating them could have.

For example, users could have a personal space, and view statistical data regarding the contributions they made. This user story was a complete change of the model since now we would have to store user information, their identifications, comments, download reasons, number of downloads, etc.

For this reason, once again, it was essential to rely on a continuous development system, as we did for the General Purpose Platform, which would allow us to carry out the implementation of Cos4Bio, which can be accessed from the following link: https://cos4bio.eu

# 2. Analysis

## 2.1 Introduction

The analysis process carried out for the implementation of Cos4Bio, or Portal of Experts for Biodiversity, has been based on the same methodology used during the development of the general purpose Platform called Design Thinking:

"Design Thinking: Empathize, define, devise, prototype and test".

This methodology is divided into four phases that we have been developing throughout this block, adapted to the specific needs of the expert portal service for Biodiversity - Cos4Bio:

- Empathize.
- Define.
- Ideate.
- Prototype.
- Test.

But before we can carry out each of the phases, we carry out detailed work in the co-design sessions to collect the suggestions of the experts who were not present in the initial architecture. Based on the list of suggestions, we were able to implement a Backlog decision table and on the whole, carry out the Design Thinking process now applied to the new Cos4Bio service.

Below we refer to the meetings held during this second period and the suggested inputs to comment on each of the stages.

## 2.2 Co-design inputs

**Co-design activities implemented for Cos4Bio**

- Sci&Tech meetings: bimonthly since April 2020.
- Co-Design workshop (March 2021): experts and citizen scientists.
- Co-Design meeting during ECSA presentation (May 18th 2021): experts and citizen sciences.
- General meeting (June 2021): interoperability integration.
- Annual meeting (November 2021): key performance index for services and EOSC publication.

Thanks to the feedback received, we made a new list of requirements as a Backlog table (Table 1) in which we specified the following columns:

- Id: Identifier of the suggestion.
- User 's feedback: suggested by an expert.
- Decision: Internal decision of whether or not to carry out the task, followed by the explanation column.
- Explanation: decision field justification.
- Priority: Low / Medium / High / Very High.
- Status: Done / Working on it / Discarded.

*Table 1. Backlog table*

| | Id | User's feedback | Decision (yes/no) | Explanation (reason why we could do it or not) | Priority | Status |
|---|---|---|---|---|---|---|
| | 1 | Statistics service | Yes | That service could be interesting to show the usage of the service by each expert and analyse the contribution of all of them. | Very High | Done |
| | 2 | Include a form with the reason for downloading information. | Yes | to understand why experts find the data useful | Medium | Done |
| | 3 | Include searches by place. | Yes | We checked in the co-design session that it can be searched by species. They ask us to also search by place. | High | Done |
| | 4 | New registration system. | Yes | The experts want to see the information associated with your contribution, so we have to create a profile for the experts. | Medium | Done |
| | 5 | Download history. | Yes | Let us show the experts all the datasets | Medium | Done |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | downloads that they did. | | |
| | 6 | Search history. | Yes | Help the experts if they want to repeat previous searches. | Medium | Done |
| | 7 | Include terms of use. | Yes | GDPR | Medium | Done |
| | 8 | Include privacy policy. | Yes | GDPR | Medium | Done |
| | 9 | Notice if you want to receive notifications. | Yes | GDPR | Medium | Done |
| | 10 | Frequently asked questions that explain the use of the service. | Yes | Support documentation | Medium | Done |
| | 11 | Feedback form. | | To improve Cos4Bio service. | High | Done |
| | 12 | Dashboard with service statistics. | Yes | General dashboard, interesting to demonstrate how this service is useful | Medium | Done |
| | 13 | Give the option to the expert to share his profile on Cos4Bio. | Yes | Visibility | Medium | Done |
| | 14 | Show results in table mode. | No | Presenting the results in this format is a design that has become obsolete in terms of usability.<br>In addition to this, it does not affect the execution of the service so we discard it. | Low | Discarded |
| | 15 | Filters by date range | Yes | To create more advanced filters that allow you to filter your observations by date range | High | Done |
| | 16 | About page | Yes | A page that provides more information about the project | High | Done |

| | | | | | | |
|---|---|---|---|---|---|---|
| ● | 17 | More complex data structure | Yes | Interoperability and Engagement | Very High | Done |

## 2.3 Stages

### 2.3.1 Empathize

The conclusion of deliverable 4.1 General purpose integration platform was that we needed to carry out the implementation of an interoperability layer closer to the understanding of the Observatories, easy to integrate, based on standards already known by the community such as Darwin Core Terms, and supported by international organizations such as GBIF or TDWG. This development helps us to create an easy way to integrate for COs, which will allow more COs to join the Cos4Bio Service.

So during this second period besides to include more services and functionalities advised by the co-designers we have developed this new interoperability layer based on these 2 standards widely known and validated by the International Community. To do that we have had to implement a Mapping model based on these standards that explain in the section API DWC. Below we can see the diagram of the proposed Model.

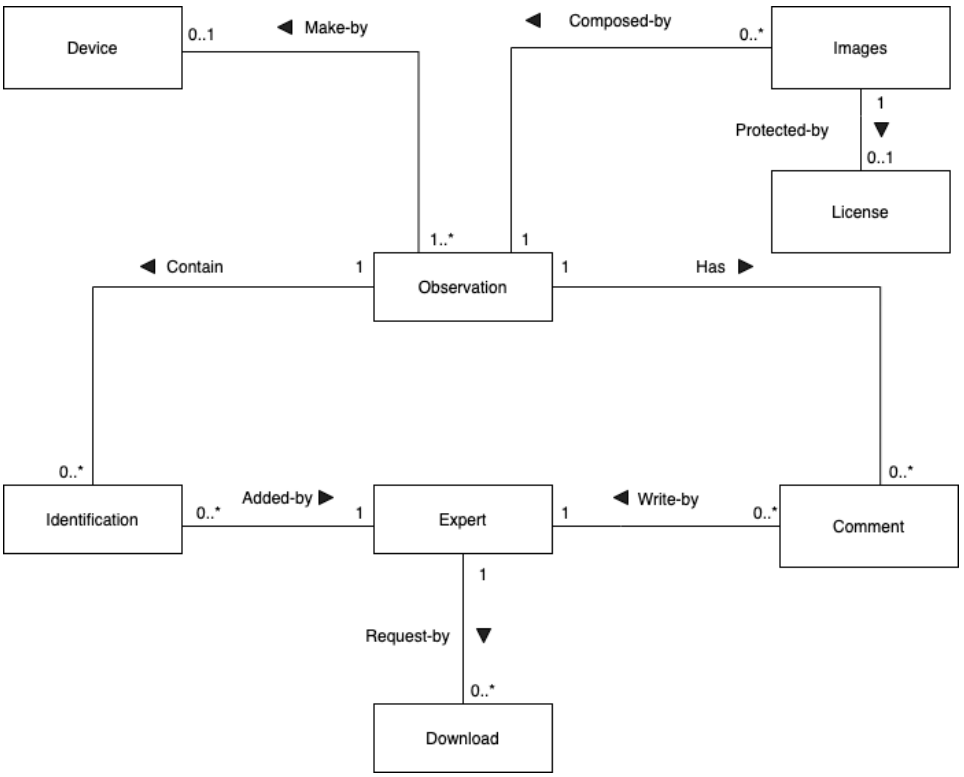



*Figure 1. Data model diagram that represents the information managed in the interoperability layer based on Darwin core terms standard*

### 2.3.2 Define

Once we finished the analysis stage, we defined with the experts who could be the potential users for Cos4Bio. Mainly they will be experts capable of identifying species, but also they will be citizens involved in citizen science, interested in learning more about biodiversity and environmental variables.

These are the user profiles that will be involved in Cos4Bio:

| **Table of users** | |
|---|---|
| **Role** | **What will they find on the platform?** |
| New user | Easy registration intuitive interface.<br>Familiar interface <study of other platforms>.<br>Simple learning process.<br>Ease to understand how the platform works. |
| Expert user | Well classified information.<br>Easy access to information.<br>Easy to see news. |

| | |
|---|---|
| New expert | Intuitive actions.<br>Similarity to known platforms.<br>Simple learning process. |
| Senior expert | Easy export of information.<br>Quick access to information.<br>Well classified and filtered information. |

*Table 2. Table of users*

On the other hand, we defined the typical user of our platform, which could be a person with the following profile:

- Approximate age between 25 and 65 years.
- Interested in citizen science, biodiversity, environmental variables and sustainability.
- And with advanced knowledge about the identification and validation of biodiversity observations and environmental measurements.

### 2.3.3 Ideate

In this stage, we begin to shape the user flows and sitemap of the platform represented with several diagrams that help us better understand the objectives, functionalities and flows defined during the different technical meetings before going to the next stage.

- Flow Diagram
- Task Flow Diagram
- UI Flow Diagram
- Site Map Diagram
- Wireframes

All of these diagrams also helped us to identify improvements in information flows, detect useless functionalities and make sure that we take all the relevant aspects of the service into account.

### 2.3.3.1 Flow Diagram

The objective of the flowchart is to represent the actions that users will be able to take to complete the different processes covered by the Cos4Bio platform.

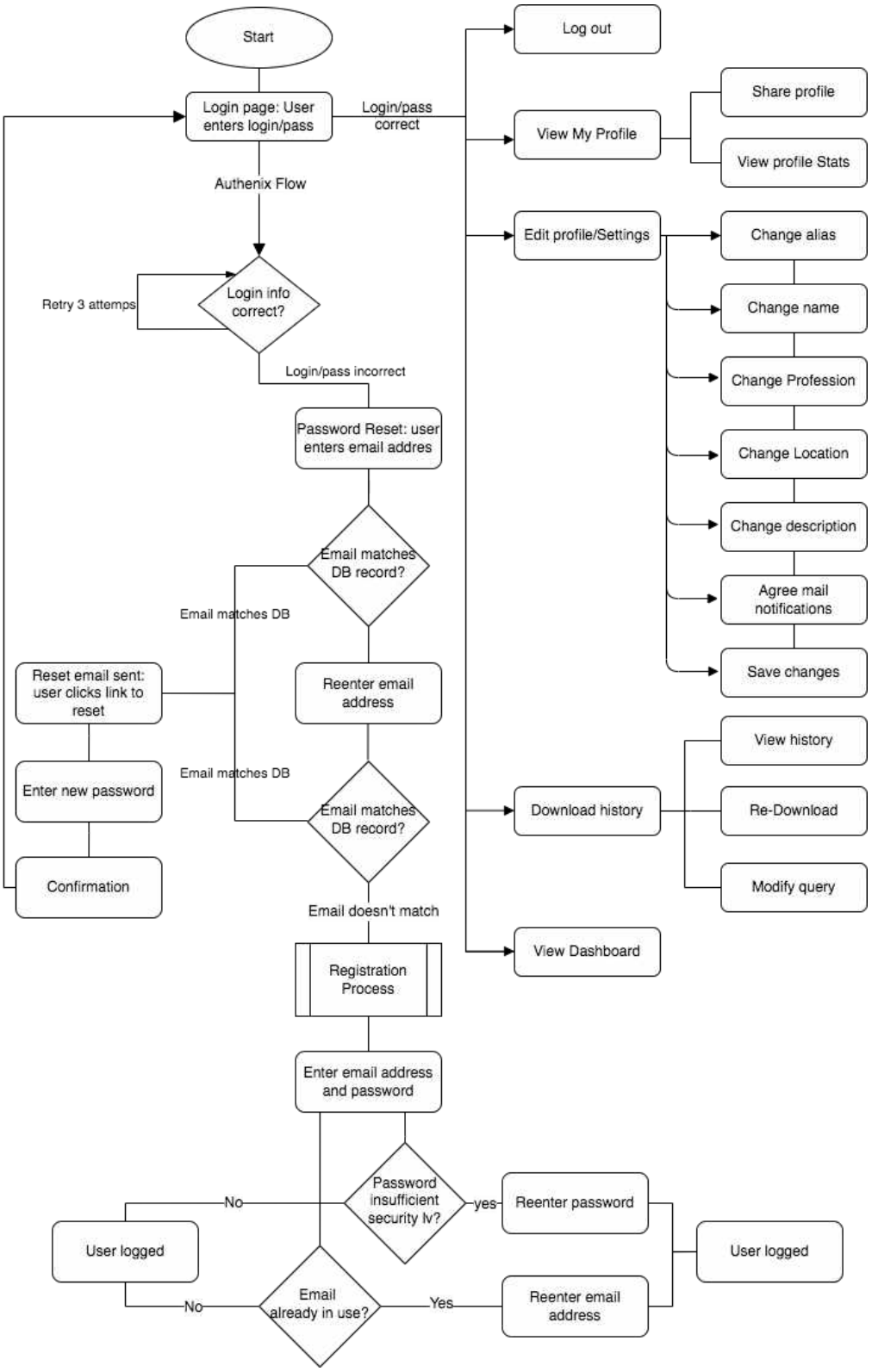


*Figure 2. Flow Diagram representing the actions that users will be able to take to complete the different processes covered by the Cos4Bio platform*

In this first flowchart, unlike the flowchart of the general purpose platform implemented in the first deliverable, we see that there is a space reserved for each expert that allows them to provide, if they deem it appropriate, their personal information: aliases, name, profession, location, description. On the Cos4Bio Terms of Service page, they can see the use that we will make of this information. Through their profile dashboard they can also see the contributions they have made in the form of statistics and can even share their profile with the scientific community, valuing their help and knowledge.

In addition, so that you can exhaustively track your queries and downloads, we have included a search cache system and a history of the downloads made by each user, in which you can see the search parameters and filters that have been used. This way experts can re-execute these queries and even modify these parameters to repeat them in, for example, another date range. In this way, Cos4Bio can reduce experts' consultation times, improve their research, increase the contribution of identifications and increase the visibility of the work they have carried out.

Once again, we rely on another of the Cos4Cloud services, Authenix, integrated into the base of the general purpose platform and used as the foundation of Cos4Bio so users can use a federated and secure authentication system.

It is important to mention two aspects:

1. These changes, as we have seen in the Co-Design section, were promoted thanks to the feedback provided by the experts in each workshop.
2. The use of agile methodologies and having an architecture based on the development of the General Purpose Platform has greatly helped us to reduce implementation times and demonstrate the value of the work done during the first deliverable.

In addition to these new aspects, we have improved the search processes, the filtering system, and the information provided in Cos4Bio, as can be seen in the second flow diagram:

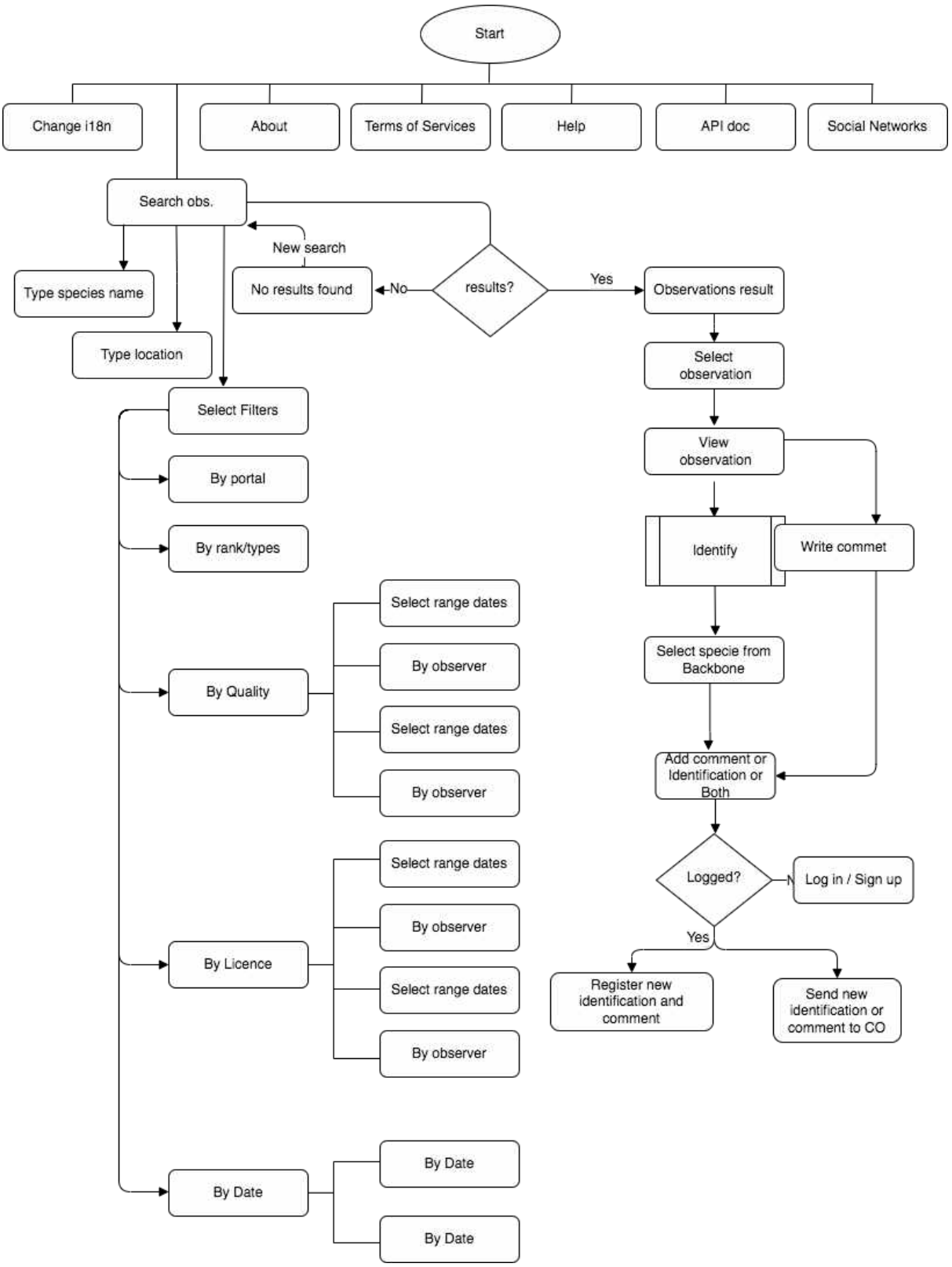



*Figure 3. Flow Diagram demonstrating the search processes, the filtering system, and the information provided in Cos4Bio*

The search process at the species and location level can be used independently or in combination and this service can be supported by the improved filtering system with respect to the base implementation of the general purpose platform, focusing on useful filters for the time to look for biodiversity information such as:

- Filters by portal.
- Filters by species range.
- Filters by information quality.
- Filters by licence.
- And filters by date or date range.

