EducaWood: a Semantic Web Application for Forestry Education

Guillermo Vega-Gorgojo [a,b,*], Juan I. Asensio-Pérez [a], Miguel L. Bote-Lorenzo [a], Pablo García-Zarza [a], José M. Giménez-García [a,b], Felipe Bravo [b] and Irene Ruano [b]
[a] *Group of Intelligent and Cooperative Systems, Universidad de Valladolid, Spain*
E-mails: guiveg@tel.uva.es, juaase@tel.uva.es, migbot@tel.uva.es, pablogz@gsic.uva.es, jm.gimenez.garcia@gsic.uva.es
[b] *SMART Ecosystems research group, iuFOR – Sustainable Forest Management Research Institute, Universidad de Valladolid, Spain*
E-mails: felipe.bravo@uva.es, irene.ruano@uva.es

Abstract. There are few applications available for educational purposes in the forestry domain. These applications have significant limitations, including not exploiting existing biodiversity datasets, lacking flexible and consistent use of domain concepts, and generating annotations that are not easily shareable or reusable by other applications. In this paper, we introduce EducaWood, a novel Semantic Web application designed for forestry education that overcomes these limitations by leveraging Linked Open Data (LOD). Users can easily create tree annotations through a web form that hides the complexity of Semantic Web technologies. These annotations adhere to the Simple Tree Annotation ontology and are saved in a triplestore, facilitating seamless sharing with other users and applications. Moreover, EducaWood offers scalable and efficient visualization of semantic tree data across various zoom levels on a map interface. Access to LOD is handled through a REST API that allows read and write operations over multiple data sources. An implementation of EducaWood has been successfully tested by more than 650 users, including real students and teachers in a pilot educational experience.

Keywords: semantic spatial data, access to LOD, semantic annotation, data visualization, semantic user interfaces, forestry education

1. Introduction

Environmental education emerges as an effective approach to bring the forestry world closer to society, encompassing both urban and rural communities. Multiple researchers argue that a better understanding of environmental sciences is achieved through active learning experiences grounded in real-life settings [1, 2]. Therefore, contextualized environmental education activities hold significant promise to better understand Earth's ecosystems and promote more responsible attitudes toward the conservation and sustainable use of our planet. Such contextualized environmental education activities might address a wide variety of forestry-related learning objectives such as: (i) learning to differentiate forest species, pests, or diseases through digital inventories; (ii) increasing interest in nature by discovering the diversity of trees in the environment and the biodiversity associated with them; (iii) integrating other subjects, such as mathematics, through

*Corresponding author. E-mail: guiveg@tel.uva.es.

the calculation of variables from inventory data (e.g., structural diversity indices, carbon sequestration, etc.); and (iv) fostering teamwork and outdoor learning. These activities can be adapted for different educational levels, from primary school (e.g., finding a specific tree species in the local environment) to secondary school (e.g., identifying tree species, measuring diameter and height, recognizing microhabitats) and high school (e.g., calculating the carbon sequestered by a forest). All these examples of learning activities shared a common element which is the need for the learners to locate and describe forestry-related physical entities (trees, leaves, timber, etc.) *in situ*, a process also known as **annotation**.

Some applications such as Integrate Tree Microhabitat [3], Observation.org [4], and iNaturalist [5] support the annotation of forestry-related physical entities. However, they have important limitations for environmental education that can be addressed with Semantic Web technologies. First, they do not exploit existing biodiversity datasets like tree inventories, land cover maps, or taxonomic classifications—Linked Open Data (LOD) and knowledge graphs are especially suitable for data integration. Second, they tend to make an inconsistent use of domain concepts (tree species, dead materials, decay stages, etc.), a difficulty that can be alleviated by employing ontologies. Third, they make an intensive use of environmental annotations that cannot be shared or reused by other applications. In this regard, the publication of environmental annotations as LOD enable their sharing and reuse, facilitating follow-up educational activities using the same or a different environmental education application.

In spite of the aforementioned potential benefits of Semantic Web technologies for the support of environmental annotations, their application to educational activities bring a new set of challenges:

CHALLENGE #1 Human-computer interaction with the Semantic Web is quite demanding [6–8]. Target users, particularly forestry teachers and students, do not usually know RDF or SPARQL and should be able to easily visualize and author environmental annotations while carrying out educational tasks.

CHALLENGE #2 Access to LOD is complex [9, 10], especially when dealing with read and write operations across multiple sources. This is the expected situation when different forestry datasets need to be exploited.

CHALLENGE #3 Forestry data tends to be very large and is geospatial by nature [11], requiring efficient approaches for visualizing semantic geospatial data.

In response to these challenges, we introduce EducaWood, a Semantic Web application designed for forestry education that showcases: (a) good practices in the design of web applications aimed at hiding the complexity of Semantic Web technologies to end users; (b) an easier approach to dealing with read and write access over multiple LOD endpoints; (c) efficient visualization of large geospatial environmental semantic datasets. More concretely, EducaWood features: (1) a web architecture aimed at supporting the description of physical entities (e.g., trees) by means of user-friendly web forms for authoring various types of annotations (e.g., location, tree status, taxon) while concealing the complexity of RDF, OWL, and SPARQL; (2) access to LOD through a CRAFTS API. CRAFTS [12] is an API generator for LOD that supports read and write operations across multiple endpoints, largely reducing the development effort when interacting with multiple datasets. Thanks to the CRAFTS API, EducaWood publishes environmental annotations in a triplestore, while consuming data from the Spanish National Forest Inventory [13] (IFN – *Inventario Forestal Nacional*), Wikidata [14], and DBpedia [15]; (3) an interactive map for seamless tree browsing and filtering by taxon. Since the application integrates more than one million trees from IFN, smart data management is critical to ensure performance across varying map resolutions and to minimize unnecessary data downloads.

Thanks to this use of Semantic Web technologies, EducaWood makes feasible the collaborative building of digital inventories of forestry-related physical entities. This way, EducaWood enables the design and enactment of complex, but also innovative learning scenarios in vocational and higher education such as the creation of a forest herbarium for botany courses or the compilation of a collection of pests and diseases for forestry pathology classes. Similarly, at the university level, EducaWood might support a challenging learning scenario which is explored in grater detail in this work: urban tree management.

Beyond the educational benefits for the forestry domain brought by EducaWood, the paper presents a set of contributions of special interest for the Semantic Web research community: (a) the Simple Tree Annotation (STA) ontology for structuring environmental annotations (mostly focused on trees); (b) a demonstrator of a web application architecture that hides the complexity of Semantic Web technologies from users, and that streamlines the read and write access to multiple LOD sources; (c) a novel mechanism for the scalable and efficient visualization of semantic tree data on a map.

The rest of the paper is organized as follows. Section 2 reviews interactive annotation applications in forestry education, as well as semantic approaches for the aforementioned challenges. Section 3 provides a technical description of EducaWood, including its requirements, its ontology, its architecture, its data sources, its main functionalities, and implementation details. Section 4 illustrates the functionality of the application and describes a forestry education scenario. Section 5 reports on the impact of EducaWood so far and includes a pilot study with forestry students. The paper ends with a discussion in Section 6 and the main conclusions and future research lines in Section 7.

2. Related work

Section 2.1 examines interactive annotation applications relevant to forestry education, identifying their main limitations—the lack of biodiversity dataset integration, inconsistent use of domain concepts, and siloed content—. While Semantic Web technologies offer a promising solution to these issues, they also introduce their own challenges. Therefore, we review existing approaches in three key areas: (1) human-computer interaction with the Semantic Web (Section 2.2), (2) simplifying access to Linked Open Data (LOD), particularly for read and write operations across multiple sources (Section 2.3), and (3) visualizing semantic geospatial data (Section 2.4)—corresponding to the challenges identified in the introduction.

2.1. Interactive annotation applications in forestry education

Currently, there are few applications available that could potentially be used for forestry education. One of them is the Integrate Tree Microhabitat application [3], developed by the European Forest Institute [16] to support training exercises for forest managers, inventory personnel, and other groups in identifying and describing tree microhabitats. However, its utility is limited to a series of training and demonstration plots known as "marteloscopes" where all trees are measured and geopositioned and where foresters can conduct virtual tree marking for training. This network of marteloscopes includes 224 sites across 25 European countries and four additional sites in Vietnam. Marteloscopes serve multifaceted purposes, including training for both students and professionals, public outreach, and research endeavors such as human behaviour concerning forests [17] or thinning effects on biodiversity conservation and socio-economic co-benefits [18]. However, a limitation of the Integrate Tree Microhabitat application is its inability to incorporate new data, a restriction imposed to uphold data integrity and facilitate consistent comparisons across different time frames and analyses.

Observation.org [4] is another application that may be used for forestry education. Nowadays, it serves as a global hub for citizen science where naturalists, citizen scientists, and biologists collaborate to gather, verify, and exchange biodiversity data. Users can create their own projects, located anywhere in the world, allowing them to generate biodiversity annotations through the website or the mobile application. Observation.org is more intensively used for animal projects (especially birds and insects) since it only includes a very general section for plants. More specifically, users can only annotate tree species, photos, and locations, but no further information such as dendrometric measures or tree status.

iNaturalist [5] is very similar to Observation.org, allowing users to gather, verify, and collaboratively exchange biodiversity data. iNaturalist has been used as a blended learning framework for biodiversity monitoring [19] and to engage the community in the organism identification in outdoor activities [20]. Again, iNaturalist annotations are limited to tree species, photos, and locations.