Again some of these new features came to light thanks to input from experts. These contributions also allowed us to delve into the value of filter types:

For the observation quality filters, we wanted to be able to filter by observations that have photos, with geolocation information, that have not yet been identified or that already have their scientific name, and for the licence type filters we have based on the type of licences used by GBIF: CC0, CC By, CC By-NC y CC By-SA.

We have also incorporated the internationalization system that allows modifying the language of the platform in seven languages, which we will present in more detail in the internationalization section corresponding to task 4.2: Multilingual support.

#### 2.3.3.2 Task Flow Diagram

The Task flows are focused on how users travel through the platform while performing a specific task. They generally show only one path and don't include multiple branches or pathways like a traditional user flow might. These are best used when the task being analyzed is accomplished similarly by all users. When using task flows, it is assumed that all users will share a common starting point and have no variability in the way the task is carried out.

Each of the Task Flows is presented below, taking into account the changes made and the new functionalities implemented for the Cos4Bio service.

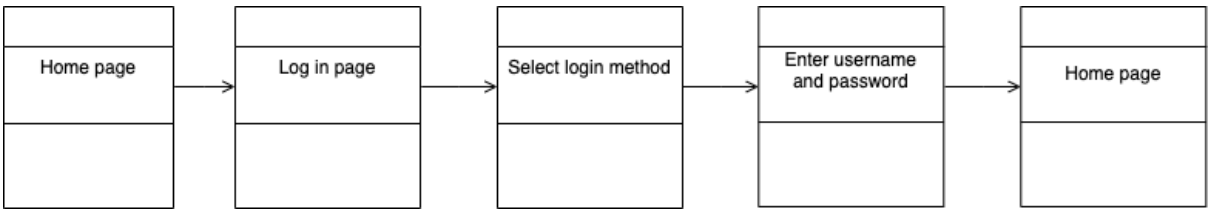



*Figure 4. Login. Task Flow.*

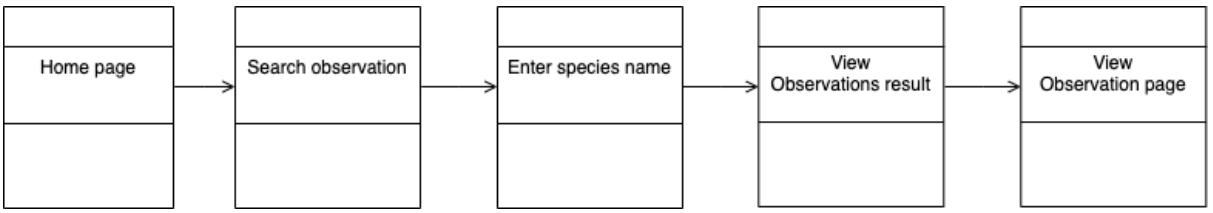



*Figure 5. Search by species name. Task Flow.*

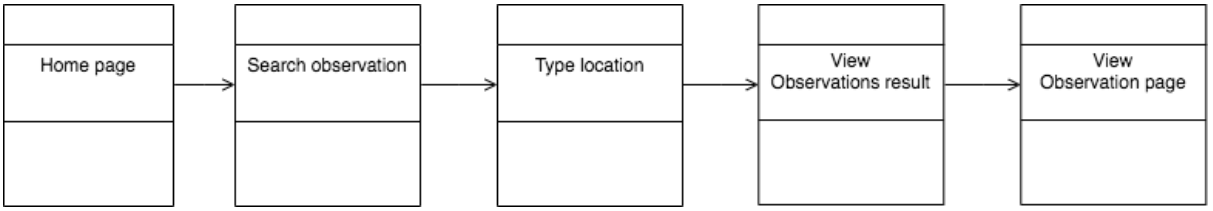



*Figure 6. Search by location. Task Flow*

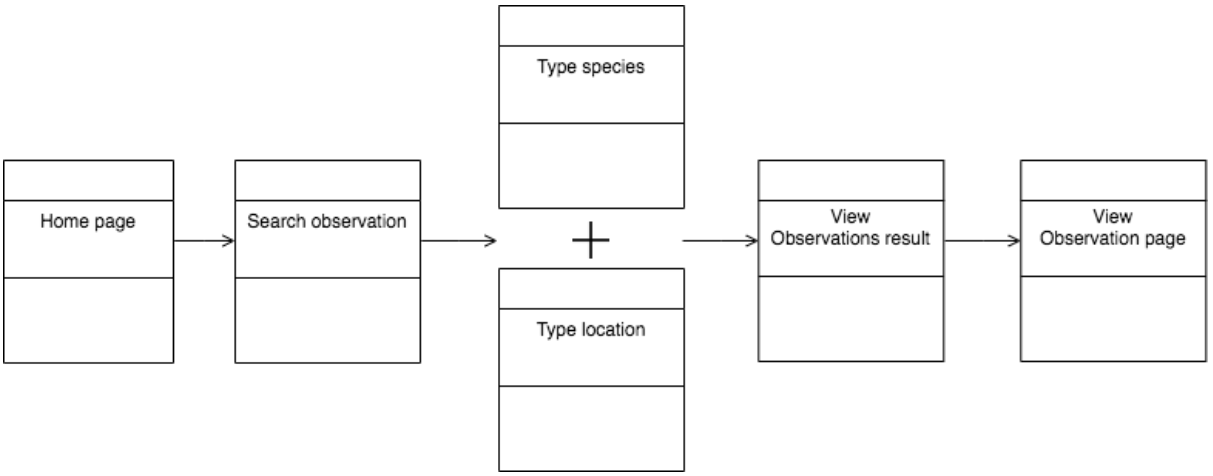



*Figure 7. Search by species name combined with location. Task Flow.*

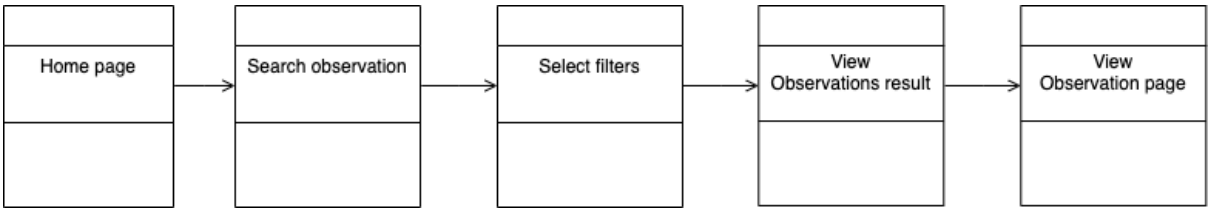



*Figure 8. Search using filters. Task Flow.*

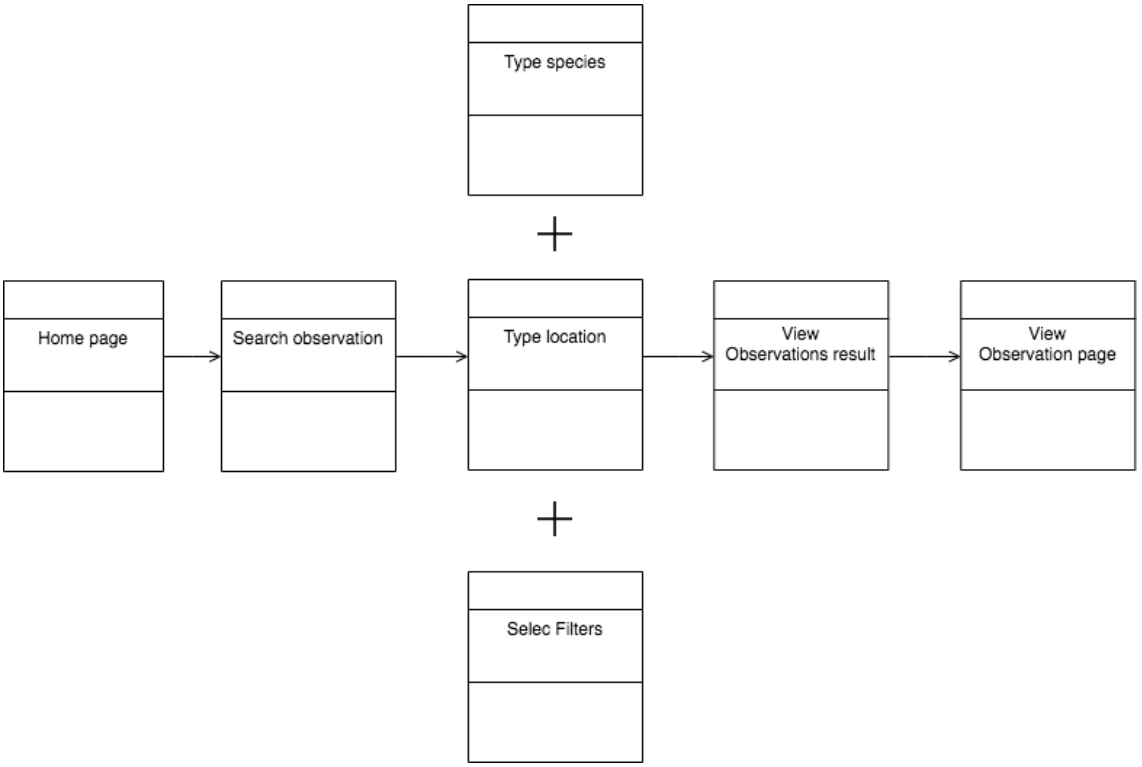



*Figure 9. Search by species name, location and filters. Task Flow.*

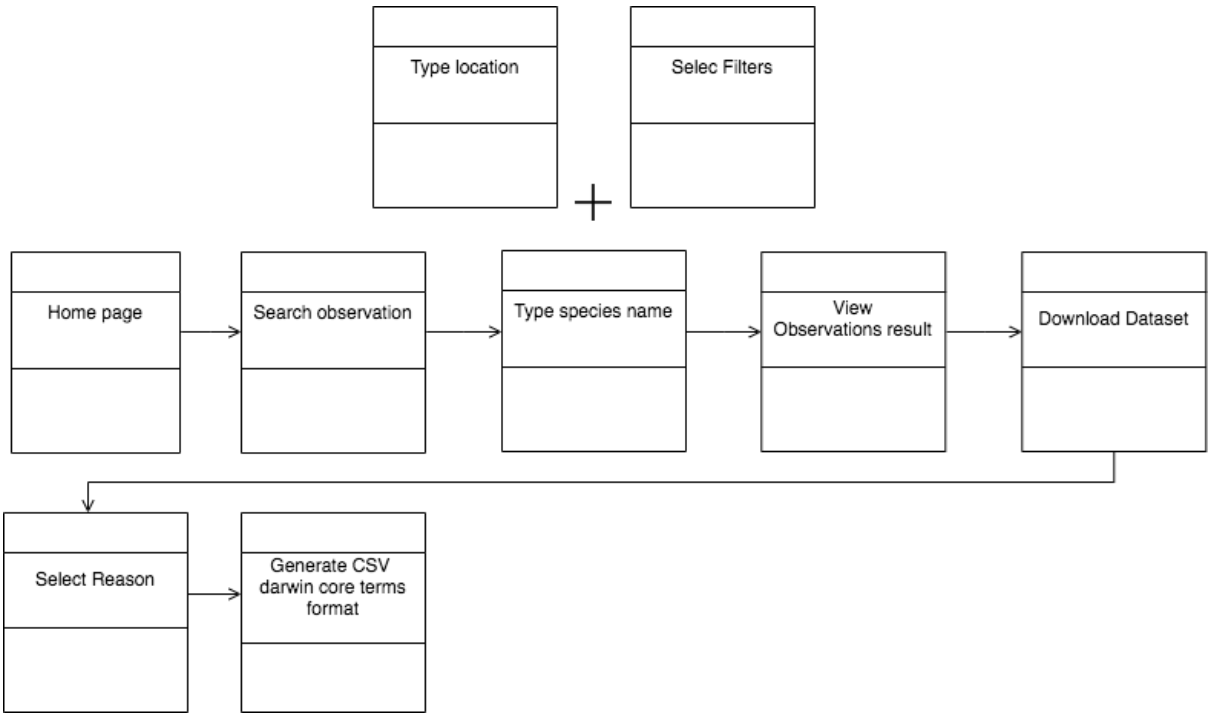



*Figure 10. Download dataset searching by species name, location and applying filters. Task Flow.*

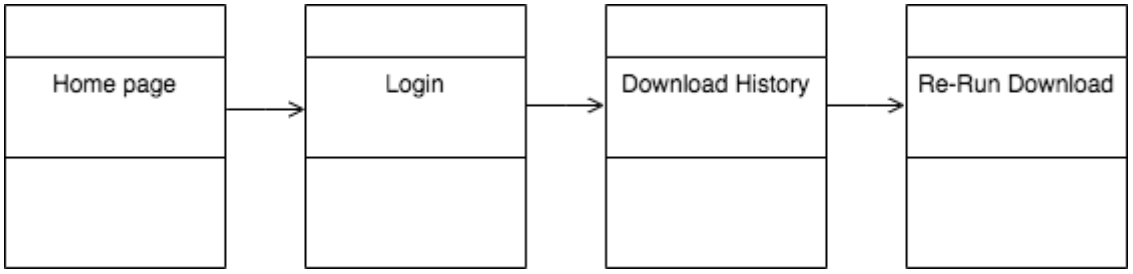



*Figure 11. Download dataset searching by species name, location and applying filters. Task Flow.*