2.2. Human-computer interaction with the Semantic Web

As noted in Section 1, there are several challenges with the use of Semantic Web technologies in interactive applications spanning multiple domains, including forestry education. The first challenge entails facilitating human-computer interaction for stakeholders unfamiliar with Semantic Web technologies. Addressing this issue involves offering appropriate user interfaces with familiar conventions, thereby facilitating a transparent utilization of Semantic Web technologies while enabling seamless data analysis [21]. Several examples in the literature adopt this approach, although they are typically limited to LOD consumption without support for write operations. For example, the suite of Sampo portals [22] allows users to query and filter semantic Cultural Heritage data using form interfaces comprising textboxes and selectors, dynamically presenting data in tables, graphs, and maps. LOD4Culture [23] is another Semantic Web application from our previous work that follows this approach; it offers an interactive map and a table-based browser of Cultural Heritage entities sourced from Wikidata and DBpedia. Additionally, Linked Data browsers such as [24] aid users unfamiliar with Semantic technologies in visualizing semantic data.

Expanding upon this challenge, enabling end users to add or modify semantic data poses a significant hurdle. Andrews et al. [25] reviews the different annotation types employed in interactive annotation applications. Interestingly, the type of data to be recorded should influence the user interface design to introduce user input. For example, a clickable map interface may prove effective for capturing the coordinates of a place. In our previous work, we have successfully employed web forms to gather user data; this is the case of CHEST [26], where teachers can easily create spatial objects and learning tasks, which are then transparently saved as LOD. Notably, Wikidata is acknowledged as the leading open knowledge base in the world [27], also leveraging web forms for user inputs.

2.3. Access to Linked Open Data

While the availability of LOD and knowledge graphs has grown across all domains, access to LOD is quite demanding even for knowledge engineers (refer to CHALLENGE #2 in the introduction). Beyond expertise with RDF, OWL, and SPARQL, access to LOD requires familiarity with the ontologies used and domain knowledge. To address this challenge, the Semantic Web community has proposed different approaches. Some define an HTTP interface over Linked Data such as Linked Data Fragments [9] which offers a limited API for efficient consumption of Linked Data, although write access is not supported. Other approaches define new serializations of Linked Data and SPARQL results to JSON like JSON-LD [28] and SPARQL transformer [29]; unfortunately, they do not support SPARQL query formulation, a much more demanding task for web developers than output transformations.

Since web developers typically employ REST APIs and JSON as interchange format, it is therefore desirable to follow these conventions when accessing LOD. As a result, there is a number of proposals that support the creation of REST APIs on top of triplestores: RAMOSE [30], R4R [31], OBA [32], grlc [33], BASIL [34], and CRAFTS [12]. RAMOSE, grlc, and BASIL essentially allow the provision of APIs that encapsulate parametrized SPARQL queries. R4R, OBA, and CRAFTS also allow the exposition of RDF resources over an API. Only OBA and CRAFTS support write operations, although partial updates through HTTP PATCH [35] are only available in CRAFTS. All of these API generators provide one-to-one mappings between API calls and SPARQL queries. CRAFTS, on the other hand, uses one-to-many mappings, offering greater control over data exposure. Lastly, CRAFTS is the only API generator that can work with multiple endpoints from a single API. Vega-Gorgojo [12] includes a thorough comparison of API generators over Linked Data.

2.4. Visualizing semantic geospatial data

Lastly, some application domains like forestry heavily rely on geospatial data, which brings their own set of challenges [11]. Here, we particularly focus on the visualization of semantic geospatial data (CHALLENGE #3), requiring effective interfaces that ease access and analysis. We can find several proposals for visualizing geospatial Linked Data that are targeted to Semantic Web experts. This is the case of GeoYASGUI [36], a GeoSPARQL editor that provides a map visualizer of result sets. Sextant [37] is an advanced visualization application that can combine spatial data from several endpoints, although it still requires knowledge of SPARQL in order to use it.

Visualization of semantic spatial data should not be limited to Semantic Web experts. Given the ubiquitous use of map applications, map-based interfaces seem a suitable approach for lay users. However, spatial data is inherently complex and tends to be quite large, needing thoughtful design decisions when presenting data directly on the map. In this regard, Gaigg [38] proposes techniques for dealing with large amounts of spatial data, including data filtering, layering, and clustering. Following these principles, our previous work introduced Forest Explorer [39], a visualization application for official forestry data, including national forest inventories and land cover maps from Spain and Portugal. The application is read-only, meaning users cannot modify the underlying dataset. It handles various feature types—administrative regions, land cover patches, inventory plots, and sampled trees—which are displayed on the map depending on the zoom level. At lower zoom levels the application provides aggregated forest information within administrative regions, gradually revealing more detailed data such as land cover maps, national inventory plots, and sampled trees with their measurements as the zoom level increases. Forest Explorer leverages a semantic dataset that integrates multiple forestry data sources, as described in [40]. The application relies on SPARQL query templates for data retrieval, as explained in [39]. Forest Explorer is designed for forestry experts, data journalists, and the general public, offering an accessible yet powerful way to explore forest-related data.

There exist other seldom visualizers of geospatial LOD designed for non-Semantic Web experts, such as LinkedGeoData [41] and Map4rdf [42]. LinkedGeoData is a dedicated visualization tool for OpenStreetMap data (transformed to adhere to Linked Data principles), while Map4rdf is a browsing tool of geospatial RDF datasets that uses a faceted interface to control the information to display. However, the current status of these tools appears uncertain.

In summary, several interactive annotation applications can be applied to forestry education. However, they present important limitations, such as the lack of integration with biodiversity datasets, the inconsistent use of domain concepts, and the confinement of content in data silos. Semantic Web technologies are particularly well suited to address these issues, although they also introduce specific challenges. The first concerns human–computer interaction with Semantic Web data. Proposed solutions include innovative user interfaces, which are typically restricted to LOD consumption; notable exceptions, such as Wikidata and CHEST, enable end users to perform write operations on semantic data through web forms. The second challenge relates to accessing LOD and knowledge graphs. Many approaches support the creation of REST APIs over triplestores, with OBA and CRAFTS being among the most versatile. The third challenge involves the visualization of semantic geospatial data. While some solutions require proficiency in SPARQL, map-based interfaces appear to be a suitable alternative for non-experts. In all cases, particular attention must be paid to the volume and complexity of spatial data.

3. Design of EducaWood

In this paper we propose EducaWood, a new Semantic Web application designed for forestry education, addressing the limitations of existing interactive annotation tools. Given the challenges of human-computer interaction with the Semantic Web, we employ map interfaces and web forms to enable content authoring without requiring users to interact directly with RDF, OWL, or SPARQL. Since our approach involves both reading and writing across multiple LOD sources, we leverage a CRAFTS API to streamline data access. For effective visualization of semantic geospatial data, we implement efficient strategies for retrieving and rendering tree data at different zoom levels.

The primary objective of EducaWood is to support learning activities based on the social annotation of trees, while also allowing the exploration of forestry information within specific regions of interest. Tree annotations can be of different types and are published as LOD. The application exploits existing semantic

Table 1

Prefixes and namespaces employed in this paper.

Prefix	Namespace	Purpose
`sta`	http://educawood.gsic.uva.es/sta/ontology/	terms of the STA ontology
`dc`	http://purl.org/dc/terms/	importing `creator` and `created` from Dublin Core
`foaf`	http://xmlns.com/foaf/0.1/	importing `Person`, `Image`, and `nick` from FOAF
`w3cgeo`	http://www.w3.org/2003/01/geo/wgs84_pos#	importing `SpatialThing`, `lat`, `long` from Basic Geo
`ifn`	http://crossforest.eu/ifn/ontology/	importing class `Taxon` from IFN
`ifntx`	https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/	importing taxons from IFN
`user`	http://educawood.gsic.uva.es/user/	creation of users in EducaWood
`tree`	http://educawood.gsic.uva.es/tree/	creation of tree entities
`posann`	http://educawood.gsic.uva.es/posann/	creation of tree position annotations
`spann`	http://educawood.gsic.uva.es/spann/	creation of tree species annotations
`treestann`	http://educawood.gsic.uva.es/treestann/	creation of tree status annotations
`heightann`	http://educawood.gsic.uva.es/heightann/	creation of tree height annotations
`diamann`	http://educawood.gsic.uva.es/diamann/	creation of tree diameter annotations
`observann`	http://educawood.gsic.uva.es/observann/	creation of tree observation annotations
`img`	http://educawood.gsic.uva.es/img/	creation of tree images
`imgann`	http://educawood.gsic.uva.es/imgann/	creation of tree image annotations

datasets from Spain that we have released as LOD in our previous work [39, 40], specifically the Spanish National Forest Inventory (IFN – *Inventario Forestal Nacional*). Moreover, EducaWood also consumes third-party semantic data such as tree species taxonomic data from Wikidata and DBpedia. Throughout this paper, we use the prefixes and namespaces listed in Table 1. The domain educawood.gsic.uva.es is consistently used for both the ontology and the dataset, ensuring that all URIs remain under the authoritative source of EducaWood. This design choice follows common practice in the Semantic Web community, as illustrated by well-known examples such as Wikidata and DBpedia.

3.1. Requirements

We have carried out a requirement analysis for EducaWood using as sources our own experience in the field, the gaps found in the literature (see Section 2), and the feedback collected from users when testing early prototypes of the application. Table 2 summarizes the main requirements, organized as functional (FRx) and non-functional (NFRx). The forestry experts in our team first identified the core functionalities: (i) collaborative authoring of tree annotations of different types, (ii) visualization of these annotations, and (iii) interactive exploration of trees through a map. These functionalities are captured by requirements FR0–4 in Table 2 and represent the minimal set needed for a viable prototype. Subsequently, the full team refined the requirements by adding desirable, though not critical, features related to map visualization and geospatial data (FR5–8 in Table 2).

The first group of functional requirements (FR0–3) addresses CHALLENGE #1: *Supporting semantic annotations by non Semantic Web experts*. EducaWood should provide comprehensive visualizations of the tree annotations available (FR0); tree annotations can be created by registered users in the application (FR1), using an appropriate web form; trees can be described by annotations of different types (FR2), a location is always required, while the rest of annotation types (image, dendrometric measures, tree status, etc.) are optional; annotations can be made incrementally by different users, so the application has to handle multi-author tree annotations and deletions (FR3).

The second group of functional requirements (FR4–8) corresponds to CHALLENGE #3: *Visualizing and managing large geospatial datasets*. FR4 refers to one of the main functionalities, the exploration of

Table 2
Requirements for EducaWood: functional (FRx) and non-functional (NFRx), and connection to the identified Semantic Web challenges.

ID	Requirement	Challenge
FR0	Provide comprehensive visualizations of tree annotations	Supporting semantic annotations by non Semantic Web experts
FR1	Allow the creation of tree annotations by registered users hiding RDF and SPARQL querying	
FR2	Support different types of annotations (location, tree status, taxon, height, diameter, image, and observation)	
FR3	Handle multi-author tree annotations and deletions	
FR4	World-wide exploration of trees through an interactive map	Visualizing and managing large geospatial datasets
FR5	Map view adaptable to different zoom levels	
FR6	Allow filtering of trees by taxon	
FR7	Include tree data from forest inventories	
FR8	Support downloads of tree data (at least in CSV format)	
NFR0	Portability (mobile phones, tablets, and desktop computers)	
NFR1	Provide mechanisms to keep latency low	
NFR2	Localized to English and Spanish	

trees through an interactive map; the scope should be worldwide, while the map view has to be adaptable to different zoom levels (FR5), so as to facilitate the exploration of small areas—showing markers for trees—but also large ones, providing appropriate aggregation mechanisms to avoid cluttering the view with too many markers; as species information is quite relevant in forest education, the application should provide a taxon filtering mechanism (FR6); the map view should also display trees from forest inventories available as LOD (FR7), specifically, EducaWood will integrate IFN data as this source contains reliable and relevant information of native trees (although limited to Spain); tree data should also be downloadable at least in CSV format (FR8) to allow the realization of different types of analysis for forestry education such as allometric equations fitting, tree mingling analysis, or environmental effect on species distributions. The remaining CHALLENGE #2, *Access to LOD*, is addressed by the architecture of the application and is not, per se, perceived by the EducaWood users directly).

In addition to functional requirements, we defined key non-functional requirements. Portability (NFR0) is critical for ensuring that EducaWood can be used by students and teachers, with mobile devices as the primary target, and tablets and desktop computers as secondary. Low latency (NFR1) is also necessary for maintaining a responsive and usable application. At minimum, the application has to support English and Spanish (NFR2), with the possibility of adding further languages in the future. The following subsections describe in detail how the above requirements have been addressed in the design and implementation of EducaWood.

3.2. Simple Tree Annotation ontology

The Simple Tree Annotation (STA) ontology[1] is the basis for the main functionalities of EducaWood. It has been conceived for describing trees (FR0) and supporting different types of annotations (FR2); it also allows multi-author annotations (FR3), as well as a mechanism for conflict resolution. The ontology is relatively small, consisting of 412 axioms, 36 classes, 16 object properties, 10 data properties, and 8 annotation properties. Its logical expressivity is *RRESTRH(D)*, which reflects the use of existential restrictions, role hierarchies, property chains, transitive properties, and datatypes, while avoiding constructs such as nominals or qualified cardinalities.

[1]STA is available in GitHub at https://github.com/guiveg/STA.

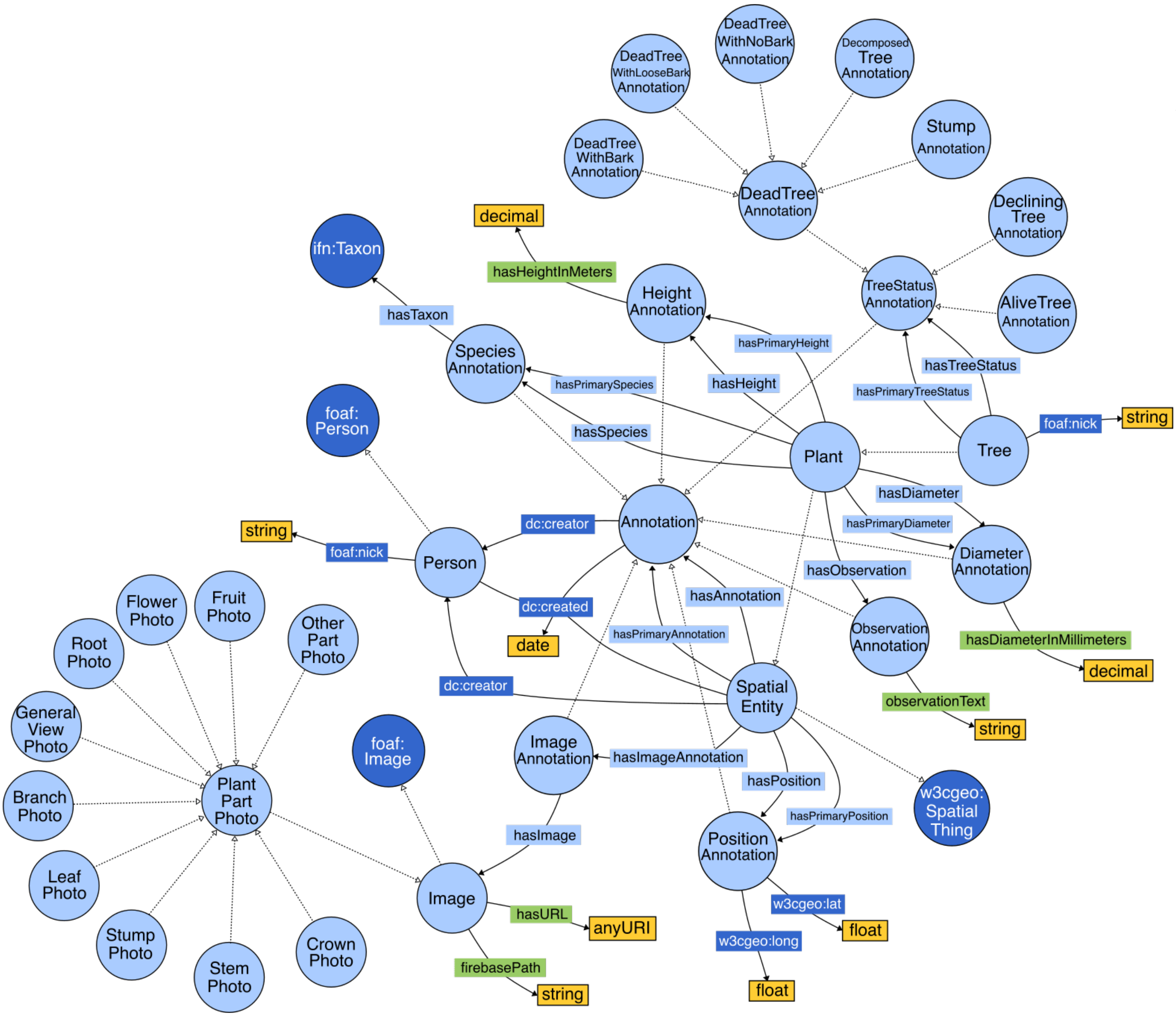

Figure 1. Visualization of the Simple Tree Annotation ontology with VOWL [46].

STA reuses terms from Basic Geo [43], FOAF [44], IFN [40], and Dublin Core [45]. Basic Geo is employed to represent spatially located things (such as trees) with latitude and longitude in WGS84. FOAF provides concepts for *Person* (the annotators in STA) and *Image* (a type of annotation). For tree species, STA relies on the taxonomy defined in IFN. Dublin Core is used to specify metadata such as the creator and date of annotations. Figure 1 depicts STA graphically, while Table 3 organizes the ontology elements by category and type, indicating the requirements they address.

The main classes of STA are `sta:SpatialEntity`, `sta:Annotation`, `sta:Person`, and `sta:Image`. `sta:SpatialEntity` is a specialization of `w3cgeo:SpatialThing` [43], denoting anything with spatial extent. From `sta:SpatialEntity`, we create subclasses `sta:Plant` and `sta:Tree` for the forestry domain. We borrow `foaf:Person` and `foaf:Image` from the FOAF ontology [44] in order to create our own specialized concepts, `sta:Person` and `sta:Image`. `sta:Annotation` is the main class for anno-

Table 3
Elements of the Simple Tree Ontology, organized by category (with mapping to requirements) and type.