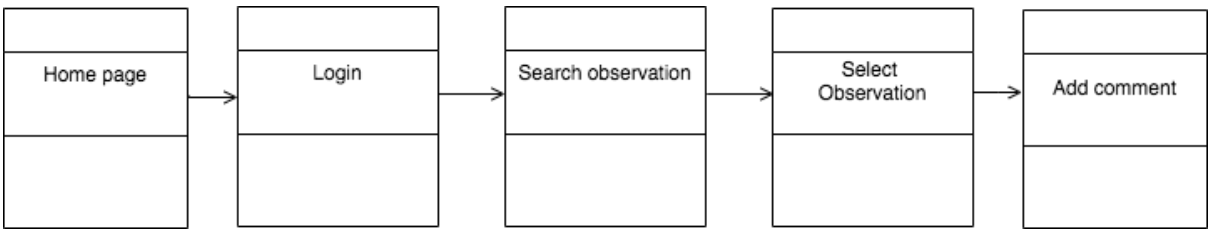



*Figure 12. Add a comment. Task Flow.*

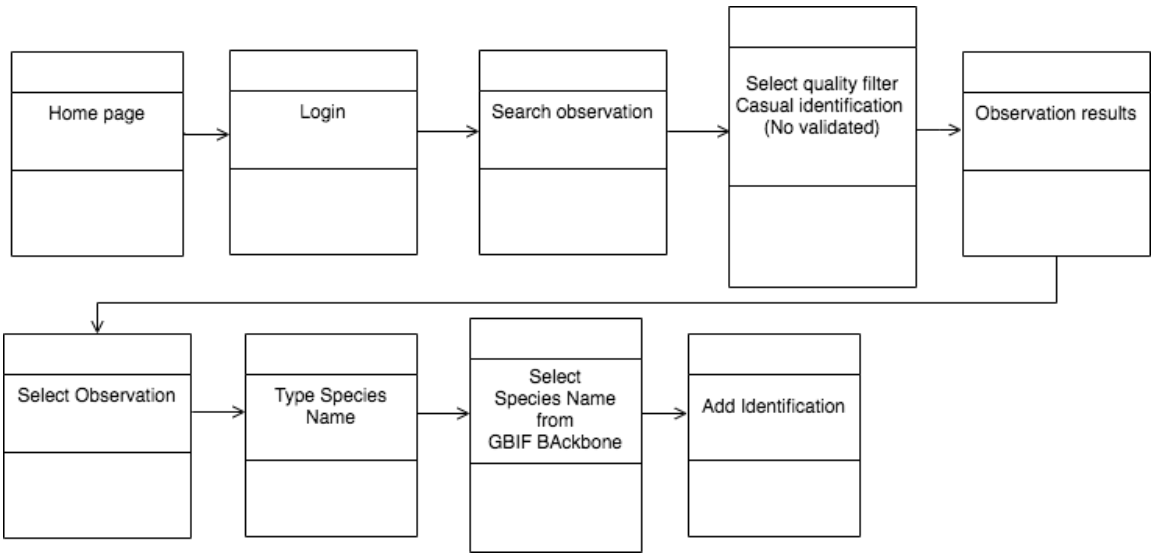



*Figure 13. Add an Identification. Task Flow.*

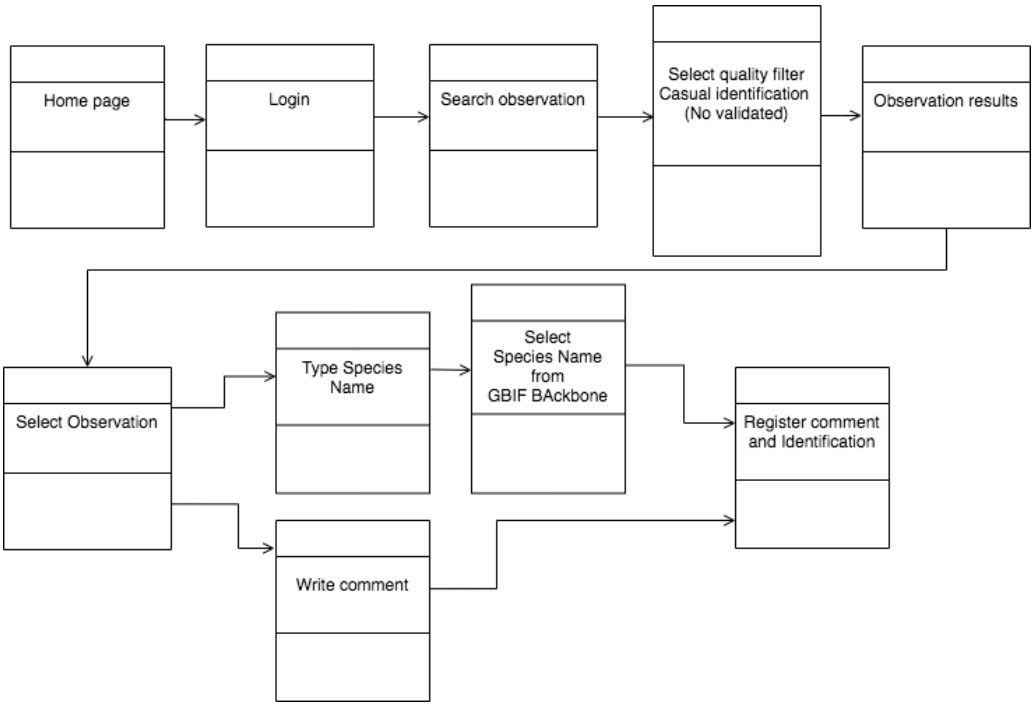



*Figure 14. Add identification and comment. Task Flow.*

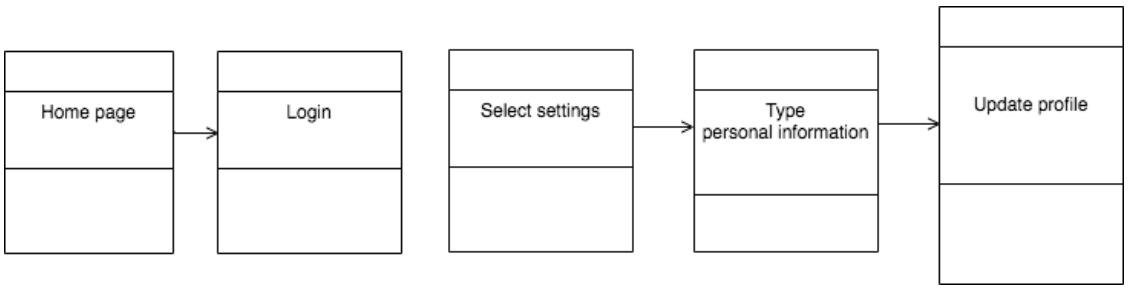



*Figure 15. Update profile. Task Flow.*

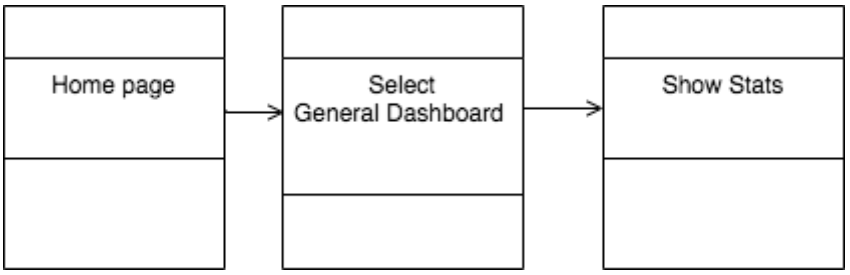



*Figure 16.Show general dashboard. Task Flow.*

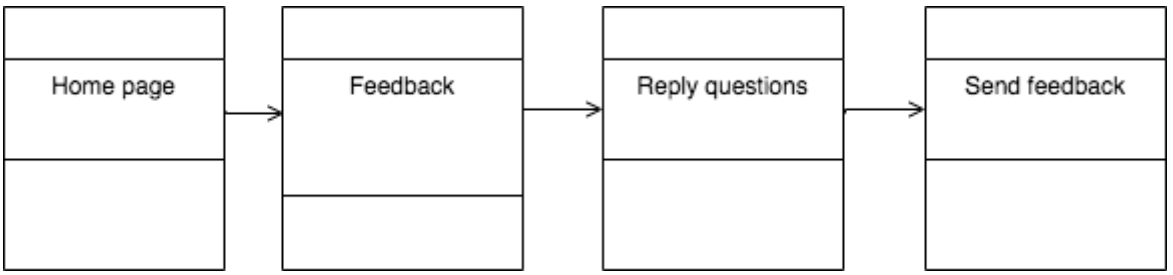



*Figure 17. Send feedback. Task Flow.*

#### 2.3.3.3 UI Flow Diagram

User interface flow diagrams are typically used for two purposes. First, they are used to model the interactions that users have with the software, as defined in a single-use case. For example, it can refer to several screens and provide an insight into how they are used. Based on this information, we are going to develop a user interface-flow diagram that reflects the behavioural view of the single-use case. Second, they enable us to gain a high-level overview of the user interface for the Expert Portal.

This overview is effectively the combination of all the behavioural views derived from our use cases, the result is called the architectural view of the interface.

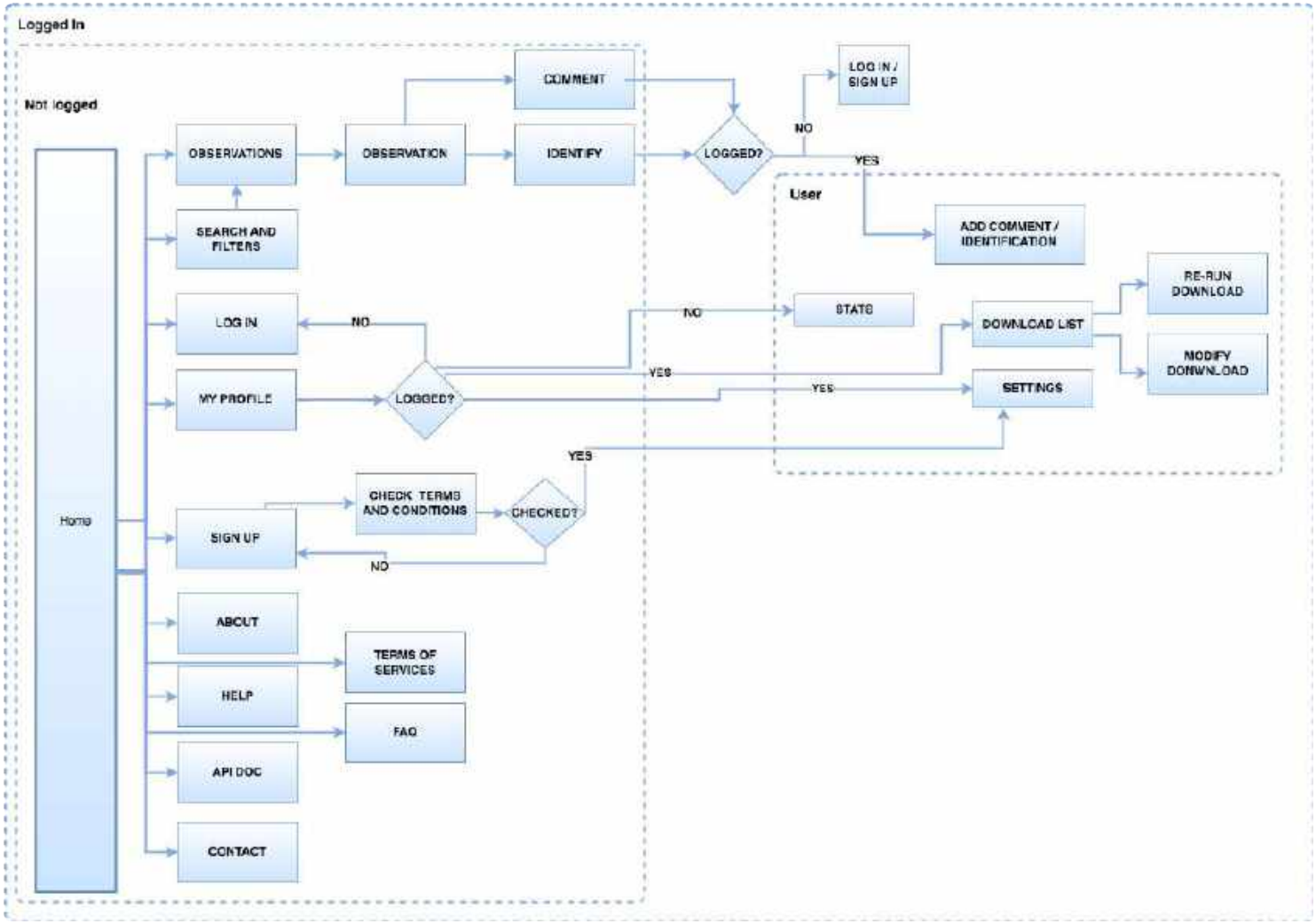



*Figure 18. UI Flow diagram.*

### 2.3.3.4 Site Map Diagram

A sitemap is a list or diagram that helps you plan out your website. It should contain all the pages of a website in a way that shows how the user will access them, starting with the homepage and branching out to include all the subpages.

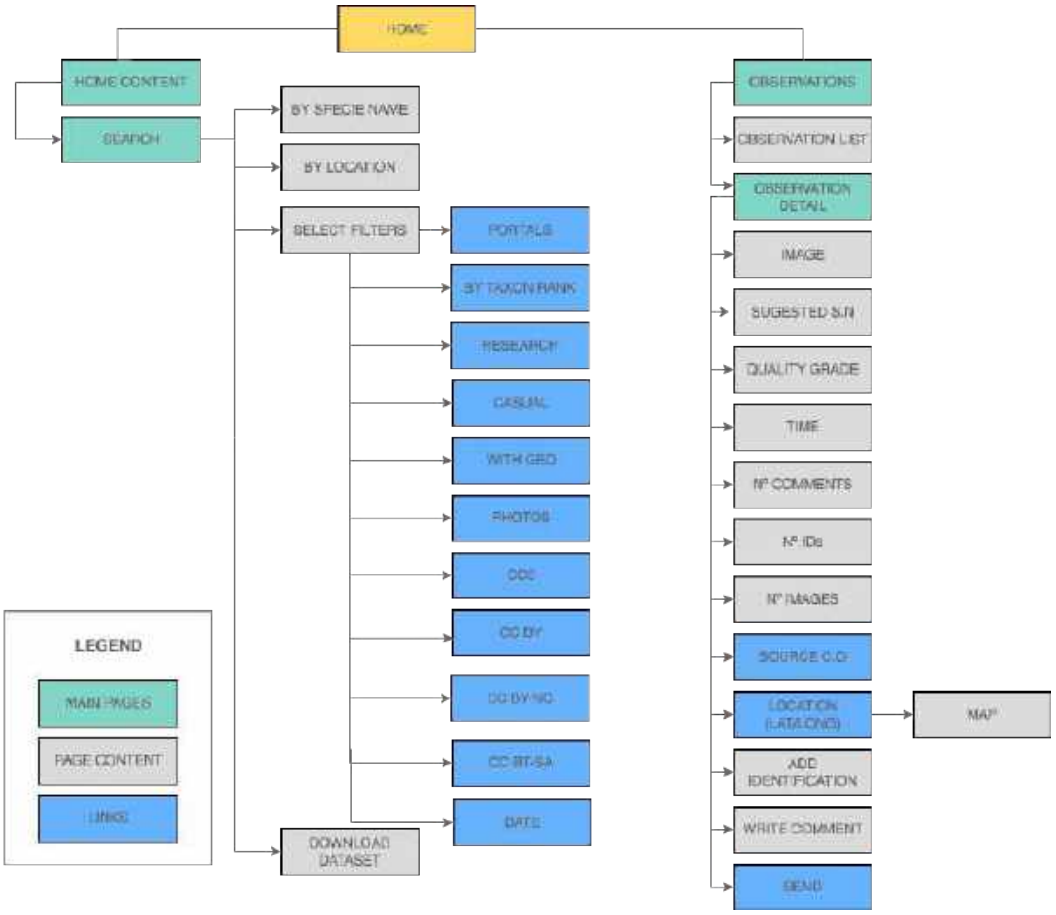



*Figure 19. Site Map 1*

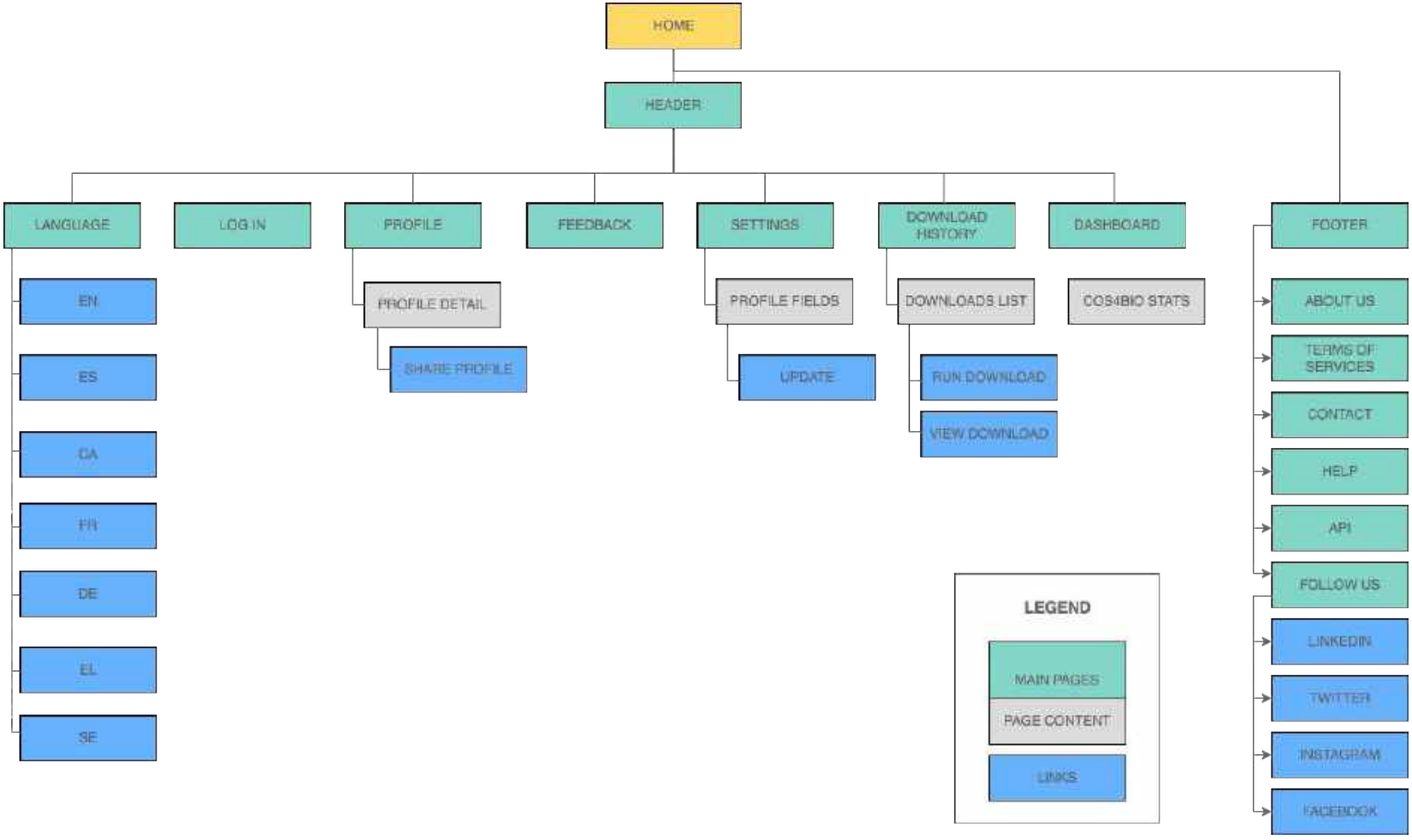



*Figure 20. Site Map 2.*

### 2.3.3.5 Mockups

Mockups are a visual guide that represents the design of a website. They include the navigation structure and design elements in detail and help to visualise ideas and concepts. Following, we can see the different mockups that compose Cos4Bio. (https://cos4bio.eu)

**Figure 21:** In this mockup, we can see the Cos4bio header section with: the logo, the language selector, the user profile icon, and the menu options. Then the cover text, followed by the search bar and the filter options. At the bottom of this section, we can see a list of the observations ordered by date.

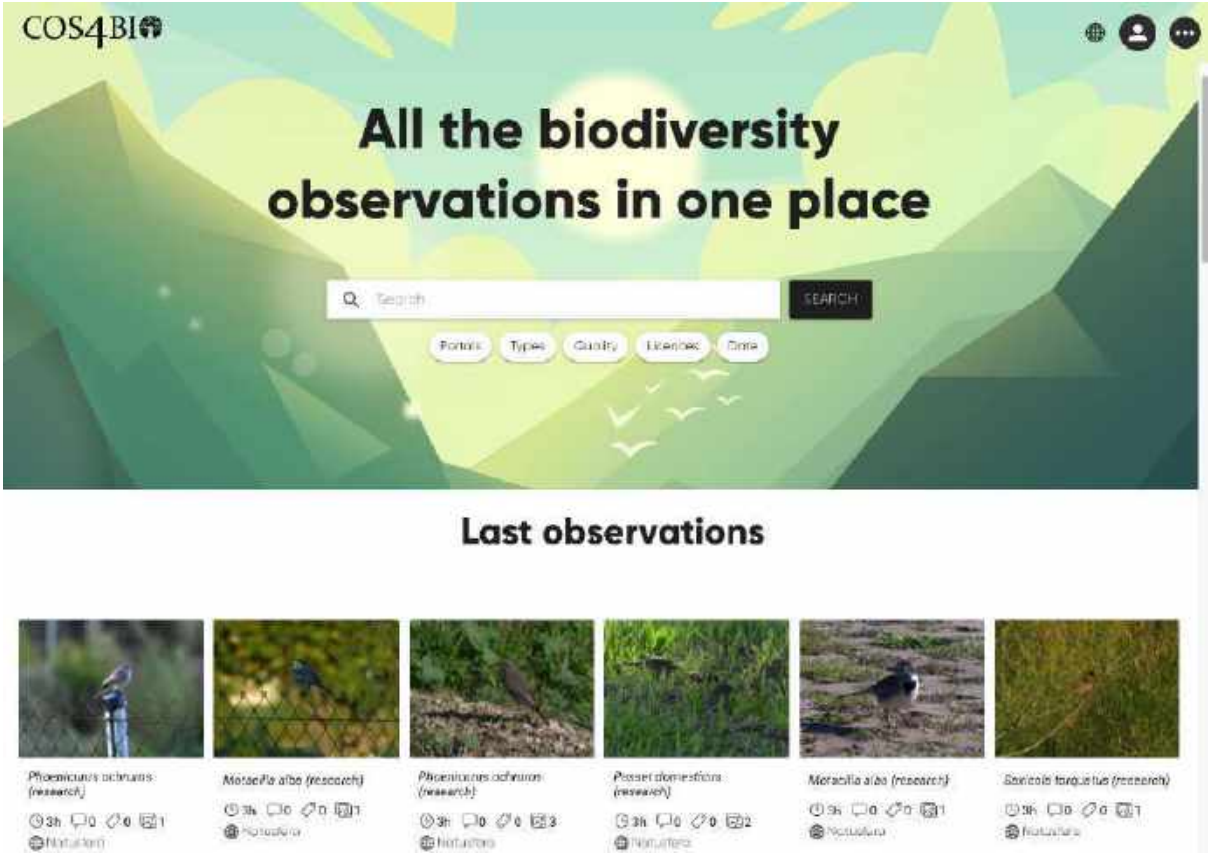



*Figure 21. Mockup. Home view with header and search/filters section.*

**Figure 22:** In figure 22, we see the footer with the following sections: About, Terms of Service, Contact, Help, API and the links to social networks. In addition to highlighting that the project is part of an H2020.