Category	Type	Elements
Main elements (FR0–1)	classes	`sta:SpatialEntity`, `sta:Tree`, `sta:Plant`, `sta:Person`, `sta:Annotation`, `sta:Image`
	object properties	`sta:hasAnnotation`
Location annotations (FR0–2, FR4)	classes	`sta:PositionAnnotation`, `w3cgeo:SpatialThing`
	object properties	`sta:hasPositionAnnotation`
	data properties	`w3cgeo:lat`, `w3cgeo:long`
Species annotations (FR0–2, FR6)	classes	`sta:SpeciesAnnotation`, `ifn:Taxon`
	object properties	`sta:hasSpeciesAnnotation`, `sta:hasTaxon`
Tree status annotations (FR0–2)	classes	`sta:TreeStatusAnnotation`, `sta:AliveTreeAnnotation`, `sta:DecliningTreeAnnotation`, `sta:DeadTreeAnnotation`, `sta:DeadTreeWithBarkAnnotation`, `sta:DeadTreeWithLooseBarkAnnotation`, `sta:DeadTreeWithNoBarkAnnotation`, `sta:DecomposedTreeAnnotation`, `sta:StumpAnnotation`
	object properties	`sta:hasTreeStatusAnnotation`
Height annotations (FR0–2)	classes	`sta:HeightAnnotation`
	object properties	`sta:hasHeightAnnotation`
	data properties	`sta:hasHeightInMeters`
Diameter annotations (FR0–2)	classes	`sta:DiameterAnnotation`
	object properties	`sta:hasDiameterAnnotation`
	data properties	`sta:hasDiameterInMillimeters`
Image annotations (FR0–2)	classes	`sta:ImageAnnotation`, `sta:PlantPartPhoto`, `sta:LeafPhoto`, `sta:FruitPhoto`, `sta:FlowerPhoto`, `sta:StemPhoto`, `sta:BranchPhoto`, `sta:CrownPhoto`, `sta:RootPhoto`, `sta:StumpPhoto`, `sta:GeneralViewPhoto`, `sta:OtherPartPhoto`, `foaf:Image`
	object properties	`sta:hasImageAnnotation`, `sta:hasImage`
	data properties	`sta:imageURL`, `sta:firebasePath`
Observation annotations (FR0–2)	classes	`sta:ObservationAnnotation`
	object properties	`sta:hasObservationAnnotation`
	data properties	`sta:observationText`
Multi-author annotations (FR0–3)	classes	`foaf:Person`
	object properties	`sta:hasPrimaryAnnotation`, `sta:hasPrimaryPosition`, `sta:hasPrimaryDiameter`, `sta:hasPrimaryHeight`, `sta:hasPrimarySpecies`, `sta:hasPrimaryTreeStatus`
	data properties	`dc:creator`, `dc:created`, `foaf:nick`

tations[2], although it is not purposed for direct instantiation; instead, we have created the special-

[2] `sta:Annotation` is a class in STA and it is unrelated to `owl:AnnotationProperty`.

izations `sta:PositionAnnotation`, `sta:ImageAnnotation`, `sta:SpeciesAnnotation`, `sta:TreeStatusAnnotation`, `sta:HeightAnnotation`, `sta:DiameterAnnotation`, and `sta:ObservationAnnotation`.

A `sta:Tree` (or more generally a `sta:SpatialEntity`) can have an arbitrary number of annotations. `sta:hasAnnotation` is the object property for linking a `sta:SpatialEntity` (domain) to a `sta:Annotation` (range). Since a tree may have multiple and possibly contradictory annotations of the same type, we also define the object property `sta:hasPrimaryAnnotation` for conflict resolution. A tree should only have a primary annotation of a specific type, e.g. diameter, although it may have multiple diameter annotations (possibly measured at different times and by different people). We then create subproperties of `sta:hasAnnotation` and `sta:hasPrimaryAnnotation` for the different types of annotations, e.g. `sta:hasHeight` and `sta:hasPrimaryHeight`. We have not defined primary annotations for `sta:ImageAnnotation` and `sta:ObservationAnnotation`, as they are naturally multivalued for a single tree.

We have created additional terms in STA to support the different types of annotations: we reuse `w3cgeo:lat` and `w3cgeo:long` to determine the WGS84 coordinates of a `sta:PositionAnnotation` (corresponding to a point); for `sta:ImageAnnotation` we define data properties `sta:firebasePath`[3] and `sta:hasURL` with domain `sta:Image` and we include a taxonomy of part plants for describing tree images (`sta:PlantPartPhoto` and subclasses); in the case of `sta:SpeciesAnnotation` we define the object property `sta:hasTaxon` and then reuse the taxonomy of species employed in IFN (subclasses of `ifn:Taxon`); we create specializations of `sta:TreeStatusAnnotation` such as `sta:AliveTreeAnnotation` or `sta:DeadTreeAnnotation`[4]; `sta:HeightAnnotation` uses the data property `sta:hasHeightInMeters`; `sta:DiameterAnnotation` employs the data property `sta:hasDiameterInMillimeters`; and `sta:ObservationAnnotation` makes use of the data property `sta:observationText`. We use `dc:creator` and `dc:created` metadata annotations to define the creator and the datetime of any instance of `sta:Annotation` or `sta:SpatialEntity`. Finally, we employ `foaf:nick` to give nicknames to people and trees.

3.3. Application architecture

EducaWood is designed as a web application with an architecture aimed at facilitating its users to visualize and carry out semantic annotations without needing technical expertise on Semantic Web technologies (refer to CHALLENGE #1 in the introduction). The web architecture of EducaWood can be described by the routes shown in Table 4.

R0 is a landing page that presents the application and includes a link to route R1, corresponding to the interactive map functionality. R1 includes a required query parameter, `loc`, that defines a specific position and zoom level with the format `LAT,LONG,ZOOMz`[5]; `taxon` can be set to filter the trees shown in the map (FR6), e.g. `ifntx:Species23` is the IRI of *Pinus pinea* in the IFN dataset; `esri` is a boolean query parameter for using the satellite base map provided by Esri [49]; and `ifn` can be activated to show the trees from the IFN dataset (FR7). In this way, R1 can be used to specify the location of any place in the world, with a specific zoom level, and with optional taxon filter, satellite base map, and display of IFN data, such as the route /map?loc=41.751849,-4.585419,10z&ifn=true&esri=true&taxon=ifntx:Species23[6].

New trees are created with a web form available at route R2 (FR1); query parameter `loc` has the same format as in route R1. Route R3 is used to provide functionalities FR0 and FR2, allowing the visualization of the annotations of a tree `treeId` and providing controls for creating and removing annotations. The management of trees and their annotations involve write operations that are restricted to registered

[3]This property is used to reference the path of an image if using Cloud Storage for Firebase.

[4]Dead trees and down woody materials annotations follow the scale proposed by Maser et al. [47] and consolidated by Hunter [48] for managing forest ecosystems to sustain biodiversity. Dead materials are key resources for many species and act as biodiversity harbors in the forest matrix.

[5]`LAT` and `LONG` assume the WGS 84 datum. `ZOOM` represents the zoom level in powers of two, typically ranging from 0 (the whole world is entirely represented in a tile) to 20 (∼1 trillion tiles are needed to show the entire world).

[6]The rendering of this route in EducaWood is shown in Figure 3(c).

Table 4
Routes exposed in EducaWood. Query parameters marked with * are required.

ID	Path	Query parameters	Description
R0	/	—	Landing page of EducaWood
R1	/map	loc*, taxon, esri, ifn	Map centered in loc with optional taxon filter, optional esri satellite layer, and optional display of ifn trees
R2	/newtree	loc*	Creation form of a new tree positioned in loc
R3	/tree/{treeId}	—	Page of tree treeId
R4	/lasttrees	pe, showann, pae	Last created trees in EducaWood paginated with optional pe parameter, or last annotations if showann is true, paginated with optional pae parameter
R5	/user/{userId}	pe, showann, pae	Page of user userId with their last created trees paginated with optional pe parameter, or their last annotations if showann is true, paginated with optional pae parameter

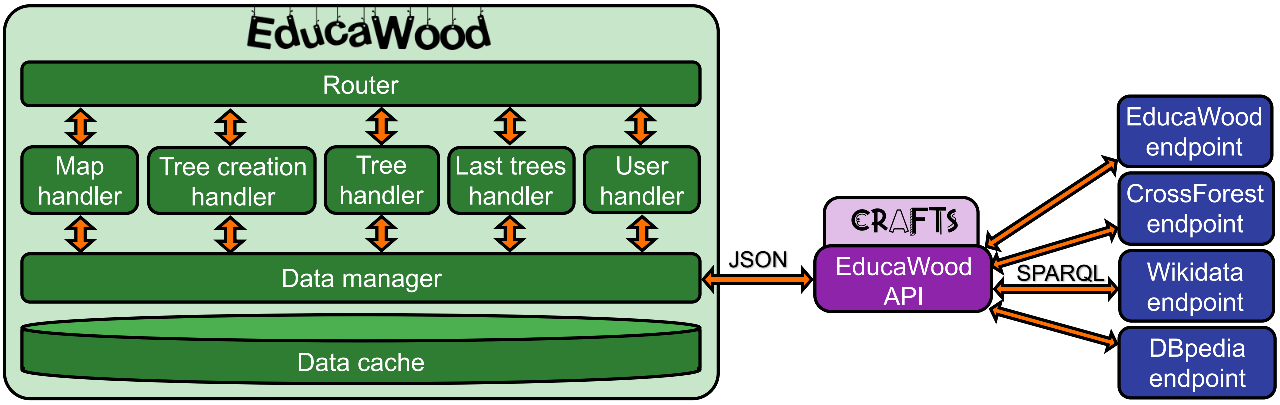

Figure 2. Architecture of EducaWood.

users (FR3). Route R4 is used to display the last created trees in EducaWood; the optional query parameters are employed to switch from tree creations to annotations (`showann`), while `pe` and `pae` are used for pagination. Route R5 defines user pages employing the path parameter `userId` for identifying the user; tree creations and annotations are also displayed in user pages and for this purpose we use the same query parameters as in route R4.

In order to satisfy the non functional requirements of portability (NFR0) and low latency (NFR1), EducaWood has been designed as a single-page application (SPA). SPAs are web applications that initially load a single web document and then update their body content with data from the server, thus avoiding full-page reloads. SPAs tend to provide performance gains and a more dynamic experience [50].

The architecture of EducaWood is graphically depicted in Figure 2. The *Router* component is in charge of performing client-side routing; if the browser URL changes, the *Router* detects it and checks its validity. A valid URL has to follow one of the routes in Table 4. The *Router* dispatches R1-compliant URLs to the *Map handler*, R2-compliant URLs to the *Tree handler*, R3-compliant URLs to the *Tree creation handler*, R4-compliant URLs to the *Last trees handler*, and R5-compliant URLs to the *User handler*. A *Handler* updates the view according to the incoming request, i.e. the refreshed URL, and provides appropriate controls for user interaction. The *Handlers* will make requests to the *Data manager* to carry out their tasks. This latter component centralizes data access by making calls to the *EducaWood API*. Responses from the API are locally stored in the *Data cache* to minimize future exchanges; indeed, the *Data manager* first checks the *Data cache* and in case of a miss will make a call to the API.

In order to address the CHALLENGE #2 described in the introduction (dealing with complexity associated with the access to multiple LOD sources), the *EducaWood API* is built with Configurable REST APIs For Triple Stores (CRAFTS) [12]. The case of EducaWood is quite suitable for using CRAFTS given the need of a highly flexible data access with read and write operations over four different endpoints. CRAFTS provides a simple REST API exposing RDF resources and parametrized SPARQL queries, using JSON as interchange format, and caching SPARQL queries from the source endpoints. In other words, the use of a CRAFTS-based API serves to reduce the complexity of creating a LOD-based application such as EducaWood. This complexity is transferred to the creation of a configuration file that is used in a CRAFTS site to translate REST calls into SPARQL queries. Vega-Gorgojo [12] describes the elements of a CRAFTS configuration file, while the OpenAPI specification of CRAFTS is browsable (and actionable) at https://crafts.gsic.uva.es/docs/.

Appendix A depicts the configuration file of the *EducaWood API*—essentially a JSON object with a collection of keys and values. `apiId` contains the identifier of the API, `educawood`. `endpoints` includes the information for accessing the four endpoints in EducaWood[7]. `model` contains an array of the different RDF resources exposed by the API; each one defines mappings of RDF data to JSON by referring to datatype properties (`dprops`), object properties (`oprops`), and class membership (`types`). `queryTemplates` list a number of parametrized SPARQL queries. Table 5 includes a sample of the API calls used in EducaWood.

3.4. Data sources

As shown in Figure 2, EducaWood integrates data from four distinct sources (EducaWood, CrossForest, Wikidata, and DBpedia). We have established a dedicated triplestore, referred to as *EducaWood*[8], to store all tree annotations generated within the application. This dataset is continuously updated with user contributions, which requires write access via SPARQL Update [51]. While the CRAFTS API includes the necessary credentials for SPARQL Update (not explicitly shown in Appendix A), *EducaWood* also incorporates the STA ontology, as detailed in Section 3.2.

The *CrossForest* dataset was created as part of our involvement in the European CrossForest project [52]. This comprehensive resource encompasses national forest inventories and land cover maps from Spain and Portugal. A detailed description of the ontologies we designed to define the required forestry terminology, as well as the process of transforming the source databases into Linked Open Data, is provided in [40]. Within EducaWood, the *CrossForest* dataset serves two primary purposes: (1) providing access to tree data from the Spanish National Forest Inventory (IFN), which includes approximately 1.4 million native trees; and (2) serving the IFN species taxonomy to support tree species annotations. Each tree species in *CrossForest* is annotated with scientific and multilingual common names, linked to higher taxonomic ranks (e.g., genus, family, class), and aligned with external knowledge bases such as Wikidata and DBpedia.

The remaining data sources, *Wikidata* and *DBpedia*, are well-established within the Semantic Web community. While their role in EducaWood is not central, they exemplify the advantages of Semantic Web-enabled data integration. By leveraging the alignments between *CrossForest* and these sources, EducaWood retrieves additional information such as species images and links to Wikispecies [53], Wikipedia [54], and the Global Biodiversity Information Facility (GBIF) [55] from Wikidata, as well as multilingual textual descriptions from DBpedia.

3.5. Rendering maps

EducaWood addresses CHALLENGE #3 (efficient visualization of large geospatial semantic datasets) using a novel approach for rendering maps. At launch time, the *Data manager* prepares the taxonomy of tree species in a bootstrapping routine by sending several C0 and C1 calls (see Table 5). C0 serves to obtain the hierarchy of taxons that derive from the ancestor class *Gymnospermae* (`ifntx:Class2`) by using the

[7]The *EducaWood* endpoint includes credentials for using SPARQL Update, although not shown in Appendix A
[8]Endpoint URL https://semanticforest.gsic.uva.es/sparql and graph IRI http://educawood.gsic.uva.es.

Table 5
Sample calls to the EducaWood API.

ID	Op.	Route	Description
Bootstrapping			
C0	`GET`	/apis/educawood/query?id=subclasses&ancestor=https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Class2	Retrieve the subclass relations between pairs of classes derived from the ancestor class *Gymnospermae* (`ifntx:Class2`)
C1	`GET`	/apis/educawood/resources?id=Species&ns=https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/&nspref=ifntx&iris=ifntx:Class2&iris=ifntx:Genus211&iris=ifntx:Species23	Retrieve the representations of class *Gymnospermae* (`ifntx:Class2`), genus *Pinus* (`ifntx:Genus211`), and species *Pinus Pinea* (`ifntx:Species23`)
Map exploration			
C2	`GET`	/apis/educawood/query?id=educatreesinbox&lngwest=-4.6875&lngeast=-4.5&latnorth=42&latsouth=41.8125&limit=1&offset=1000	Retrieve tree #1001 within a specified map cell
C3	`GET`	/apis/educawood/query?id=counteducatreesinbox&lngwest=-4.6875&lngeast=-4.5&latnorth=42&latsouth=41.8125	Count the number of trees within a specified map cell
C4	`GET`	/apis/educawood/query?id=educatreesinbox&species=https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Species23&lngwest=-4.6875&lngeast=-4.5&latnorth=42&latsouth=41.8125	Retrieve the trees of species *Pinus Pinea* (`ifntx:Species23`) within a specified map cell
C5	`GET`	/apis/educawood/resources?id=BasicEducaTree&ns=http://educawood.gsic.uva.es/tree/&nspref=tree&iris=tree:x3tbrTrWLn1C&iris=tree:9bq9FeHxsgfS	Retrieve the basic representations of trees `tree:x3tbrTrWLn1C` and `tree:9bq9FeHxsgfS`
Tree view			
C6	`GET`	/apis/educawood/resource?id=EducaTree&iri=http://educawood.gsic.uva.es/tree/Neik7P0woiDY	Retrieve the representation of `tree:Neik7P0woiDY`
Tree management			
C7	`PUT`	/apis/educawood/resource?id=EducaTree&iri=http://educawood.gsic.uva.es/tree/yUhX0LzFP-57	Create `tree:yUhX0LzFP-57` with the tree representation enclosed to this call
C8	`PATCH`	/apis/educawood/resource?id=EducaTree&iri=http://educawood.gsic.uva.es/tree/yUhX0LzFP-57	Update `tree:yUhX0LzFP-57` with the patch representation enclosed to this call
C9	`DELETE`	/apis/educawood/resource?id=SpeciesAnnotation&iri=http://educawood.gsic.uva.es/spann/JXbiTiPApexo	Delete species annotation `spann:JXbiTiPApexo`
C10	`DELETE`	/apis/educawood/resource?id=EducaTree&iri=http://educawood.gsic.uva.es/tree/yUhX0LzFP-57	Delete `tree:yUhX0LzFP-57`

`subclasses` template query included in the *EducaWood API*; a trivial replacement of `ifntx:Class2` with `ifntx:Class1` serves to gather the hierarchy of taxons that derive from the ancestor class *Angiospermae*[9]. C1 is then used to retrieve representations of the different taxons found, using the RDF resource `Species` from the `model` in Appendix A[10]; while C1 includes three taxons for illustration, the *Data manager* will make C1-like calls packing a larger number of taxons so as to limit exchanges with the *EducaWood API*.

The *Map handler* is in charge of supporting the map navigation functionality (FR4), showing the trees on the map view using LOD as source (*EducaWood* and *CrossForest* endpoints). An interactive map is used for this purpose, supporting typical panning and zooming operations that are naturally supported for both point-and-click and touchscreen interfaces. The *Map handler* carries out this task by handling R1-compliant routes. Upon an incoming request, the map view is centered in the location extracted from the browser URL, and with the indicated zoom level. Then, a rectangular grid, centered in point `LAT=0, LONG=0`, is employed to fill the map view; cell side is configured to 12º for zoom level 4 and scaled to other zoom levels[11]. In this way, a cell in unambiguously identified by its x and y indexes at zoom level z, independently of the device. Screen size and resolution will determine which cells are required to fill the

[9]Trees are seed plants belonging to either the *Gymnospermae* clade, predominantly composed of conifers, or the *Angiospermae* clade, which consists of flowering plants.