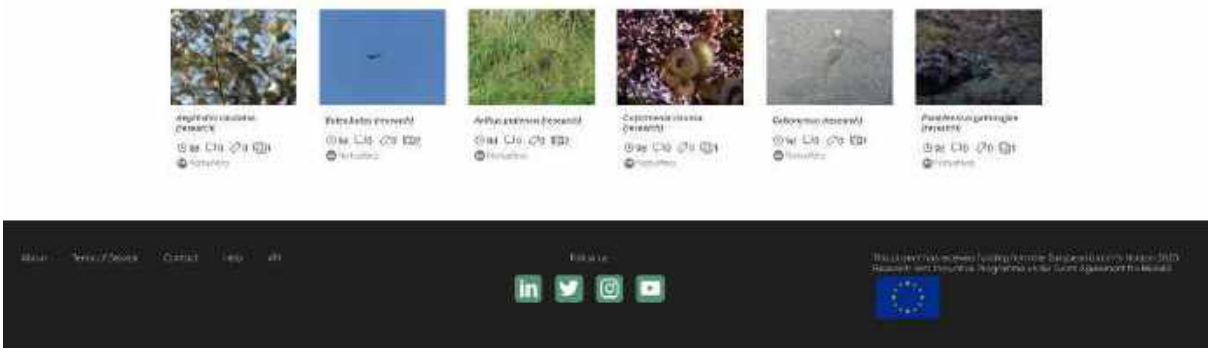


*Figure 22. Footer of the home page of Cos4Bio.*

**Figure 23:** In the upper right, we can see three icons, the language selector appears first, followed by the login button and the menu.

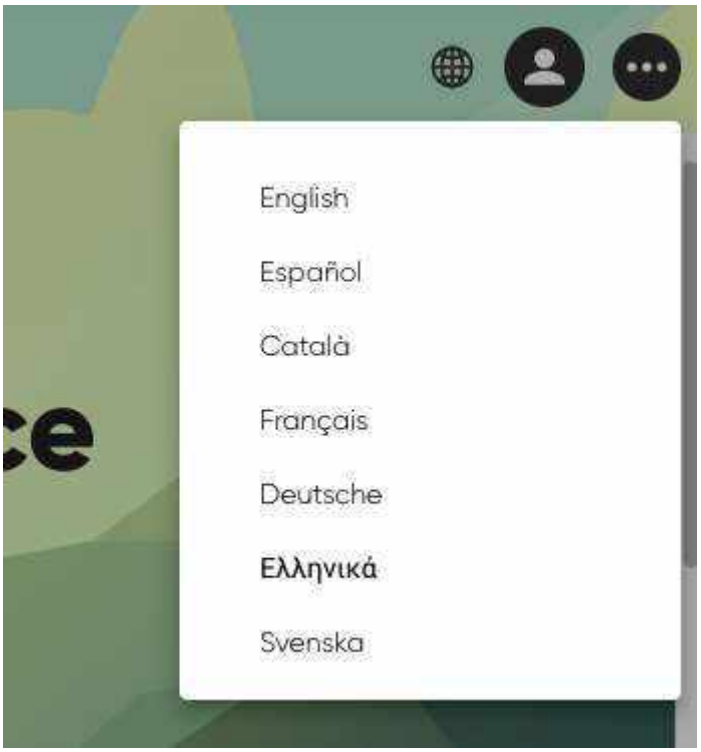



*Figure 23. Choose language options.*

**Figure 24:** By clicking the "login" button you can log in or visit the Authenix website if you need more information about the Authenix service.

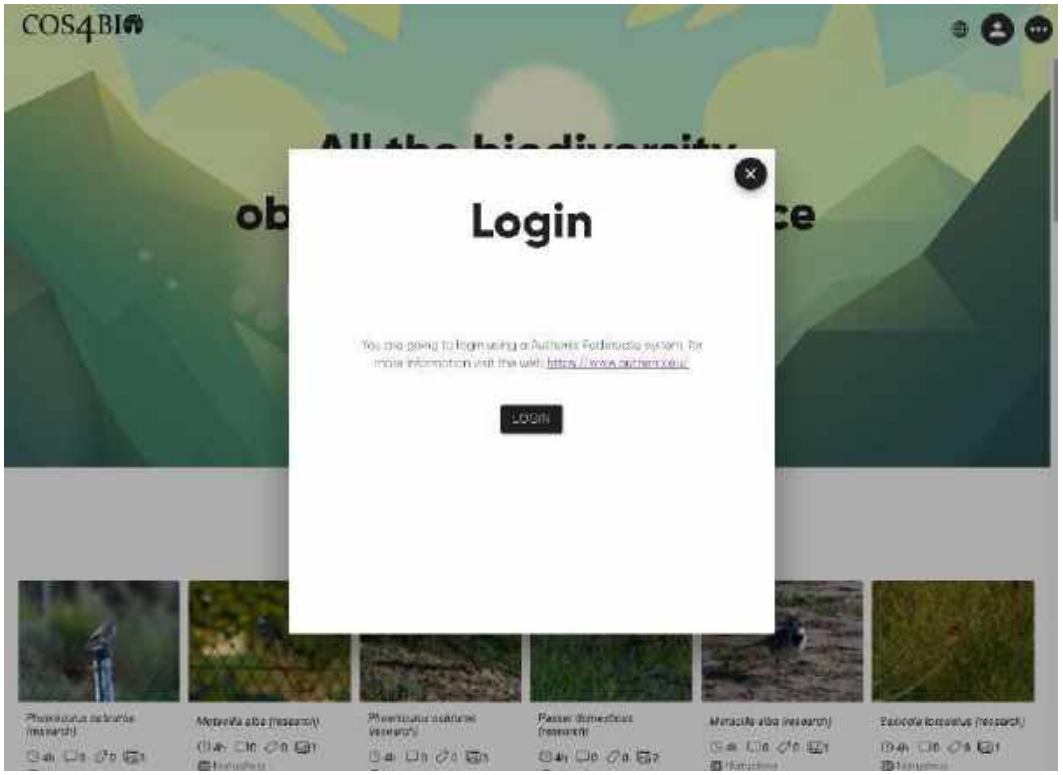



*Figure 24. Login in Cos4Bio.*

**Figure 25:** If you select login you can find your login provider in Authenix.


*Figure 25. Log in Authenix site.*

**Figure 26:** These are the menu options that we can see before logging in: “Login” by Authenix, “Feedback” of Cos4Bio, and “Dashboard” with general indicators of Cos4Bio.


*Figure 26. Menu options before login in Cos4Bio.*

**Figure 27:** Once we have logged in, the options that we can find in the menu are: “Logout”, “Feedback”, “Settings”, “Download history”, and “Dashboard”.

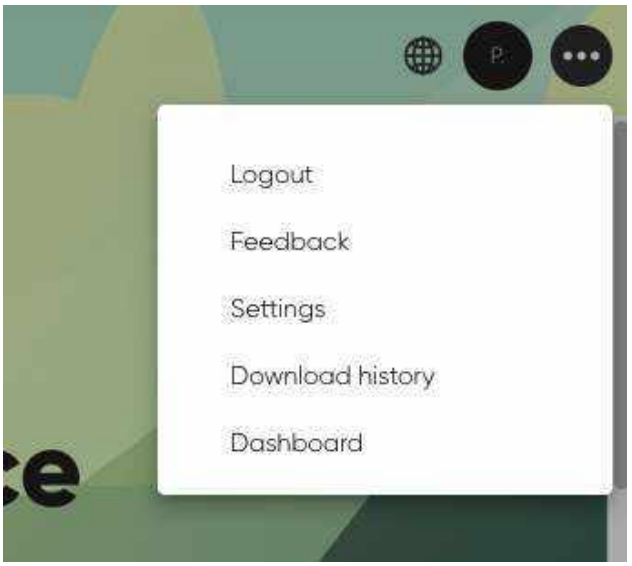



*Figure 27. Menu options when you are already logged in.*

**Figure 28:** Download history made by a user, with the possibility of repeating the same searches or reusing them by modifying any of the filters.


*Figure 28. Download history.*

**Figure 29:** In the next figure we can see the global dashboard of Cos4Bio that include several charts in which we could see the usage of the Service and helps us to demonstrate how the useful this service will be the following information: (https://cos4bio.eu/dashboard)

In the first figure we can see the number of comments and identifications made in Cos4Bio (this information also can be seen in more detail in the graph below). A total number of data downloads and the number of users who are part of Cos4Bio.


*Figure 29. Cos4Bio global dashboard with the information of count of identification and comments, number of downloads, Users, and Time series of Comments and Identifications.*

**Figure 30:** Next, In the Cos4Bio global dashboard, we can see two charts with "Comments & identifications by Portal" and "Download history by date".

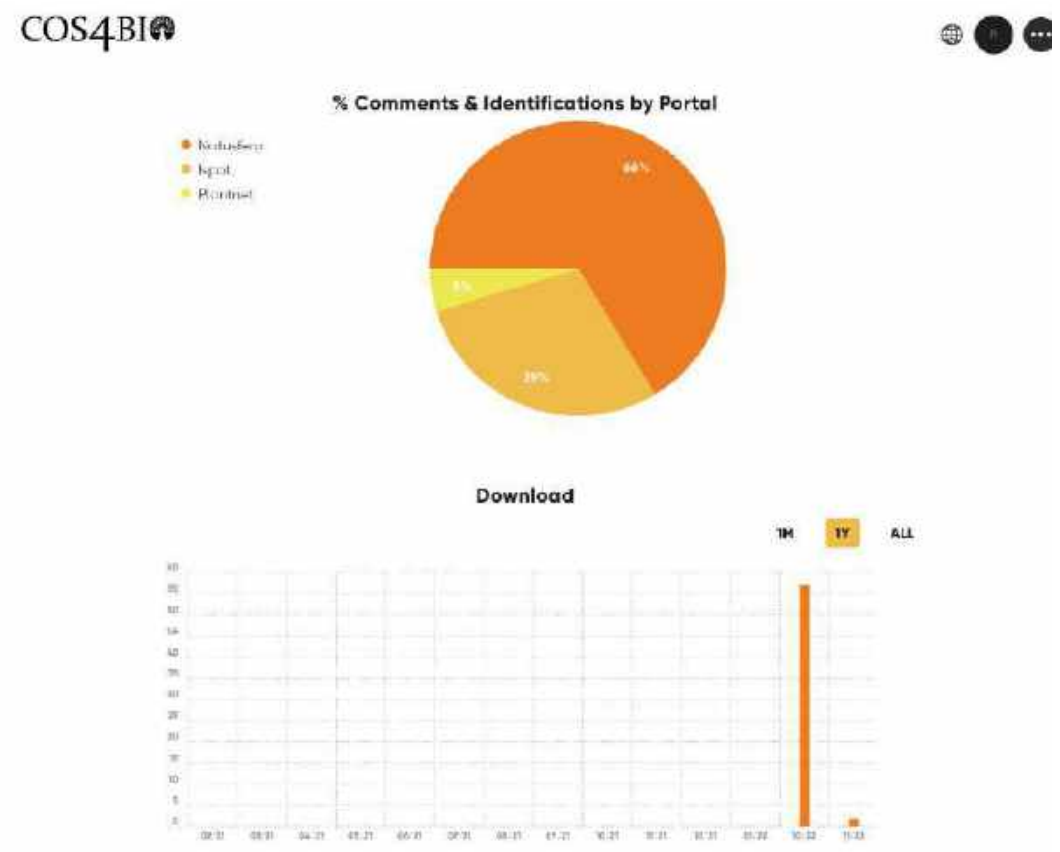



*Figure 30. Comments & identifications by Portal. // Download history by date.*

**Figure 31:** And finally,, also we can see a chart with the professional profile of the users that are part of Cos4Bio. This information will help us to better meet our audience and create in the future new marketing campaigns, more focused on the people that are more interested in the Cos4Bio service, and maybe create new ones to reach new audiences.

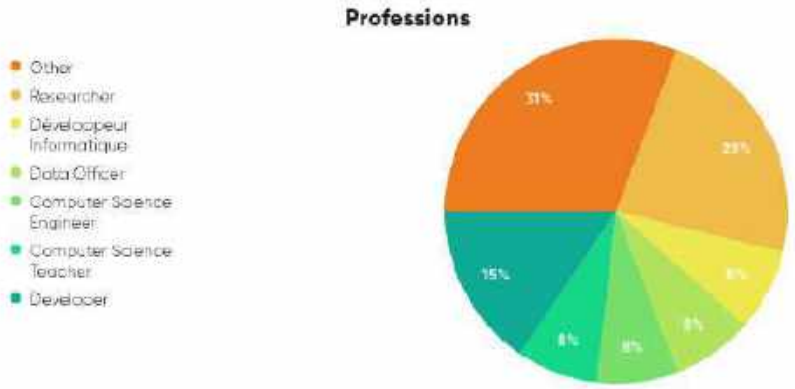



*Figure 31. Professional profile of the users that are part of Cos4Bio.*

**Figure 32:** If the experts want to see their profile page, they just click on the user profile icon, which we show in the following image. On this page each user can read the public information of each expert such as: the profile image and user's description, followed by global counts such as total number of identifications and comments and total number of downloads done by the expert..

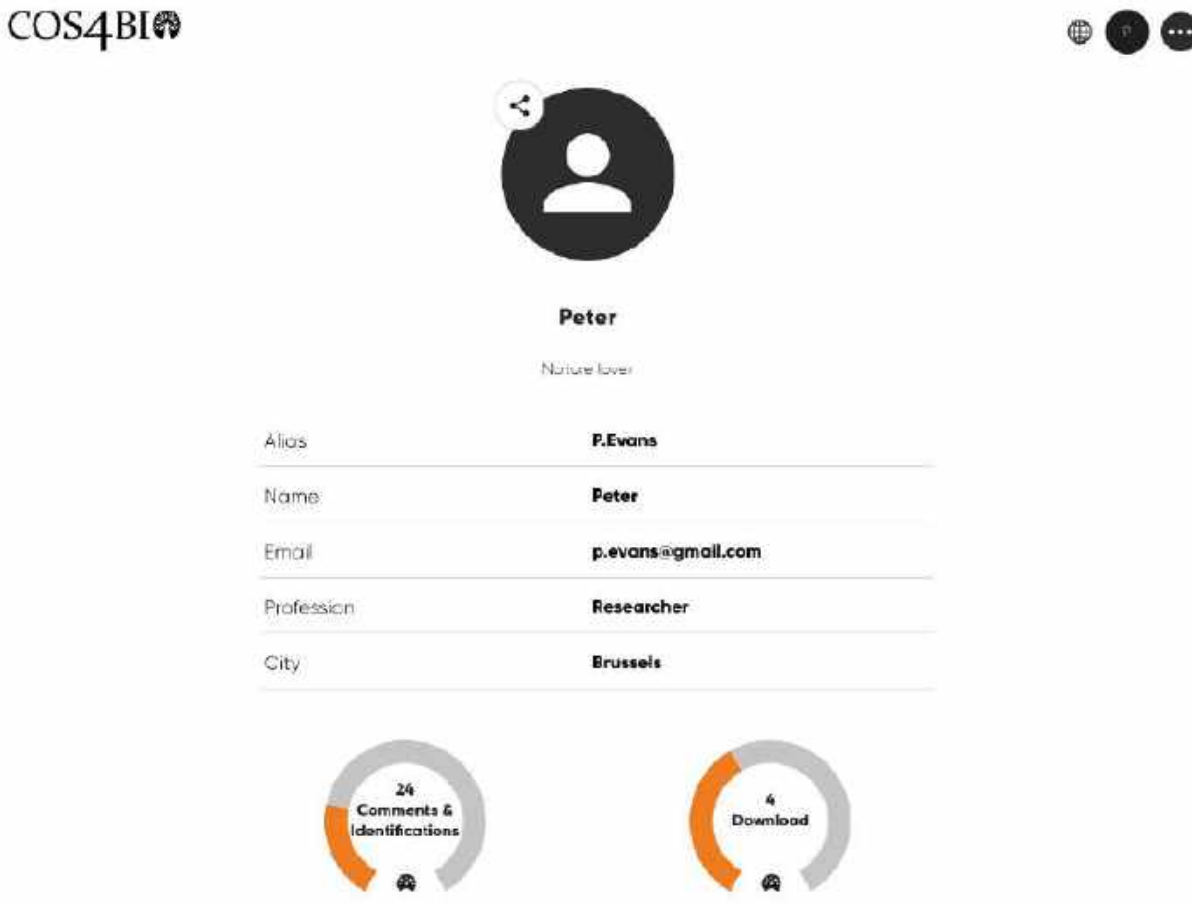



*Figure 32. User profile information and contribution charts.*

**Figure 33:** Below on the profile page, each user can also see more charts related to their number of contributions by month, year or all time, and the percentage of contributions per Citizen Observatory. This information will let us to know what is the users that do more contributions and what is the platform that receive more or less controtributions and when, to be able to establish some kind of relation with any event such as: holidays, weekends, BioMarathons etcs… and understand better the behaviours of our experts.