[10]The RDF resource `Species` in Appendix A combines data from *EducaWood*, *Wikidata*, and *DBpedia* endpoints. This is illustrated in the visualization of *Pinus pinea* in Figure 4(b).

[11]We use powers of two for scaling: a cell at zoom level z corresponds to four cells at zoom level $z+1$.

map view; a mobile phone will typically use 20–30 cells, while a desktop computer with a 21" screen can easily employ 90–120. Nonetheless, a location at a given zoom level will always correspond to the same grid cell.

A grid cell is the unit of work to display trees on the map. The *Map handler* begins by identifying the cells corresponding to the map view and subsequently sends individual data request for each of these cells to the *Data manager*. When a cell request is received, the *Data manager* gathers only the **essential** data required for display, rather than retrieving **all** available data (FR5). This is particularly relevant at low zoom levels, where a single cell may encompass numerous trees, potentially reaching into the tens or even hundreds of thousands[12]. Thus, the *Data manager* will follow the following procedure for cell requests:

1. Request tree #1001 of the map cell (call C2 in Table 5)
   (finish if tree #1001 exists, otherwise continue)
2. Count the number of trees within the map cell (C3)
   (finish if the count is 0 or greater than 100, otherwise continue)
3. Discover the trees within the map cell (C4)
4. Obtain basic representations of the trees found in step #3 (C5)

Step #1 serves to assess whether there is a large number of trees within a cell, i.e. more than one thousand, without requiring to count them all (an expensive operation in SPARQL). If this is not the case, the actual count is obtained in step #2. In the range of 1–100 trees, it makes sense to display individual markers, so step #3 serves to discover the IRIs of the trees and then step #4 to retrieve their basic representations. This procedure illustrates the case of the *EducaWood* endpoint; if the IFN dataset is selected (parameter `ifn` in route R1), a similar procedure will be carried out with the *CrossForest* endpoint using alternative query templates and RDF resources, e.g. `treesinbox` instead of `educatreesinbox` (check Appendix A for more details). Taxon filtering (FR6) is also supported in cell requests: the sample C4 call in Table 5 contains a *Pinus pinea* filter, while all the query templates employed for map exploration include an optional taxon filter parameter. It is also relevant that responses from the *EducaWood API* are always cached, thus allowing to reuse previous cell results when requested again. Moreover, the *Data manager* exploits the *Data cache* to derive new information without making further API calls, as in the following cases:

- If cell C_i has more than 1K trees and cell C_j contains cell $C_i \Rightarrow$ cell C_j has more than 1K trees.
- If cell C_i has 0 trees and cell C_j is contained in cell $C_i \Rightarrow$ cell C_j has 0 trees.
- If cell C_i has a taxon filter Tx_i and more than 1K trees, and cell C_j contains cell C_i and has no taxon filter $\Rightarrow$ cell C_j has more than 1K trees.
- If cell C_i includes a list of tree representations Lt_i and cell C_j is contained in cell $C_i \Rightarrow$ the subset of tree representations of cell C_j can be derived from Lt_i.
- Etc.

Once cell data is retrieved, the *Map handler* can proceed with rendering. Figure 3 shows the map interface of EducaWood in different places of the world (requirement R4 in Table 2) at different resolutions (FR5). Snapshot (c) illustrates tree filtering by taxon (FR6), in this case `ifntx:Species23` (*Pinus pinea*). Trees from the IFN dataset are displayed in snapshots (b, c), thus illustrating requirement FR7. The map controls in Figure 3 include a download button in the right panel for supporting FR8.

3.6. Viewing and creating tree annotations

Here we focus on CHALLENGE #1 (support of semantic annotations by non-experts on Semantic Web technologies) by providing a holistic view of how the EducaWood components work together when

[12]This scenario is highly plausible with the IFN dataset, considering it contains approximately 1.4 million trees within the Spanish territory.

Figure 3. Snapshots of the map interface of EducaWood. (a) Route /map?loc=41.843949,0.548121,6z, corresponding to a large area in South-West Europe. (b) Route /map?loc=41.752276,-4.585411,10z&ifn=true, focused on a mid–size area in North Spain. (c) Route /map?loc=41.751849,-4.585419,10z&ifn=true&esri=true&taxon=ifntx:Species23; this is the same area as (b), but restricted to *Pinus pinea* species and using the Esri satellite base map. (d) Route /map?loc=41.986754,-4.516886,18z&esri=true, showing a tiny urban area with numerous tree markers.

viewing and creating tree annotations. We outline the sequence of significant API calls, illustrating how the requirements in Section 3.1 are satisfied.

The creation of a new tree (FR1) uses an R2 route. The *Tree creation handler* is responsible for rendering a tree creation form with various fields, each corresponding to different types of annotations in the STA ontology (see Section 3.2). Figure 4(a) shows the tree creation form; it employs different widgets to facilitate content authoring. The route enforces the inclusion of a valid position for the tree. The remaining form elements are optional and can be easily used to include additional types of annotations (FR4). Once the user pushes the 'Create tree' button, the *Tree creation handler* generates a unique ID for the tree and uses the creation form to prepare a JSON object that follows model `educatree` in Appendix A. The *Tree creation handler* will transfer this object to the *Data manager* to actually create the tree in the dataset. This will be simply achieved by making a C7-like call in Table 5 to the *EducaWood API* with the JSON object enclosed in the body request. The API will validate this call and will then make an `INSERT DATA` operation with the triples to be inserted into the *EducaWood* endpoint.

EducaWood uses R3 routes to provide comprehensive visualizations of trees (FR0). The *Tree handler* is in charge of this functionality; upon extracting the `treeId` from the route, it will ask the *Data manager* to obtain the tree record. If it is not cached, the *Data manager* will make a C6-like call to the *EducaWood API*. With the response, the *Tree handler* will prepare a webpage that adequately presents data to users, as illustrated in Figure 4(c).

The *Tree handler* may include controls to make tree annotations and delete the tree depending on the user identity (FR2 and FR3). New annotations are supported for each annotation type with appropriate widgets to easily include new values, as shown in Figure 4(c). The *Tree handler* will gather the value introduced by the user and prepare a JSON PATCH [56] referred to the model `educatree` in Appendix A. The patch will be transferred to the *Data manager* that will make a C8-like call to the *EducaWood API*. The latter component will validate the call and then proceed with the update of the *EducaWood* endpoint.

Deletion of tree annotations is handled very similarly to annotation creations. Once the user has confirmed the deletion of an annotation, the tree has to be updated with a PATCH to reflect changes: the *Data manager* will make a C8-like call to the *EducaWood API*. Additionally, it will delete the dangling annotation by making a C9-like API call. As for tree deletion, this case requires a C10-like call to remove the tree; the *Data manager* will also make explicit C9-like deletion calls to each associated annotation. This is because CRAFTS does not propagate deletions to other RDF resources by design [12].

Regarding *Last trees handler* and *User handler* in Figure 2, these components are simpler than the previous handlers as they only provide visualizations and do not deal with data updates. *Last trees handler* is purposed for displaying the latest trees and annotations produced in the application. As always, the *Data manager* obtains the content by using query templates `mostRecentEducatrees` and `mostRecentAnnotations` of the API (check details in Appendix A). Unsurprisingly, the *User handler* prepares webpages of users in EducaWood. The *Data manager* employs the model `Person` to retrieve essential information such as nick or creation date (see Appendix A). As user webpages also contain their latest trees and annotations, query templates `mostRecentEducatrees` and `mostRecentAnnotations` are reused for this purpose, in this case setting parameter `user` to only obtain their created trees and annotations.

3.7. Implementation details

EducaWood is coded in JavaScript; this programming language is the natural choice for developing web applications. We use the JavaScript module syntax [57], the recommended way for developing modern web applications, Node Package Manager (npm) [58] as package manager, and Parcel [59] as build tool. Notably, the *Map handler* relies on Leaflet [60] for the interactive map through the use of markers, popups, map controls, and interaction capabilities. As base maps we employ OpenStreetMap [61] and Esri World Imagery [49].

We use Bootstrap [62] as a front-end framework to easily accommodate different browsers and screen sizes in a responsive way. The top-left bar of the map view in Figure 3 uses Bootstrap components. Web pages for tree creation (Figure 4(a)), tree visualizations (Figure 4(c)), last trees, and users are entirely based on the Bootstrap framework. We use Mustache [63] templates in the creation of HTML pages, greatly