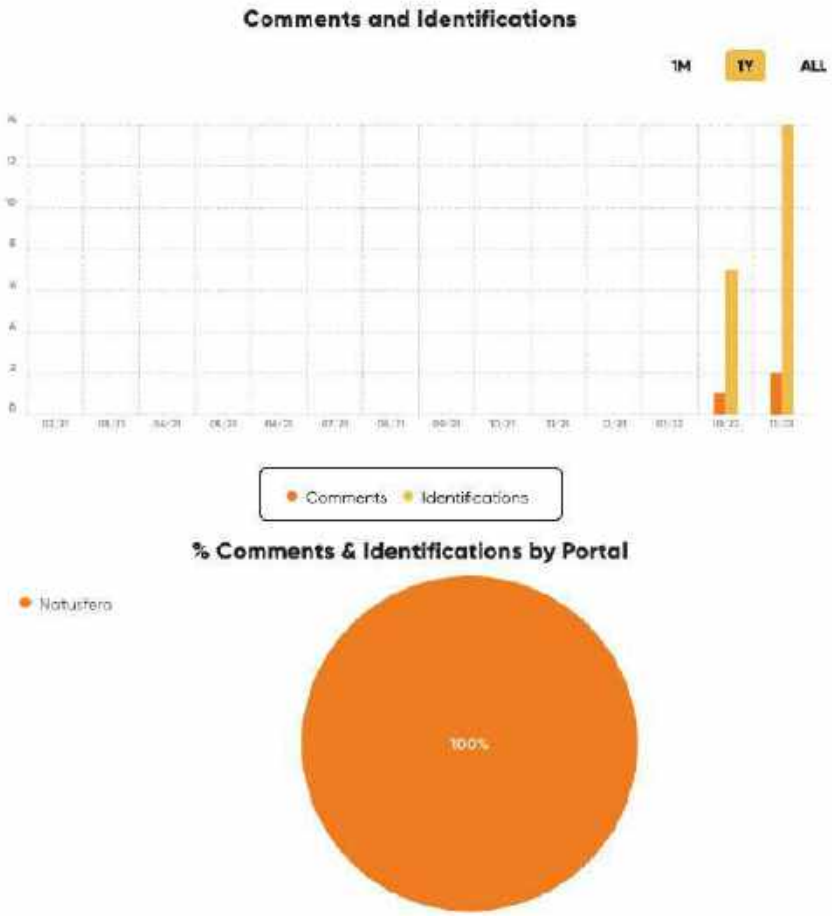



*Figure 33. User comments and identifications contributions time series and Comments/Identifications by Citizen Observatory.*

**Figure 34:** Besides previous charts, each user can also see the information graphs related to the downloads. Number of downloads by month, year or all and the percentage of download reasons. All of this information helps us to understand when the datasets are more requested and why.

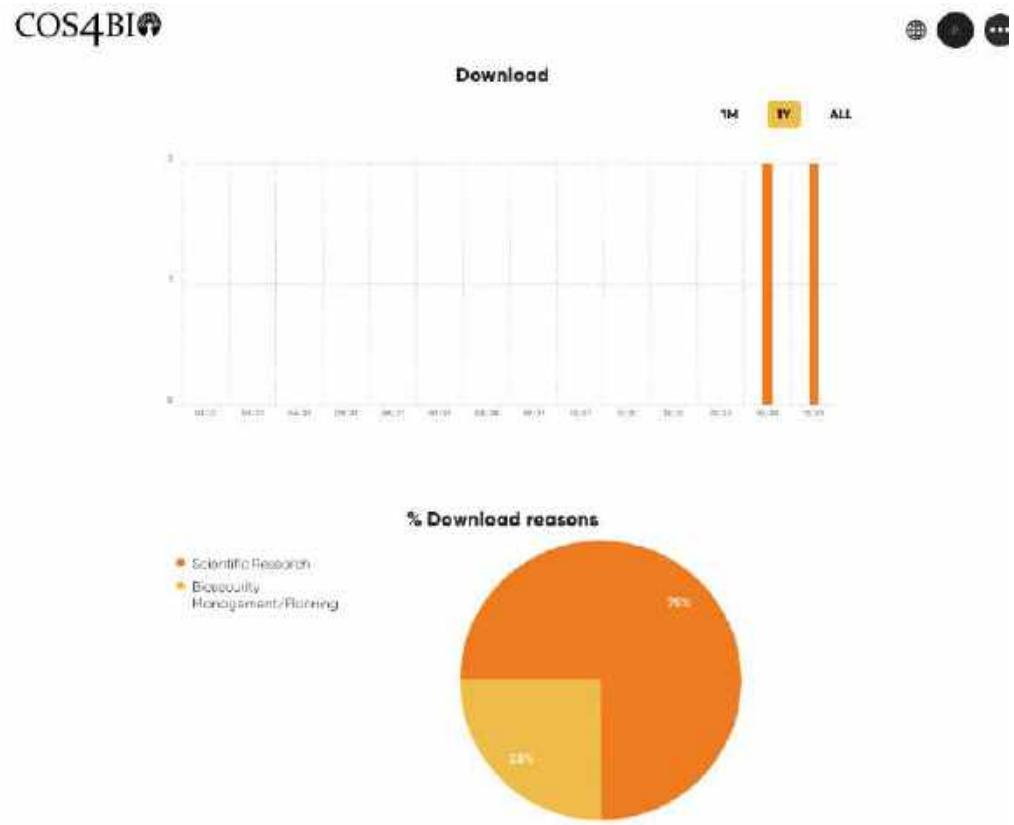



*Figure 34. User data download information.*

**Figure 35:** On the main page of Cos4Bio, we can see two of the main services, that are search and filter services. The experts can filter the information by different criterias, as we can see in the image below.

- By Portal: with the observatories that have been integrated into Cos4Bio.
- By Type: plantae, arachnida, mollusca, insecta...
- By Quality: we can filter by the quality of the observation: research, casual, with geolocation or photos.
- By Type of licence: CC0, CC-BY...
- By Date or range of Dates.

All of these filters can be used independently or combined with each other.

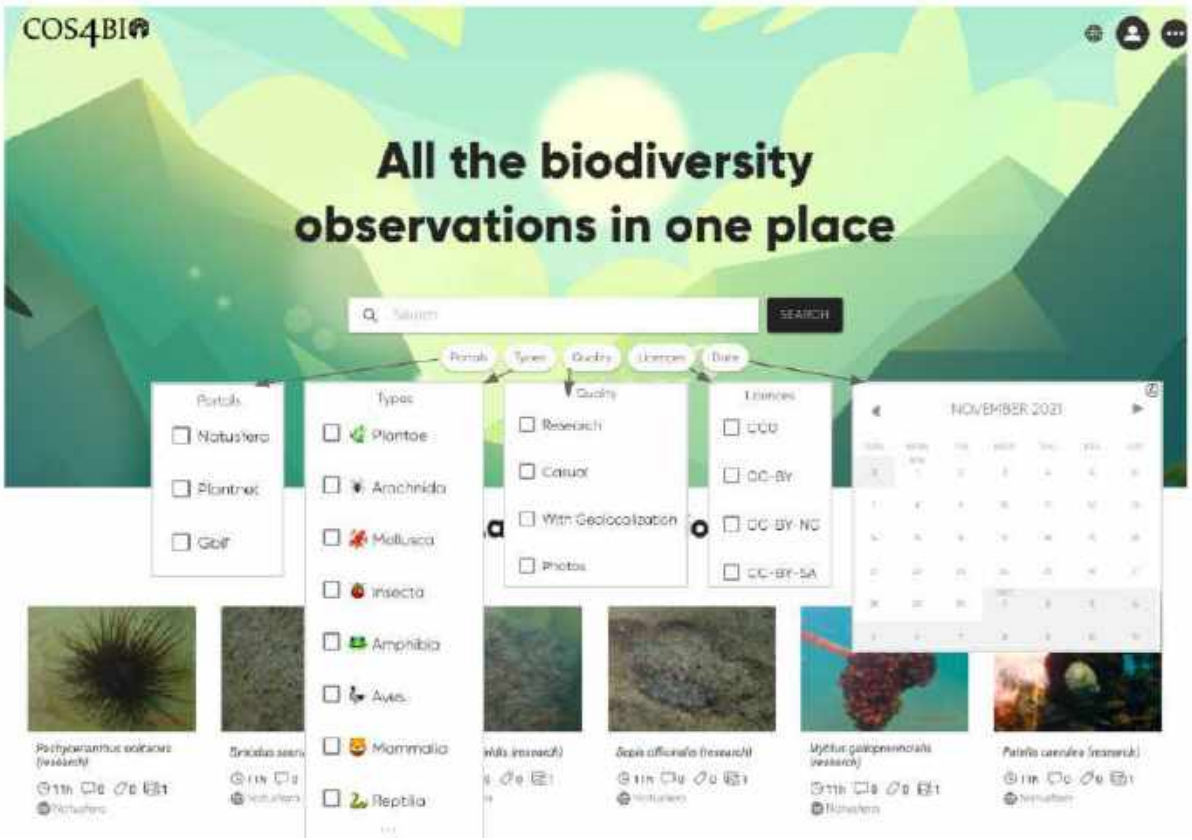



*Figure 35. Cos4Bio Filters by portal, types, quality, licences and date.*

**Figure 36:** The first thing that we find on the main page is the search service. The experts can use this service to find observations by species name, location or combining both types of searches. Besides this we have implemented a history of searches that he user has done recently in case that they want to search again using the same terms, as we can see in the next image:

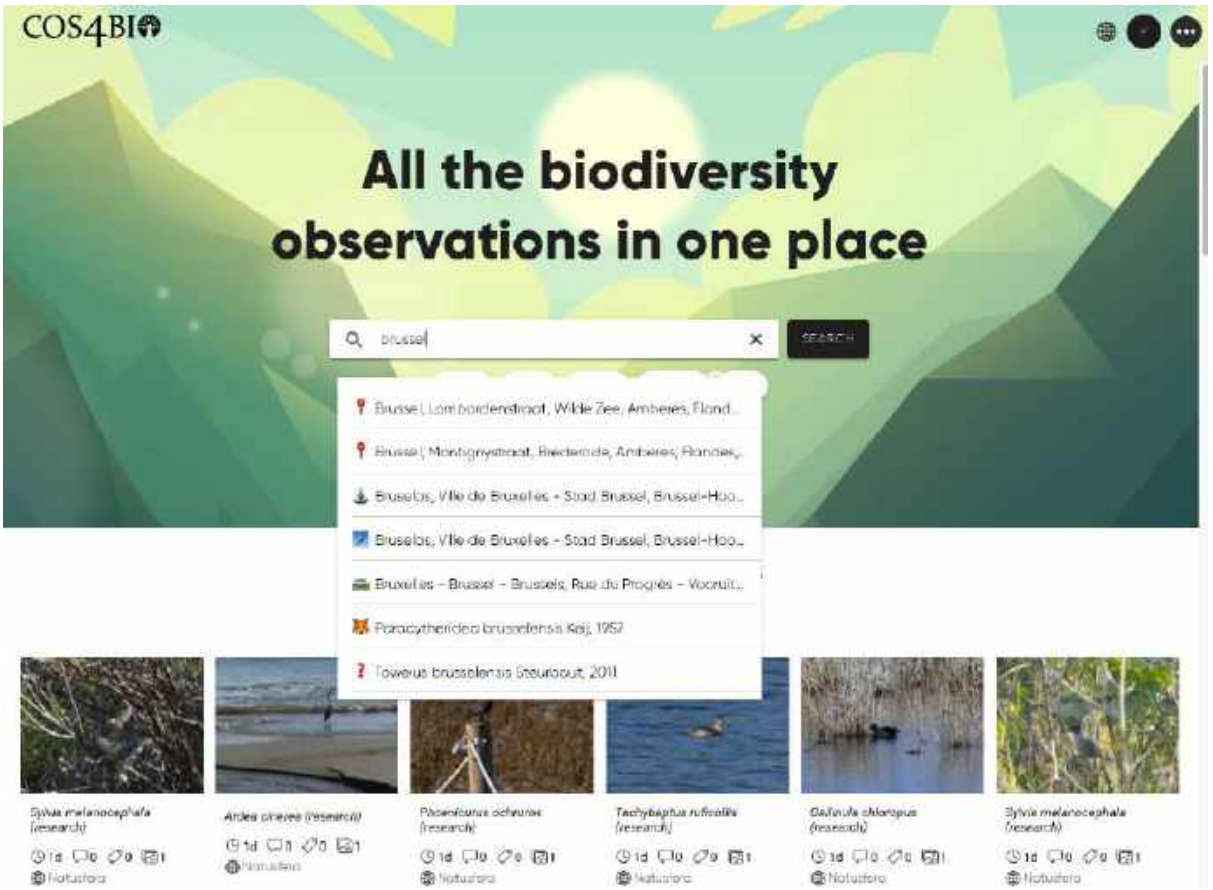



*Figure 36. Search service by species + location, and history of recent searches.*

**Figure 37:** Once we have carried out a search or have filtered the information, we have the option of downloading the data that meets the criteria that we have applied in CSV (Comman-separeted values) format.

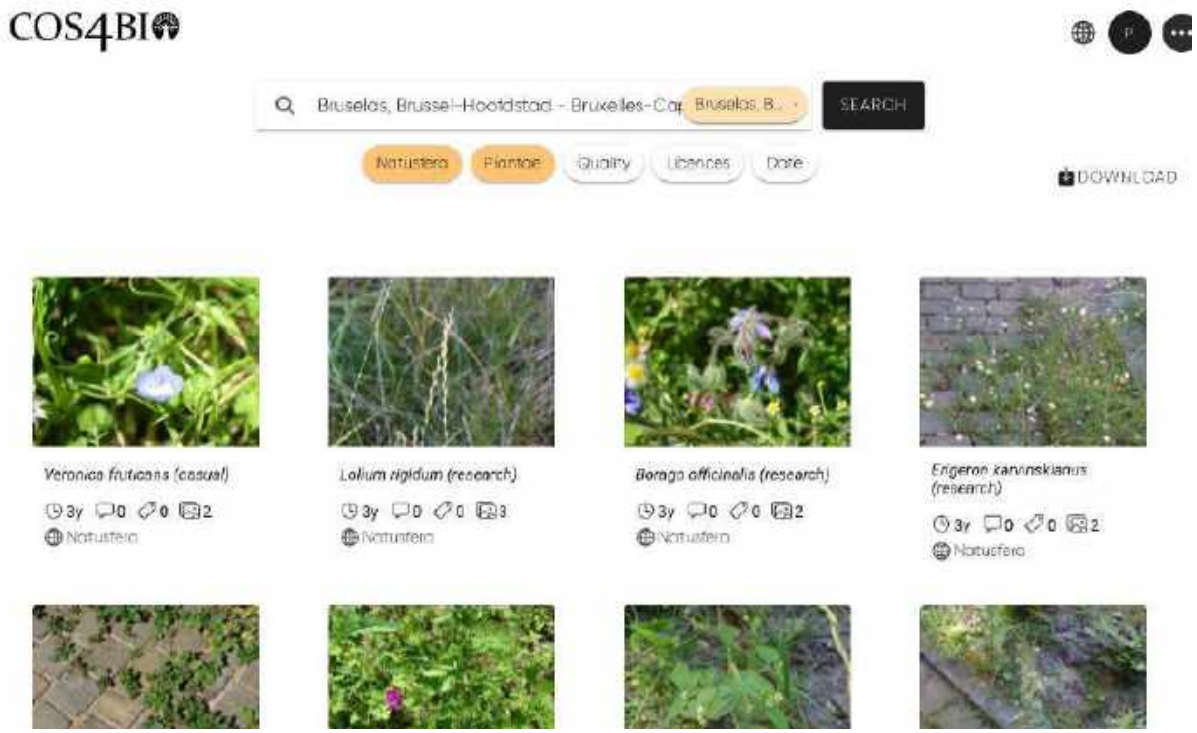



*Figure 37. Download information to a csv file.*

**Figure 38:** In this mockup, we can see the detail of an Observation where the following information appears: photograph, scientific name, and characteristics of the observation (observation record date, portal from which the observation came, latitude and longitude where it was taken and comments made by other users). We will also find a species search engine as well as the possibility of adding new comments and identifications.

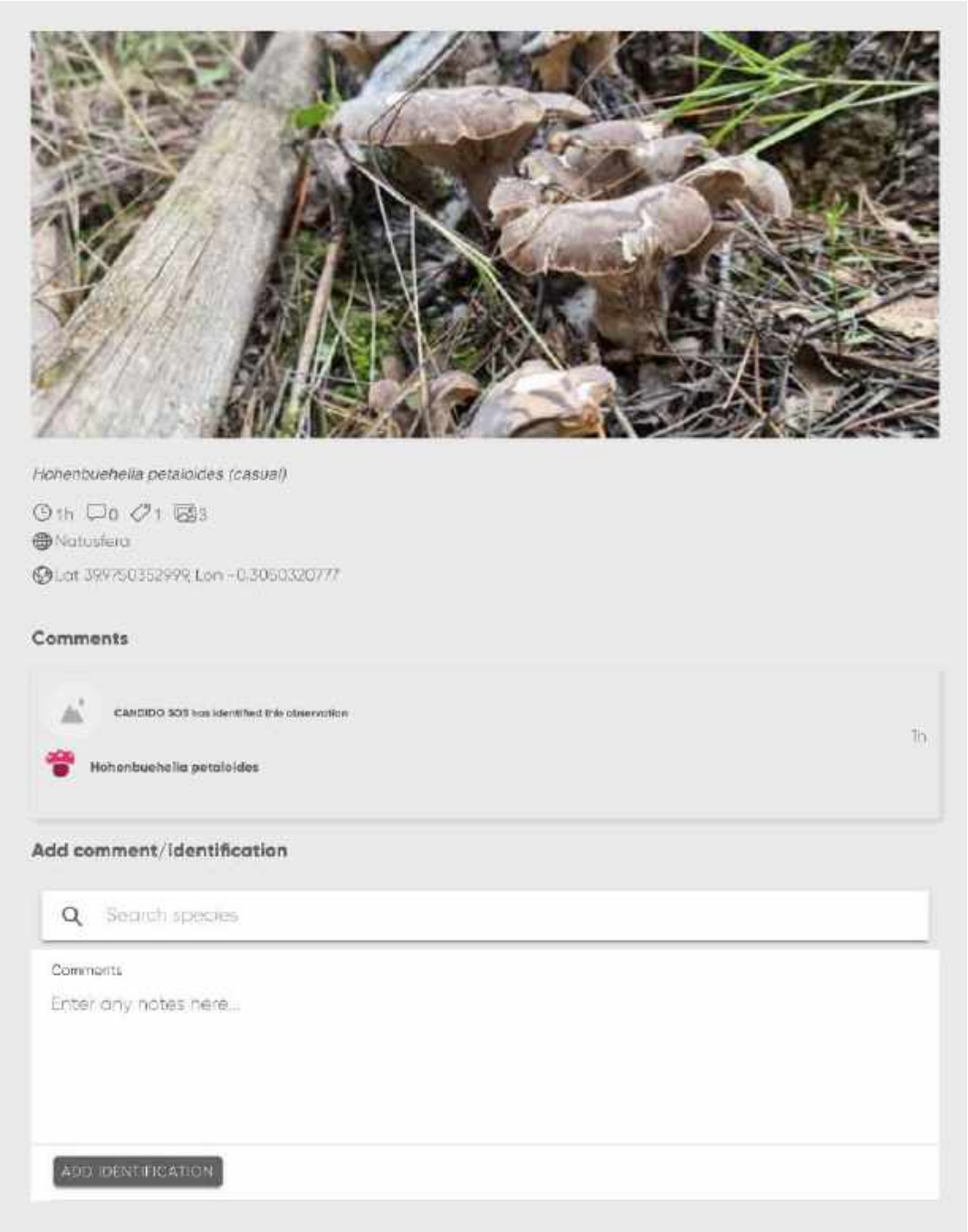



*Figure 38. Observation detail.*

**Figure 39:** The user has the possibility to update the information of his profile from the "settings" option that he will find in the menu.


*Figure 39. Settings.*

**Figure 40:** Cos4Bio users can give their feedback on the service by selecting the feedback option in the menu.


*Figure 40. Feedback.*

### 2.3.4 Prototype

As a result of the diagrams presented and the agreements reached during the different co-design meetings, we made an initial prototype, which could be tested by both internal experts and the experts of the Co-design group and which allowed us once reviewed to go to the next phase, the implementation phase , which we describe in section 3. Development The prototype implemented during this phase is available at the following public link https://cos4bio.eu

### 2.3.5 Test

In this section we detail which requirements were tested in the scope of Cos4Cloud, how testing has been implemented and how it was applied throughout the project.

The following requirements were tested and can be view in Table X:

1. What are the **relevant user stories** developed for Cos4Cloud?
2. What **criteria have to be fulfilled** in order to consider the user story as done?
    a. What testable criteria can "proof" that the business requirement described by the story is implemented?
    b. Each user story shall have a list of acceptance criteria that have to be tested and re-tested respectively.
3. Give a hint on the **maturity context**: Under what TRL the user story is considered important?
    a. What readiness level the output will have in terms of EU TRL
    b. The story may span multiple levels but ideally, a story is tailored to match a particular one
4. Under what **level of detail** tests have been performed?
    a. Gives an idea of the granularity under which testing took place
5. What aspects have been tested (**functional and non-functional**)?
    a. *non-functional* is an unworn actually, it covers all quality aspects and boundary conditions
6. How testing has been **implemented and is applied** (tools, workshops, continuous integration testing, manually following a checklist, ...)?
    a. When: automated tests, manual tests, during each build/deployment, before each release, ...
    b. How: see the list in functional and non-functional for some tooling
    c. Context: use Co-Design sessions, datathons, hackathons, ... to test certain aspects

7. Is there relation to other Cos4Cloud services? What are testable use cases when **integrating with other services**? For example
    a. A cross-point in the Technical Integration Experiment (TIE) table (describe the particular use case)
    b. An service integration of any other kind

#### 2.3.5.1Test implementation

The next steps hopefully will help to give answers on what requirements are actually put under test in the scope of Cos4Bio, how testing has been implemented and how it was applied throughout the project.

1. What are the **relevant user stories** developed for Cos4Bio?
2. What **criteria have to be fulfilled** in order to consider the user story as done?
    a. What testable criteria can "proof" that the business requirement described by the story is implemented?
    b. Each user story shall have a list of acceptance criteria which have to be tested and re-tested respectively.
3. Give a hint on the **maturity context**: Under what TRL the user story is considered important?
    a. What readiness-level the output will have in terms of EU TRL (See Annexe: TRL Calculation Cos4Bio )
    b. Story may span multiple levels but ideally a story is tailored to match a particular one.
4. Under what **level of detail** tests have been performed (see the test pyramid)?
    a. Gives an idea of the granularity under which testing took place.

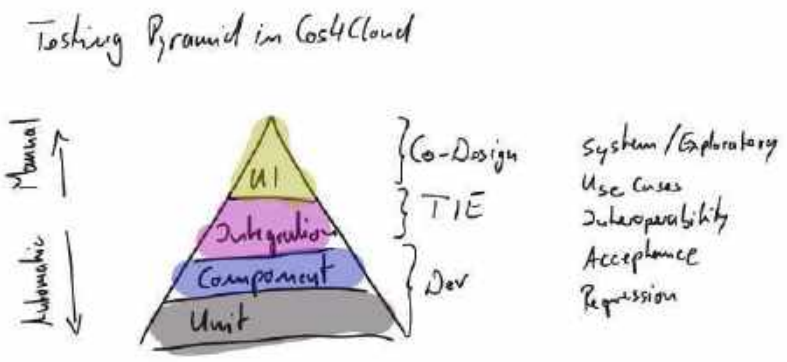



*Figure 41. Testing Pyramid.*

5. What aspects have been tested (**functional and non-functional**)?
    a. *non-functional* is an unword actually, it covers all quality aspects and boundary conditions
6. How testing has been **implemented and is applied** (tools, workshops, continuous integration testing, manually following a checklist, ...)?
    a. When: automated tests, manual tests, during each build/deployment, before each release, ...
    b. How: see the list in functional and non-functional for some tooling
    c. Context: use Co-Design sessions, datathons, hackathons, ... to test certain aspects
7. Is there relation to other Cos4Cloud services? What are testable use cases when **integrating with other services**? For example
    *a. A cross-point in the Technical Integration Experiment (TIE) table that we describe below in* Table 3: Technical Integration Experiment

| User Story | Relevance to Cos4Cloud | Relevant TRL (s) | Acceptance Criteria | Testing Granularity (see the test pyramid) | Access to be Tested | How to Verified/Tested? | Integration Test (e.g. TIE) |
|---|---|---|---|---|---|---|---|
| Experts search, filter, download, identify and comment observation from other CO using Cos4Bio | - UX/UI implementation.<br>- Service implementation.<br>- Interoperability | TRL-1<br>TRL-2<br>TRL-3<br>TRL-4<br>TRL-5 | - Accessible and Easy interface.<br>- Be able to search by species name and location.<br>- Include more filters.<br>- Be able to share their profile.<br>- Include a section for the experts | - API test<br>- Component<br>- Service | Functional:<br>- login<br>- logout<br>- search service by name and location<br>- filter service<br>Download service<br>Non-Functional<br>Queries takes no longer than 5 seconds | - Unit test<br>- API Test<br>- Co-Design sessions.<br>- Pl@ntNet integration.<br>- Natusfera integration.<br>- Unit test | References implementation:<br>https://github.com/Bineo-Consulting/Cos4Cloud |
| Natusfera integration | - Interoperability<br>- platform collaboration<br>- FAIR compliance | TLR-6<br>TRL-7 | - Include the MVE fields needed to be able to publish biodiversity observations following Darwin core terms.<br>- Don't affect a lot the model already exists in Natusfera. | - Service<br>- API test<br>- Integration test<br>- Include the minimum fields needed to be able to publish biodiversity observations following Darwin core terms.<br>- Don't affect a lot the model already exists in Natusfera. | Functional:<br>- mapping with terms of dwc:<br>- search by name<br>- search by location.<br>- filters by rank<br>- filters by quality fields.<br>- filters by licenses.<br><br>- Continuous data loading.<br>- Client consuming data.<br><br>Non-Functional:<br>- Fast response times. | - Unit test<br>- Co-Design session<br>- Cos4Cloud project in Natusfera<br>- API Test | Natusfera dwc API:<br>https://www.natusfera.org/dwc/observations/260642<br><br>https://github.com/Bineo-Consulting/natusfera-dwc |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Pl@ntNet integration | - Interoperability<br>- platform collaboration<br>- FAIR compliance | TRL-6<br>TRL-7 | - Include the MVE fields needed to be able to publish biodiversity observations following Darwin core terms.<br>- Don't affect a lot the model already exists in Natusfera. | - Service<br>- Integration test<br>- API test<br>- Include the minimum fields needed to be able to publish biodiversity observations following Darwin core terms.<br>- Don't affect a lot the model already exists in the Pl@ntNet model. | Functional:<br>- mapping with terms of dwc:<br>- search by name<br>- search by location.<br>- filters by rank<br>- filters by quality fields.<br>- filters by licences.<br>- Continuous data loading.<br>- Client consuming data.<br><br>Non-Functional:<br>- Fast response times. | - Unit test<br>- Co-Design session<br>- Cos4Cloud project in Natusfera<br>- API Test | Pl@ntNet dwc API:<br>https://my-api.plantnet.org/#/DarwinCore/getV2DwcOccurrenceSearch |
| GBIF standarization and integration | - Interoperability.<br>- Standardisation<br>- darwin core terms<br>- New API service<br>- User engagement | TRL-6<br>TRL-7<br>TRL-8 | - Create an API with the same format that GBIF built their Portal.<br>- Use standards: Darwin core and json. | - Service<br>- Integration test<br>- API test<br>- Standardisation | | - Unit test<br>- API Test | https://www.gbif.org/developer/occurrence |
| EOSC publication | - EOSC compliant<br>- Production environment | TRL-9 | - The service is accessible to users outside its original community,<br><br>- At least one service instance is running in a production environment.<br>- Public Research data which is Findable, Accessible, Interoperable and Reusable.<br>Release notes and sufficient documentation are available. | https://cos4bio.eu<br><br>About, Definition of Terms of Service, Deliverables, infographics, FAQ, Contact service, Feedback service, production environment, API based on Darwin core to apply FAIR rules, API documentation | | | https://marketplace.eosc-portal.eu/services/cos4bio?fromc=applications |

# 3. Engineering software design

## 3.1. Introduction

In this section we present the Cos4Bio Software Engineering analysis that will allow us to understand each and every one of the use cases present in the system, the interaction diagrams and the resulting domain model.

## 3.2. Vision

The main mission of Cos4Bio is to create an ecosystem that experts in Biodiversity related to Citizen Science can use to carry out searches and downloads quickly and in a standardised way, generating data sets from different sources of information, such as citizen observatories. We have also wanted to take a step forward, so that they can not only use Cos4Bio as a multi-data repository, but also contribute their knowledge and help the community by identifying the species that citizens capture through their devices, without said identification efforts remaining anonymous, since they can share their profile with the statistics generated, thus demonstrating their contribution to the community and their degree of knowledge.

**What are the differentiating elements and advantages?**

The principal advantages that we find in Cos4Bio are:

- A new way to increase the number of identifications for the CO.
- Reduce the time in which observations are validated.
- Increase the quality of identifications.
- Increase the number of experts participating.
- Be able to manage standardised information.
- Reduce the time that the experts have to spend to find information.
- Now with Cos4Bio, the experts can measure their impact at a global level about the contributions made by the experts in the different COs.
- The experts also can download all the Citizen Science information from a single site in a standardised format dwc.
- Intuitive and simple interface.

**Are there products or services that offer something similar?**

In the context of Biodiversity, the unique reference platform that aggregates information from different institutions or platforms is GBIF (Global Biodiversity Information Facility). However, this global infrastructure has several different characteristics compared to the Cos4Bio

- GBIF is not a central Citizen Science system, it also contains Biodiversity information from scientific Institutions.
- GBIF is not a real-time aggregation system, and it requires a complex publishing process.
- GBIF just contains validated information, that means, the scientific names of each observation have already been identified.

The last characteristic is what makes Cos4Bio a different service than GBIF, since in GBIF there is no type of interaction between the community. It simply acts as a huge repository, where people can consult or publish the Biodiversity that exists at the Global level in a standardised way, but importantly it does not contain Environmental information.

**Is there an equivalent product or service?**

As we mentioned in the previous section, GBIF is the service that most resembles it, in the Biodiversity information aggregation system characteristic, but Cos4Bio contains Citizen Science information in real-time, it does not have complex publication processes and allows direct interaction between citizens and experts. Also, the experts can demonstrate their contribution, and share their stats generated using Cos4Bio to the rest of the community.

**Is there a workaround that people are using that is good enough but not perfect?**

Up until now we do not know a service that performs the work we are carrying out, so we can say that Cos4Bio is an innovative system at a Global level.

**What are the strengths and weaknesses of the competition?**

If we define GBIF as a possible competitor of the Cos4Bio service, knowing the differences already mentioned in the previous sections, we can say as positive aspects that:

- GBIF has a very complex infrastructure that allows it to manage more than one thousand million records of information.
- Its organization is based on a system of national nodes in which each country pays an annual fee, which allows it to have the right to vote in the annual meetings, which gives stability to the organization.

- They have an open Community that makes contributions at different levels: technical, support, communication
- They are based on standards, and promote their use among the nodes that are part of the global network, which means that the same procedures are always followed, following the rules defined and validated by the Community.
- They actively participate in symposia and conferences that year after year give them more visibility and commitment.
- They have extensive experience, being a service that has been in operation for many years.

All these positive aspects will allow us in the future to define together with the CoNNect Group (consisting of WPs 5, 6, 7 and 8) a sustainability and leadership strategy for the Cos4Bio.

However, GBIF has a number of shortcomings due to its purely scientific nature, such as:

- There is no type of interaction with people, it works only as a repository of scientific information, that means that the information hosted on GBIF is already validated.
- It is not a real-time information system, when some institution wants to publish a dataset, it requires complex publication procedures.
- It only handles information on Biodiversity, when Biodiversity is much more than Flora and Fauna, since environmental variables have a direct impact, and this is one of the crucial aspects that cover Cos4Env.
- There is no recognition on the side of Citizens.

## 3.3. Use Case Diagram

Use cases are functional requirements that indicate what the system will do. They are the main mechanism for the discovery of these requirements and their definition, in a way that they define the way in which the system will behave.


*Figure 42. Use Case Diagram.*

# 4. Development

## 4.1 Introduction

In the development section, we will explain the technology that we have used based on the Agile Methodology and Co-design. We will also provide an explanation and an analysis from the point of view of the Frontend and Backend, to finally show an image of the system architecture.

## 4.2 Methodology

Additionally to Agile and Co-design methodology, we focus on reaching achievable milestones in short sprints to close the stages towards each objective . To reach a milestone, we follow three basic principles: Simplicity, Communication and Feedback.

- **Simplicity:** The fewer elements disperse the attention of the objective, the easier it is to achieve.

- **Communication:** This principle is indispensable. Each difficulty, doubt, comment or change is put in common to assess whether the final goal is achievable or what elements are prioritised over others.

- **Feedback:** The team seeks constant feedback from those responsible for the project, both when solving doubts, as well as to comment on positive points, strengths of the project, elements for improvement or possible changes. Thus, the team advances with confidence and with a clear objective.

## 4.3 Technology

In the technology section, we explain at the Frontend and Backend level the decision making regarding the methodologies used in the development and the choice of each one of them.

### 4.3.1 Frontend

The frontend is designed with a component-based methodology. All modern component-based frontend frameworks make project management easy, which in turn helps make it a long-term, easy-to-maintain project.

The main factor of a component-based methodology is to reject pieces of code that are intended for multiple uses, and clearer application architecture, and generally have a unified development process.

We have created 4 levels within the methodology:

- **Atom or Component:** They are the smallest blocks of the project, individual legos. We refer to text styles, buttons, icons, input fields, check boxes, etc. It does not depend on any other component or external agent to function.

- **Molecules or set of Components:** When we create a more complex component, and we need to reuse other components, this pattern of "molecules" arises. Like atoms, they do not depend on an external factor to function. For example, a search bar, a grid of images, etc.

- **Services:** They are pieces of code that aim to organize and share business logic, data and functions with different components and parts of an application. They provide very useful functions that help to divide the prototype into different small logical units that can be reused and that can be called from anywhere in the application, but strictly, it is each page that makes use of these services. For example, get data from a portal, add a comment, get credentials from an external service, etc.

- **Pages:** Pages are the modules that encompass and organize the previous levels. They are made up of multiple components and services. The main function is to organize and present the flow of data between services and components for its correct presentation.

These levels are reflected in the organisation of the interface source code:

```
/components
  /pages
    HomePage
    ObservationsPage
    ObservationPage
    ProfilePage
    DashboardPage
    HistoricDownloads
```

/modals
Login
Map
Download
Settings
Share

/shared
CommentsComponent
DownloadComponent
GridComponent
SearchComponent

/services
MappingService (API endpoint)
GbifService (Species service)
PlaceServices (Nominatim)

This organisation offers us better communication and implementation with the team. Having a naming convention and interface organisation play an important role in the success of a project. This results in less misunderstandings and much better implementation of designs and developments.

It is important to choose mature technologies because they are reliable. We have to understand that all technologies have a life cycle, so we need future-proof technologies.

The questions we must ask ourselves are:

**What companies sponsor the development of the technology in question?**
When large corporations are behind a technology, this means that in the short term it is difficult for the company to decide to abandon their support (although it can happen in some cases)

**How big is the community?**
The larger the community, the greater the chance that things will stick around for quite some time, and also give us continuous support and development updates of the tools and platforms.