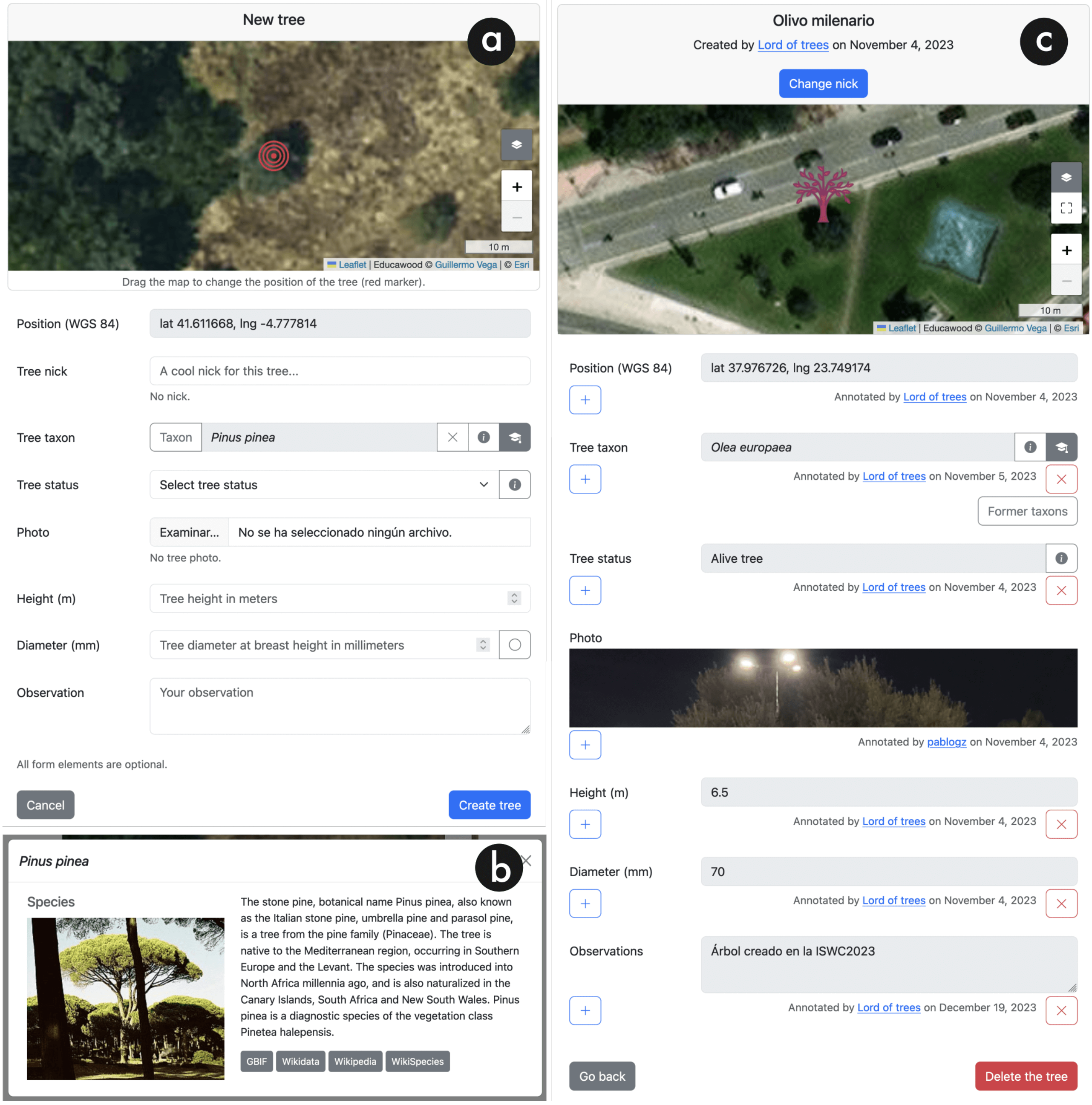

Figure 4. Snapshots of the tree creation and tree view pages of EducaWood. (a) Form for creating a tree associated with route /newtree?loc=41.611668,-4.777814,20z; the position is extracted from the route, while the user has set *Pinus pinea* in the tree taxon field; the remaining fields are currently blank. (b) View of a modal window that appears upon selecting the information button for a tree taxon (*Pinus pinea* in this case). (c) Visualization of the tree at route /tree/Neik7P0woiDY; source data corresponds to the RDF snippet in Listing 1; this view belongs to the creator of the tree, so there are controls for deleting the tree, creating new annotations, and removing existing annotations (with the exception of the photo, which was contributed by another user).

simplifying the rendering of tree and user pages. We also employ the utility functions of Underscore [64] for handling collections along the code.

We use several modules of the Firebase suite [65] for different purposes. We employ Firebase Authentication with Google Sign-in as identity provider; we extract the user's unique ID from this service to assign user IDs in EducaWood[13]. Tree images are stored in Cloud Storage for Firebase. We employ Google Analytics for Firebase to track user activity on EducaWood.

The *EducaWood API* is deployed on a test site of CRAFTS, accessible at https://crafts.gsic.uva.es/apis/educawood/. EducaWood includes a configuration file with the URL of this API along with a token for accessing CRAFTS through Bearer authentication [66]. This configuration file also contains access data to a Solr [67] text search server for looking up world-wide places; this can be seen in the text search box in Figure 3.

Since EducaWood needs to be localized to English and Spanish (requirement NFR2), *EducaWood API* is configured to extract all labels and descriptions in these two languages. Moreover, the application includes a multilingual strings file with all the labels employed in the user interface. Users can choose their language preferences in the application menu (*hamburger* button in Figure 3).

The source code of EducaWood is available on GitHub [68]. A live version of the application [69] is openly available for anybody who wants to use it.

4. Usage of EducaWood

We showcase the functionalities of EducaWood in Section 4.1. A forestry educational scenario of the application is presented in Section 4.2.

4.1. Illustration of EducaWood functionalities

A typical session with EducaWood begins with the map interface, where users can freely explore any location worldwide. Navigation controls for panning and zooming operate in the same way as common map applications. Tree data is rendered according to the approach described in Section 3.5. For instance, snapshot (a) in Figure 3 spans a vast area (zoom level six) in South-West Europe, with a large cluster labeled '430' and numerous tree markers distributed across South and North Spain, South France, and North Italy. When the *IFN* button is activated, additional trees from the IFN dataset are displayed. This is illustrated in snapshot (b), which covers a medium-sized region (zoom level 10) in the northern Spanish plateau: tree clusters proliferate with labels ranging from '101' to '+1K', while tree markers in pale green represent IFN trees.

The *Taxon* button in the upper bar allows filtering by taxon, and the *Layer* button in the right bar switches the base map. Snapshot (c) is positioned in the same area as (b), but with the `esri` query parameter enabled and the taxon filter `ifntx:Species23` (*Pinus pinea*). As a result, the Esri satellite base map is displayed, and only *Pinus pinea* trees remain visible: clusters are fewer, while individual tree markers predominate, shown mostly in pale indigo (IFN trees) with a few in solid indigo (from the EducaWood endpoint). Snapshot (d) zooms in further to a small area (zoom level 18) at the Yutera campus of Universidad de Valladolid, with the `esri` parameter still active. The map shows university buildings surrounded by numerous tree markers, all in solid green (EducaWood endpoint). One marker is expanded, revealing a popup with detailed attributes: a photo, nickname, tree status, height, diameter, creator, and creation date.

The right-hand bar in Figure 3 also features a *Download* button. When activated, users can draw a polygon on the map to define their area of interest. They can then specify the desired level of detail—either summarized tree reports or full tree annotation—and select the output format (GeoJSON, CSV, or KML). The *Map handler* will obtain the set of trees within the polygon and proceed with the download.

[13]This solution has the advantage that user IDs will not change, even if we include additional identity providers.

When the 'More information' button of a tree marker is clicked (see Figure 3(d)), a detailed visualization of the tree is presented, as illustrated in Figure 4(c). The underlying source data is shown in Listing 1, which provides a complete example of a tree annotated with STA and formatted in Turtle. Each annotation type employs the corresponding terms defined in STA (see Section 3.2. This example includes two species annotations, `sta:hasPrimarySpecies` serves to identify the primary species. All annotations—except for the image annotation—were produced by the same user. Note also that the image and observation annotations were added after the initial tree creation, as indicated by the values of `dc:created`. In the application snapshot, however, this information is presented in a user-friendly form, without exposing the RDF terminology.

Listing 1: Sample annotation of a tree with STA.

```
# basic metadata
tree:Neik7P0woiDY a sta:Tree ;
  foaf:nick "Olivo milenario" ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-11-04T21:24:49.606Z"^^xsd:dateTime .

# position annotation
tree:Neik7P0woiDY sta:hasPositionAnnotation posann:Neik7P0woiDY ;
  sta:hasPrimaryPosition posann:Neik7P0woiDY .
posann:Neik7P0woiDY a sta:PositionAnnotation ;
  w3cgeo:lat 37.976725958308535 ;
  w3cgeo:long 23.74917417246512 ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-11-04T21:24:49.606Z"^^xsd:dateTime .

# species annotations
tree:Neik7P0woiDY sta:hasSpeciesAnnotation spann:Neik7P0woiDY, spann:_X-nX5DYwmn8 ;
  sta:hasPrimarySpecies spann:_X-nX5DYwmn8 .
spann:Neik7P0woiDY a sta:SpeciesAnnotation ;
  sta:hasTaxon ifntx:Genus442 ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-11-04T21:24:49.606Z"^^xsd:dateTime .
spann:_X-nX5DYwmn8 a sta:SpeciesAnnotation ;
  sta:hasTaxon ifntx:Species66 ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-11-05T14:50:43.205Z"^^xsd:dateTime .

# tree status annotation
tree:Neik7P0woiDY sta:hasTreeStatusAnnotation treestann:Neik7P0woiDY ;
  sta:hasPrimaryTreeStatus treestann:Neik7P0woiDY .
treestann:Neik7P0woiDY a sta:AliveTreeAnnotation ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-11-04T21:24:49.606Z"^^xsd:dateTime .

# height annotation
tree:Neik7P0woiDY sta:hasHeightAnnotation heightann:Neik7P0woiDY ;
  sta:hasPrimaryHeight heightann:Neik7P0woiDY .
heightann:Neik7P0woiDY a sta:HeightAnnotation ;
  sta:hasHeightInMeters 6.5 ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-11-04T21:24:49.606Z"^^xsd:dateTime .

# diameter annotation
tree:Neik7P0woiDY sta:hasDiameterAnnotation diamann:Neik7P0woiDY ;
  sta:hasPrimaryDiameter diamann:Neik7P0woiDY .
diamann:Neik7P0woiDY a sta:DiameterAnnotation ;
  sta:hasDiameterInMillimeters 70 ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-11-04T21:24:49.606Z"^^xsd:dateTime .

# image annotation
tree:Neik7P0woiDY sta:hasImageAnnotation imgann:3GryiBuMbxHK .
imgann:3GryiBuMbxHK a sta:ImageAnnotation ;
  sta:hasImage img:3GryiBuMbxHK ;
  dc:creator user:osUzXSTSTzbjd5ORsTvAGla8ngH3 ;
  dc:created "2023-11-04T22:12:20.200Z"^^xsd:dateTime .
img:3GryiBuMbxHK a sta:Image ;
  sta:firebasePath "images/osUzXSTSTzbjd5ORsTvAGla8ngH3/3GryiBuMbxHK.png" ;
```

```
  sta:imageURL <https://firebasestorage.googleapis.com/v0/b/educawood-fbaf4.appspot.com/o/images%2
      FosUzXSTSTzbjd5ORsTvAGla8ngH3%2F3GryiBuMbxHK.png?alt=media&token=21b44d3f-2a58-4b16-a6a3-cdcde8da7166> .

# observation annotation
tree:Neik7POwoiDY sta:hasObservationAnnotation observann:l1Uoalgz7hJb .
observann:l1Uoalgz7hJb a sta:ObservationAnnotation ;
  sta:observationText "Arbol creado en la ISWC2023"@es ;
  dc:creator user:F4TwL5qWuMScby3OU3Pk27ZPOBE3 ;
  dc:created "2023-12-19T05:49:30.710Z"^^xsd:dateTime .
```