**What is long-term support like?**
In this case, we care about long-term support.

Below we summarise a list of frameworks with their main characteristics:

**React**

React is maintained by Facebook. Large companies have incorporated this framework into their stack: Netflix, Dropbox, Pinterest, etc.

Advantage:

- React enjoys a large community.
- It stands out for its performance.
- It is easy to use.

Disadvantages:

- It has too many updates.
- Two ways to develop (Classes and Functional "hooks").

**Vue**

It is maintained by the community. Like React, large companies use it: Alibaba, Nasa, Gitlab, Nintendo, Amazon, etc.

Advantage:

- Flexible and simple to use.
- Stands out for its performance.
- Has great long-term support.

Disadvantages:

- The community is much smaller.
- It is not maintained by a large corporation.
- Two ways to develop (the classic and the "hooks").

**Angular**

It is maintained by Google. Some examples of companies that use them: Microsoft, Udemy, Paypal, etc.

Advantage:

- By default it uses TypeScript.
- It has a very stable and complex architecture (beneficial for large teams).

Disadvantages:

- The weight of the applications is enormous.
- The performance is worse than the previous ones.
- Version support is 6 months up to a maximum of 18 months.

**Stencil**

Stencil is a frontend framework maintained by Ionic (a leading company in the development of mobile applications with Javascript). What stands out about this company is its total focus on the sector. Some companies that use it: Apple, Amazon, Porsche, etc.

Advantage:

- It is the union of the best of the 3 previous frameworks.
- It uses TypeScript by default.
- Great performance.
- Very light.
- Future proof, as it generates standard web components.

Disadvantages:

- It is too new.
- The versioning policy is unknown (although we know it is a long term one).

For the expert portal, it has been decided to use Stencil, since it brings together the best of the most popular frameworks and does not lack any technology:

- Web Component-based.
- Asynchronous rendering pipeline.
- TypeScript support.
- One-way Data Binding.
- Component prerendering.
- Simple component lazy-loading.
- JSX support.
- Dependency-free components.

**Implementations:**

**More Filters**

Now that every portal is talking in the same language, we added more filters to the portal. The list of filters are:

- **Portals**: The list of available portals at the moment are PlantNet, Natusfera and Gbif.
- **Types**: The complete list of observations types are: 🌿 Plantae, 🕷️ Arachnida, 🦑 Mollusca, 🐞 Insecta, 🐸 Amphibia, 🦆 Birds, 🦁 Mammalia, 🐍 Reptilia, 🐠 Actinopterygii, 🐱 Animalia and Fungi.
  Obviously, PlantNet only has Plantae.
- **Quality**:
  - Research (if the species is identified)
  - Casual (we are not sure about the species name)
  - With Geolocalization (we have the latitude and longitude of the occurrence)
  - Photos (we have the media image of the occurrence)
- **Licences**: "CC0", "CC-BY", "CC-BY-NC" and "CC-BY-SA".
- **Date**: We can filter by the event date in a range of min and max periods. So this filter needs a calendar to be really useful, for the calendar we add a lightweight library called "**When".**

**Nominatim service**

Nominatim uses OpenStreetMap data to find locations on Earth by name and address, so allow us to type free-form queries ("Paris, New York ") in any language. In this way, we can filter occurrence by the bounding box area of the place requested.

**Gbif species list**

We rely on the Gbif public API to integrate scientific names of species. The advantage is that it is a frequently updated service offering a single source of data.

### 4.3.2 Backend

We need a backend to be able to map the observations of the different portals, perform the CSV download and verify the credentials of the users using Authenix service. We could define the backend as the data access layer.

For this case we have used a very popular stack (Node - Swagger - Express - Docker):

- **Node:** We choose this framework because it is lightweight and is the same language as the frontend.
- **Swagger:** We need an API that maps the different portals (Natusfera, iSpot, PlantNet, ArtPortalen, ...), for this we have documented the API with the Swagger specification.
- **Express:** Very popular node microframework, high performance and stable since 2012.
- **Docker:** Users can download the docker virtual container/image to install locally and make changes or customise the software when needed.

**Interoperability layer**

**API DWC**

In the backend we implemented a mapping service that is based on Darwin Core terms, so every portal should speak and understand these terms in order to be able to be included in the expert portal.

At the moment only origins like Natusfera, PlantNet and Gbif had implemented their API service based on Darwin core.

The client sends a request to the Node API, and the Dwc controller processes the request and determines which origins should be fetched in order to build the response, so we define the origin file "origins.js" for the endpoints of every origin. After queried the origin(s), the mapping function takes place, and it returns the mapping result.

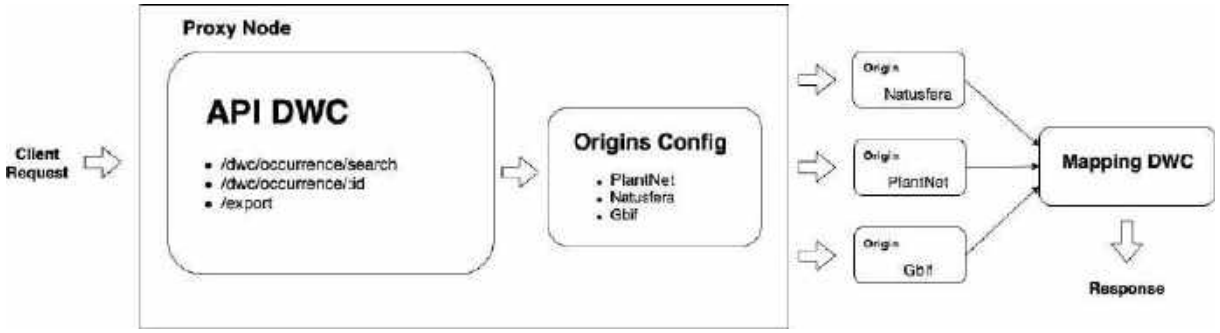



*Figure 43. API DWC.*

The mapping task consists of merging the response that comes from every origin(each Citizen Observatory) and finding the equivalent terms in the Darwin core standard Terms as shown in Table 4. Mapping terms table:

- **Terms**: the terms that we can find in the Citizen Observatories.
- **Value**: The possible values that the terms could have
- **Map dwc:term**: the term in dwc that we use to map with the term in the Citizen Observatory.
- **Description**: is the description of the dwc term.

| Term | Value | map dwc:term | Description dwc |
|---|---|---|---|
| basisOfRecord | PreservedSpecimen, FossilSpecimen, **LivingSpecimen**, MaterialSample, Event, HumanObservation, MachineObservation, Taxon, Occurrence | http://rs.tdwg.org/dwc/terms/**basisOfRecord** | Additional information that exists, but that has not been shared in the given record. |
| type | **StillImage**, MovingImage, Sound, PhysicalObject, Event, Text | http://purl.org/dc/elements/1.1/type | The nature or genre of the resource. |
| language | en, es, pt | http://purl.org/dc/elements/1.1/language | A language of the resource. |
| licence | http://creativecommons.org/publicdomain/zero/1.0/legalcode, http://creativecommons.org/licenses/by/4.0/legalcode | http://purl.org/dc/terms/license | A legal document giving official permission to do something with the resource |
| rightsHolder | The Regents of the University of California | http://purl.org/dc/terms/rightsHolder | A person or organisation owning or managing rights over the resource. |
| accessRight | not-for-profit use only | http://purl.org/dc/terms/accessRights | Information about who can access the resource or an indication of its security status. |

| | | | |
|---|---|---|---|
| id | 298381 | http://rs.tdwg.org/dwc/terms/**occurrenceID** | An identifier for the Occurrence (as opposed to a particular digital record of the occurrence). In the absence of a persistent global unique identifier, construct one from a combination of identifiers in the record that will most closely make the occurrenceID globally unique. |
| occurrenceStatus | PRESENT / ABSENT | http://rs.tdwg.org/dwc/terms/occurrenceStatus | A statement about the presence or absence of a Taxon at a Location. |
| user | José E. Crespo. Oliver P. Pearson \| Anita K. Pearson (where the value in recordNumber OPP 7101 corresponds to the collector number for the specimen in the field catalog of Oliver P. Pearson). | http://rs.tdwg.org/dwc/terms/recordedBy | Name of people, group, or organization responsible for recording the original Occurrence. The primary collector or observer, especially one who applies a personal identifier (recordNumber), should be listed first. |
| licence | http://creativecommons.org/publicdomain/zero/1.0/legalcode, http://creativecommons.org/licenses/by/4.0/legalcode | http://purl.org/dc/terms/license | A legal document giving official permission to do something with the resource. |
| description | text | http://rs.tdwg.org/dwc/terms/occurrenceRemarks | Comments or notes about the Occurrence. |
| | | LOCATION CLASS | |
| decimalLatitude | 41.3280161881 | http://rs.tdwg.org/dwc/terms/decimalLatitude | The geographic latitude (in decimal degrees, using the spatial reference system given in geodeticDatum) of the geographic center of a Location. Positive values are north of the Equator, negative values are south of it. Legal values lie between -90 and 90, inclusive. |

| | | | |
|---|---|---|---|
| decimalLongitude | 2.1092162071 | http://rs.tdwg.org/dwc/terms/decimalLongitude | The geographic longitude (in decimal degrees, using the spatial reference system given in geodeticDatum) of the geographic center of a Location. Positive values are east of the Greenwich Meridian, negative values are west of it. Legal values lie between -180 and 180, inclusive. |
| eventDate | 2021-04-10 | EVENT_DATE | The date-time or interval during which an Event occurred. For occurrences, this is the date-time when the event was recorded. Not suitable for a time in a geological context. |
| identificationVerificationStatus | casual | http://rs.tdwg.org/dwc/terms/identificationVerificationStatus | A categorical indicator of the extent to which the taxonomic identification has been verified to be correct. |

| Natusfera field | Value | mapd dwc:term | Description dwc | API |
|---|---|---|---|---|
| id | 6538 / 8fa58e08-08de-4ac1-b69c-1235340b7001 | http://rs.tdwg.org/dwc/terms/**taxonID** | An identifier for the set of taxon information (data associated with the Taxon class). May be a global unique identifier or an identifier specific to the data set. | |
| name | Tamarix parviflora | http://rs.tdwg.org/dwc/terms/**scientificName** | The full scientific name, with authorship and date information if known. When forming part of an Identification, this should be the name in lowest level | scientificName |

| | | | | |
|---|---|---|---|---|
| | | | taxonomic rank that can be determined. This term should not contain identification qualifications, which should instead be supplied in the IdentificationQualifier term. | |
| kingdom | Animalia, Archaea, Bacteria, Chromista, Fungi, Plantae, Protozoa, Viruses | http://rs.tdwg.org/dwc/terms/kingdom | The full scientific Name of the kingdom in which the taxon is classified. | ?kingdom=Animalia |
| phylum | Chordata (phylum). Bryophyta (division). | http://rs.tdwg.org/dwc/terms/**phylum** | The full scientific name of the phylum or division in which the taxon is classified. | phylum |
| class | Mammalia, Hepaticopsida | http://rs.tdwg.org/dwc/terms/**class** | The full scientific name of the class in which the taxon is classified. | class |
| order | Carnivora, Monocleales | http://rs.tdwg.org/dwc/terms/**order** | The full scientific name of the order in which the taxon is classified. | order |
| family | Felidae, Monocleaceae | http://rs.tdwg.org/dwc/terms/**family** | The full scientific name of the family in which the taxon is classified. | family |
| genus | Puma, Monoclea | http://rs.tdwg.org/dwc/terms/**genus** | The full scientific name of the genus in which the taxon is classified. | genus |
| specificEpithet | | http://rs.tdwg.org/dwc/terms/**specificEpithet** | The name of the first or species epithet of the scientificName. | specificEpithet |
| rank | subspecies, varietas, forma, species, | http://rs.tdwg.org/dwc/terms/**taxonRank** | The taxonomic rank of the most specific name in the | taxonRank |

| | | | | |
|---|---|---|---|---|
| | genus | | scientificName. | |
| taxonomicStatus | invalid, misapplied, homotypic synonym, accepted | http://rs.tdwg.org/dwc/terms/taxonomicStatus | The status of the use of the scientificName as a label for a taxon. Requires taxonomic opinion to define the scope of a taxon. Rules of priority then are used to define the taxonomic status of The nomenclature contained in that scope, combined with the experts opinion. It must be linked to a specific taxonomic reference that defines the concept. | taxonomicStatus |
| common_name | Andean Condor, Condor Andino, American Eagle, Gänsegeier | http://rs.tdwg.org/dwc/terms/**vernacularName** | A common or vernacular name. | vernacularName |

*Table 4: Mapping terms table*

**Natusfera DWC API**

Natusfera is based on iNaturalist core, so the implementations of darwin core terms are slightly different, but we fixed it with the new API based exclusively on Darwin Core terms.

The technology behind Natusfera is Ruby on Rails with a PostgreSQL database and Apache proxy server. Because It would be more difficult to make changes to this framework, for this purpose, we choose Nodejs again to include a Proxy server that understands and talks Darwin core terms.

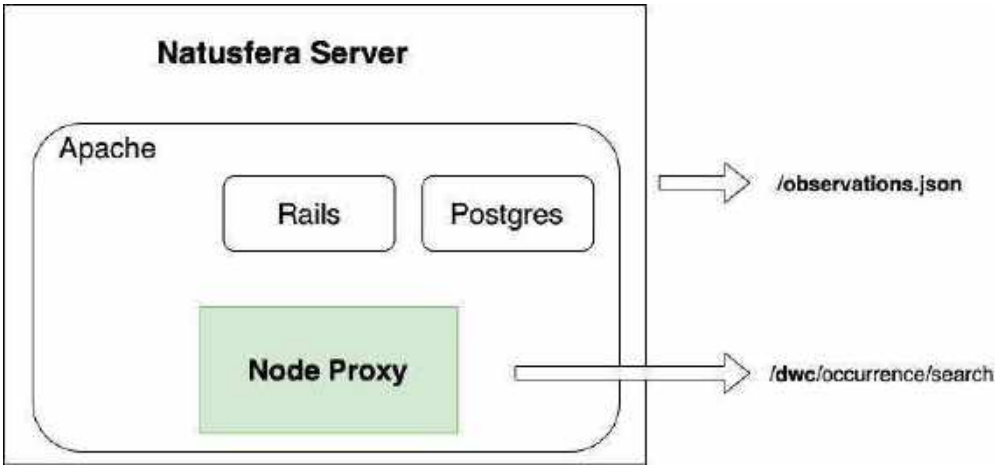



*Figure 44. Natusfera DWC API.*

The mapping from observation to occurrence (dwc) is as follow:

| **Natusfera Core** | **Occurence Dwc** |
|---|---|
| id | natusfera-{id} |
| eventDate | created_at |
| created_at | created_at |
| observedOn | observed_on |
| scientificName | species_name |
| - | ownerInstitutionCodeProperty “Natusfera” |
| identificationVerificationStatus | quality_grade |
| - | basisOfRecord “LIVING_SPECIMEN” |
| - | type “StillImage” |
| - | accessRight “not-for-profit” |
| licence | licence |
| rightsHolder | user_login |
| taxon | taxon |
| | occurrenceStatus “PRESENT” |
| description | description |

### 4.3.3 Architecture

The result of this structure of the Expert Portal is represented graphically as we can see below:

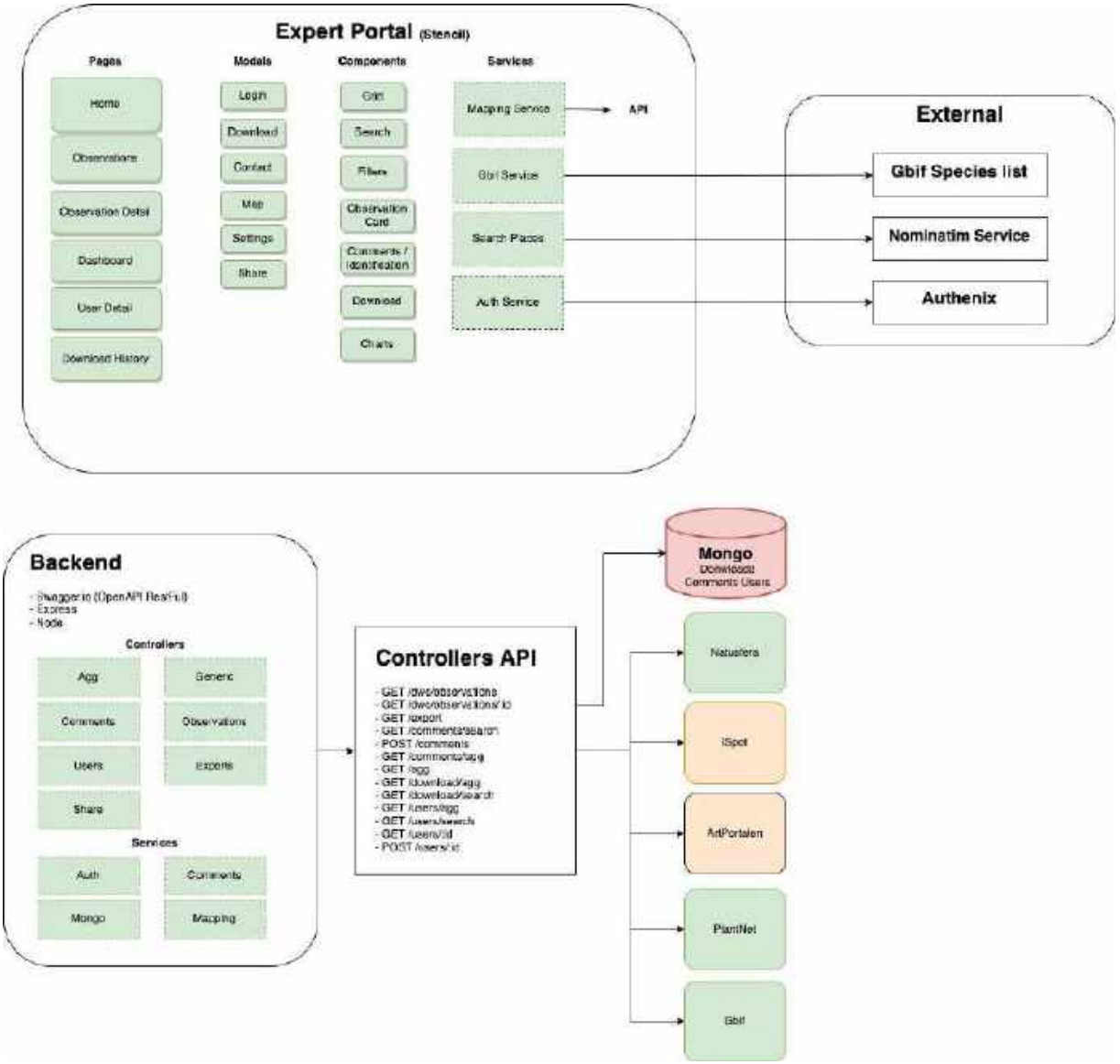



*Figure 45. Architecture Expert Portal.*

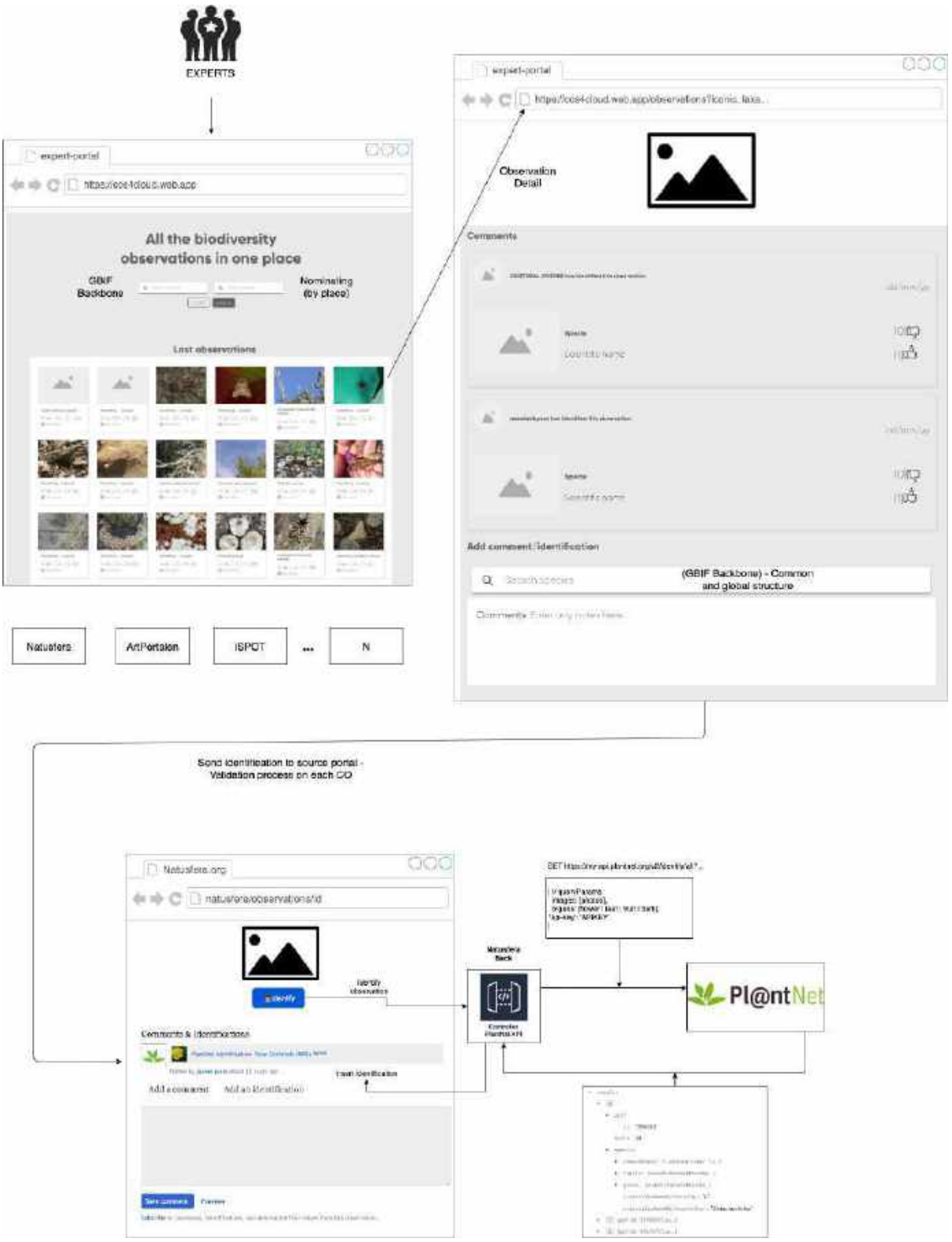



*Figure 46. Data Flow between Expert Portal and Citizen Observatories.*

**Stats and Mongo Database**

We have chosen MongoDb for the following reasons:

- It is NoSQL and document-based and stores data in JSON-like structures.
- Since we are using NodeJS as a backend, the most used architectures with Node and Mongo are MERN (Mongo Express React Node.js) or MEAN (Mongo Express Angular Node.js), which suits us like a glove.
- It has a powerful syntax for querying and grouping.
- Lastly, and most importantly, it is open source.

Apart from the API, we have the need to integrate a database for statistics purposes and substantially enhance the functionalities implemented before, like the export service and the comments:

Mongo helps us build this list of features:

- We store information about downloads, so we can show information about historic downloads for each user.
- We store user information that comes from Authenix service, this helps build the profile for each user. Also, with this information we show stats like charts and counters in each user profile.
- Every comment made from the portal is stored in mongo. This helps us build stats for comments and to show comments in portals that don't have comments features like PlantNet and Gbif.
- The information stored in Mongo helps us build a global Dashboard. This dashboard consists of a list of counters (comments, download and users) and a series of charts (Pie and Bars) showing data of the Portal.

The architecture of the API that consume the Mongo database consists in three concept endpoints:

- We have 3 models stored: Users, Comments and Downloads
- Each model has a generic endpoints:
    - /${model}/:id
    - /${model}/search
    - /${model}/agg
- /:id return an specific document by the id
- "Search" it lists all the rows of the models stored
- "Agg" lists stats grouped by Months and by type.

**Download/Export**

Because each portal offers the possibility of responding to queries in the same format (DarwinCore terms), now it is easier to export information in CSV. Also for each user we ask the reason for each request download, so we stored this information and also we offer a download history.

The reason why the user needs to download the information needs to be saved. This data collection is necessary to understand the reasons and to be able to represent it in the user's profile and the general dashboard. The reasons are listed below:

- ➔ Biosecurity management/planning
- ➔ Citizen Science
- ➔ Collection management
- ➔ Conservation management/planning
- ➔ Ecological research
- ➔ Education
- ➔ Environmental assessment
- ➔ Restoration/Remediation
- ➔ Scientific investigation
- ➔ Systematic research/taxonomy
- ➔ Species modelling
- ➔ Test
- ➔ Other

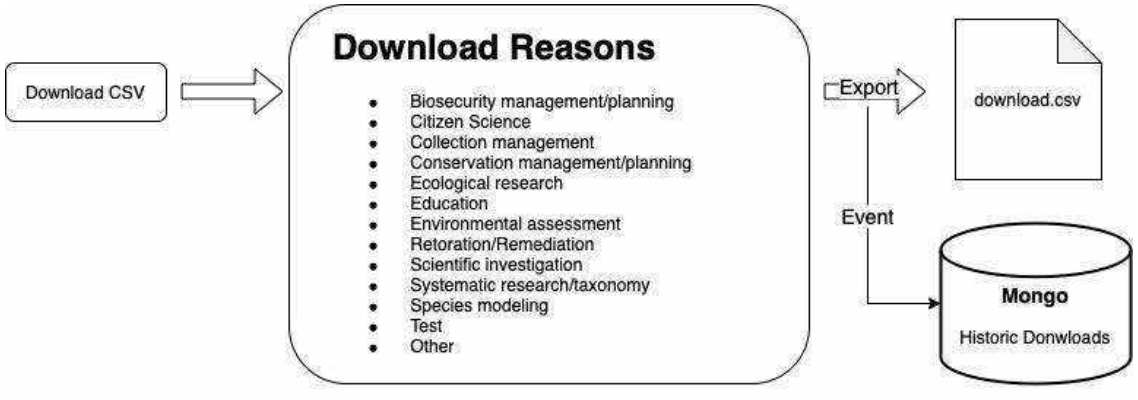



*Figure 47. Architecture download reason.*

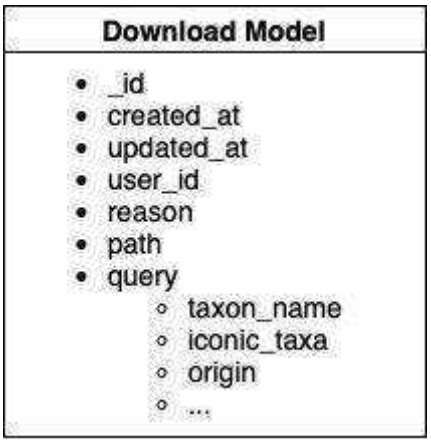



*Figure 48. Download model.*

The fact that we save the event and the reason for the download allow us to offer a download history for each user. The user in his list of downloads can consult and can re-execute his previous downloads.

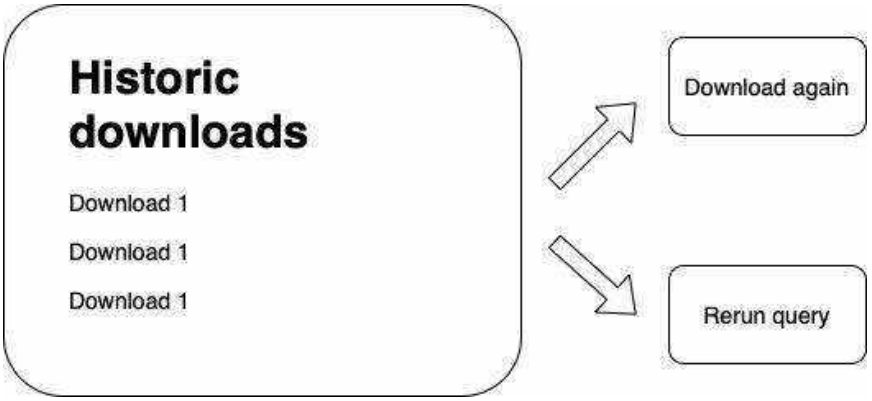



*Figure 49.Historic downloads.*

### 4.4. Code

At this moment we all the code is available on a public GitHub repository:
https://github.com/Bineo-Consulting/Cos4Cloud

## 5. Future work

Once the implementation has been tested and validated by experts and the Cos4Bio service has been published in EOSC, the future work has focused on several aspects that we detail below:

- Continuous maintenance of the system.
- A follow-up of the identifications made by the experts.
- Integrate new Citizen Observatories.
- To be in contact with the Scientific Community.
- Carry out communication work to make the service better known with the help of the Connect Group.
- Resolve possible incidents that may occur.
- Continue collecting comments from experts.
- Integrate Cos4Bio with more Cos4Cloud services such as DUNS.
- Think about adding new features.
- Plan and think about ways of sustainability for the future.

# 6. Conclusions

Cos4Bio has been a rewarding challenge. The possibility of being part of a European project has allowed us to collaborate with many partners throughout the implementation process, which is reflected in different aspects:

- Collaboration with other citizen observatories.
- Integration of other services of Cos4Cloud, such as Authenix or the Plant Identification Service.
- Application of the defined agile methodology.
- Creation of an interoperability layer that complies with the FAIR criteria.
- Learn from many experts thanks to Co-Design meetings.
- Improve in each cycle through technical meetings every 15 days.
- Join the EOSC community.

All these factors have resulted in a service that is simple to understand and at the same time powerful due to the opportunities it creates for experts and the world of citizen science.

It has also allowed us to demonstrate the value of the general purpose platform, which we developed during the first period, which with the inclusion of suggestions from experts and with several architecture modification that with the use of agile methodologies, has served as a stable base on which we have been able to build Cos4Bio. But the potential of this platform does not end here, since we are carrying out the development of Cos4Env on it.

Still, we know the job is not done yet. We hope that Cos4Bio will be a benchmark at European level as a central repository of citizen science data, that it will be a service used by all scientists to analyze data from many platforms and that they can also contribute their knowledge.

But in order to get here, we will have to get a lot of support, work with the communications service, and get more partners to make Cos4Bio a sustainable service in the future, continue including new features, continue listening to experts from the scientific community, to continue improving, contributing and making Cos4Bio a service for the Future.

# Glossary

| Term | Description |
|---|---|
| Cos4Bio | It's the name that we have defined for the service of Expert Portal for Biodiversity data validation. |
| Expert Platform | Prototype that integrates data from various citizen observatories, allowing the experts who access it to perform various operations such as: searches, downloads, identifications, comments, with the benefit of reducing the investment of time when finding data on which to investigate, and that allows increasing the degree of participation of experts in the processes of identification of the observations made by citizens. |
| Citizen Observatory (CO) | Web Portal that allows the interaction between Citizens and Scientists in obtaining data, be it data on flora, fauna, odours, temperatures, precipitation measurements, emissions ..., and that involves an identification process , validation, quality and study. |
| Observation | It is the element observed by a person or device, be it Flora, Fauna or an Environmental Variable. |
| Identification | When the observations are not known by the citizens, they are presented in a random state, not identified, since they lack the scientific knowledge to be able to define the observation with precision. For example: In the case of a magpie, it would be identified when a user identified it as Pica Pica, and defined it through its correct scientific name. |
| Casual status | An observation is accidental when it has received an identification, but this has not yet been validated by any expert in its corresponding Citizen Observatory. |
| Research grade status | An observation is scientific grade, when it has been identified and in turn has been validated by an expert. |
| Citizen | Person who participates in Citizen Observatories sharing their observations related to biodiversity or environment context that can be used by the scientific community. |

| | |
|---|---|
| Expert | Person with sufficient knowledge to be able to carry out identifications or scientists dedicated to the world of Biodiversity or the Environment. |
| Standard | It is a common framework validated by the community, in this case it applies to the use of communication protocols, the representation of information, the visualization of images and procedures. |
| Interoperability | We can define it, within the project, as the layer that allows the data from the different Citizen Observatories to coexist in the same environment. This layer allows the Expert Portal to understand the information in each portal, performing a mapping task and showing it to the end user in a unified and standardized way. |
| FAIR | FAIR are data which meet principles of findability, accessibility, interoperability, and reusability. A March 2016 publication by a consortium of scientists and organizations specified the "FAIR Guiding Principles for scientific data management and stewardship" in Scientific Data, using FAIR as an acronym and making the concept easier to discuss. |
| Actor | it is something with behaviour, such as a person, identified by a role, computer system or organization. |
| Scenario | Specific sequence of actions and interactions between the actors and the system under study. |
| SSD | In software engineering, a system sequence diagram (SSD) is a sequence diagram that shows, for a particular scenario of a use case, the events that external actors generate, their order, and possible inter-system events. (wikipedia) |

# Annexe

## Cos4Cloud TRL Calculation

**Service:**
**COS4BIO: Expert Portal for Biodiversity Data Validation**

### TRL 1: Basic principles observed

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Is there a research hypothesis defined guiding the development of the service? | yes |
| Are the basic algorithms needed for the implementation defined? | yes |
| Is there a document (specification, publication) describing the ideas of the service? | yes |

### TRL 2: Technology concept formulated

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Is there an initial design description of the service? | yes |
| Did you define the users stories to be implemented? | yes |
| Did you identify the research and development activities necessary for developing your service? | yes |
| Did you perform first tests with example data? | yes |

### TRL 3: Experimental proof of concept

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Did you define the main components of your service? | yes |
| Is there an overview design of your service available (e.g. component diagram)? | yes |
| Did you define performance criteria (e.g., number of users, amount of data) for your service? | yes |
| Did you establish a test environment for component tests (e.g., tools for conducting unit tests)? | yes |
| Are the individual components of your service implemented in a preliminary version? | yes |
| Did you test the core functions of the service? | yes |

| | |
|---|---|
| Did you document your first tests? | yes |

**TRL 4**

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Is there a detailed design of the components of the service available? | yes |
| Did you complete the first running on an internal system (development machine)? | yes |
| Did you define the environment in which the service will be operated? | yes |
| Did you successfully complete the testing of the integrated components? | yes |
| Are the test results of the integrated components documented? | yes |

**TRL 5**: **Technology validated in relevant environment**

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Did you determine the requirements for scaling your service (e.g. from test users to a public audience)? | yes |
| Did you establish a testing environment similar to the later operational environment? | yes |
| Did you successfully demonstrate your implementation in a testing environment? | yes |
| Did you conduct tests with consortium partners? | yes |
| Are the test results from the test environment documented? | yes |

**TRL 6: Technology demonstrated in relevant environment**

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Are the system requirements finalised? | yes |
| Is the operating environment fully defined? | yes |
| Are the necessary implementations of a prototype that takes into account the requirements of the operational environment? | yes |
| Did you conduct tests with selected external users (e.g. as part of hackathons)? | yes |

**TRL 7: System prototype demonstration in operational environment**

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Is there a fully integrated prototype available that was demonstrated in an operational environment? | yes |
| Did you verify the performance of the system? | |

| | |
|---|---|
| Did you conduct tests within a broader community with previously unknown users? | yes |
| Is all software testing completed? | yes |
| Does your service provide value on its own? | yes |
| Is your documentation available in the English language? | yes |
| Did you specify privacy statements, terms of use and Service Level Agreements? | yes |
| Is there a help desk available? | yes |
| Is there a plan established for ensuring regular updates of your service? | yes |

### TRL 8: System complete and qualified

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Did you define a life cycle for your software? Is there a clear plan on how to handle the end of life? | yes |
| Did you make your service available via the EOSC portal? | yes |
| Did you fix all known open issues/bugs? | yes |
| Is there a full documentation of your service available? | yes |
| Has the qualification test on EOSC been passed? | yes |

### TRL 9: Actual system proven in operational environment

| Question | Criterion fulfilles (yes/no) |
|---|---|
| Is your service operational in the EOSC? | yes |
| Is there continuous support for your service available? | yes |
| Did you finalise all related documentation? | yes |
| Does your software meet all specified requirements? | yes |
| Are all user stories successfully implemented? | yes |