Tree creation is initiated from the map interface by clicking the tree icon button[14] (see Figure 3), and then selecting the desired position on the map. A creation form, such as the one in Figure 4(a), is then displayed. The position can be fine-tuned by dragging the red marker on the map. While a position is always required, all other fields in the form are optional. In the running example, the user adds a taxon annotation: *Pinus pinea* (`ifntx:Species23`). Taxon information can be readily explored (Figure 4(b)) and is obtained from multiple sources during the bootstrapping routine described in Section 3.5: the CrossForest endpoint (scientific and common names), DBpedia (descriptive text), and Wikidata (images and links to GBIF, Wikidata, Wikipedia, and Wikispecies). When the 'Create tree' button is pressed, the JSON object shown in Listing 2 is sent to the *EducaWood API* via call C7 in Table 5. Then API then maps the object to triples and inserts them into the *EducaWood* endpoint.

Listing 2: Request body for creating `tree:yUhX0LzFP-57` with the form values in Figure 4(a).

```
{
  "iri": "http://educawood.gsic.uva.es/tree/yUhX0LzFP-57",
  "created": "2024-01-12T08:08:06.445Z",
  "creator": "http://educawood.gsic.uva.es/user/F4TwL5qWuMScby3OU3Pk27ZPOBE3",
  "position": {
    "iri": "http://educawood.gsic.uva.es/posann/yUhX0LzFP-57",
    "latWGS84": 41.611668,
    "lngWGS84": -4.777814,
    "types": "http://educawood.gsic.uva.es/sta/ontology/PositionAnnotation",
    "created": "2024-01-12T08:08:06.445Z",
    "creator": "http://educawood.gsic.uva.es/user/F4TwL5qWuMScby3OU3Pk27ZPOBE3"
  },
  "positionAnnotations": "http://educawood.gsic.uva.es/posann/yUhX0LzFP-57",
  "species": {
    "iri": "http://educawood.gsic.uva.es/spann/yUhX0LzFP-57",
    "species": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Species23",
    "types": "http://educawood.gsic.uva.es/sta/ontology/SpeciesAnnotation",
    "created": "2024-01-12T08:08:06.445Z",
    "creator": "http://educawood.gsic.uva.es/user/F4TwL5qWuMScby3OU3Pk27ZPOBE3"
  },
  "speciesAnnotations": "http://educawood.gsic.uva.es/spann/yUhX0LzFP-57",
  "types": "http://educawood.gsic.uva.es/sta/ontology/Tree"
}
```

All created trees and their annotations are publicly accessible. However, only registered users are permitted to make edits. If a user is not logged in, tree data remains visible, but editing controls are disabled. Registered users can add new annotations (blue '+' buttons in Figure 4(c)) and delete their own annotations (red 'x' buttons), but they cannot remove annotations created by others. Tree creators additionally have the right to delete their own trees (red 'Delete the tree' button in Figure 4(c)).

When a user submits a new annotation, the application generates a JSON PATCH similar to the one in Listing 3. This example illustrates the creation of a taxon annotation (`spann:JXbiTiPApexo`) that becomes both the tree's primary species annotation (via a `replace` operation) and part of its list of species annotations (via an `add` operation). The application sends this request as call C8 to the *EducaWood API*. Upon validation, the API maps the request to a `DELETE DATA` operation (removing the previous primary

[14] This button is only enabled under two conditions: (1) the user has logged in, and (2) the zoom level is 16 or higher, since lower levels yield imprecise locations.

species, `spann:yUhX0LzFP-57`) and an `INSERT DATA` operation (adding the new triples to the *EducaWood* endpoint).

Listing 3: JSON PATCH for updating `tree:yUhX0LzFP-57` with a new taxon annotation `spann:JXbiTiPApexo` (value `ifntx:Species26`, *Pinus pinaster*).

```
[
  {
    "op": "replace",
    "path": "/species",
    "value": {
      "iri": "http://educawood.gsic.uva.es/spann/JXbiTiPApexo",
      "species": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Species26",
      "types": "http://educawood.gsic.uva.es/sta/ontology/SpeciesAnnotation",
      "created": "2024-01-12T08:19:58.868Z",
      "creator": "http://educawood.gsic.uva.es/user/F4TwL5qWuMScby3OU3Pk27ZPOBE3"
    }
  },
  {
    "op": "add",
    "path": "/speciesAnnotations/-",
    "value": "http://educawood.gsic.uva.es/spann/JXbiTiPApexo"
  }
]
```

Users may also delete their own annotations. Once confirmed, the application generates a JSON PATCH like the one in Listing 4, which includes a `remove` operation to eliminate `spann:JXbiTiPApexo` from the list of taxon annotations for `tree:yUhX0LzFP-57`. At the same time, a `replace` operation restores `spann:yUhX0LzFP-57` as the primary species. This patch is sent as call C9 and processed by the API. As a result, the tree reverts to its previous state, with `spann:yUhX0LzFP-57` as primary species, while `spann:JXbiTiPApexo` remains in the dataset as an unlinked ("dangling") annotation. The application will then send call C9 to remove the annotation `spann:JXbiTiPApexo` from the *EducaWood* endpoint.

Listing 4: JSON PATCH for updating `tree:yUhX0LzFP-57` to remove taxon annotation `spann:JXbiTiPApexo`.

```
[
  {
    "op": "remove",
    "path": "/speciesAnnotations/0"
  },
  {
    "op": "replace",
    "path": "/species",
    "value": "http://educawood.gsic.uva.es/spann/yUhX0LzFP-57"
  }
]
```

Finally, tree creators may choose to delete their trees entirely. After confirmation, the application issues call C10 to remove `tree:yUhX0LzFP-57`. Deletion of the tree does not automatically remove its annotations; the application will then send C9-like calls to delete each associated annotation (`spann:yUhX0LzFP-57` and `posann:yUhX0LzFP-57` in this case).

4.2. A forestry education scenario

We present an urban tree management scenario that can be supported with EducaWood. It is inspired by the educational design of the "Reforestation, Nurseries, and Gardening" course in the third year of the Forestry and Environmental Engineering degree at the Universidad de Valladolid. The scenario has three main learning goals: (1) managing urban tree inventories; (2) identifying and calculating the key variables of an urban tree management plan; and (3) writing a strategic plan based on those variables.

In a first stage, students collaboratively create a tree inventory with EducaWood. The target area should be an urban garden of suitable size to ensure a manageable workload for students. Using the application, they annotate trees by recording positions, species identifications, images, dendrometric measurements, tree status, and textual observations. This activity may be conducted in a single session or distributed across several sessions, depending on time constraints and contextual factors. A preliminary training session with EducaWood is recommended to familiarize students with the application's functionality. Optional follow-up sessions may also be scheduled to correct missing or inaccurate annotations, e.g. an incorrect species identification.

In a second stage, students work with the previously created tree inventory. With EducaWood, they select the target area and download the tree data, which serve as the basis for calculating several variables: quality indicators (number of trees per inhabitant or per hectare), biodiversity indicators (number of species, proportion of the most abundant species), dimensional indicators (distribution of trees by diameter classes or height ranges), and condition indicators. Finally, students draft an urban tree management strategic plan informed by these calculated variables.

5. EducaWood in practice

In Section 5.1, we present evidence regarding the impact of EducaWood thus far. Additionally, we report a pilot in an urban tree management activity with forestry engineering students in Section 5.2.

5.1. Preliminary impact

In 2021, we presented an early demonstrator of EducaWood in the 16th European Conference on Technology Enhanced Learning (EC-TEL 2021) [70]. Although the functionality of this demonstrator was limited, it allowed testing key system components, particularly the creation of trees through a CRAFTS API. Following this, EducaWood received the third award in the "*III Desafío Aporta*" [71], a Spanish open data challenge sponsored by the Spanish Ministry of Digital Transformation.

Encouraged by this early success, we worked on a new version of EducaWood that meets the requirements outlined in Section 3.1. In July 2023, we released a new prototype [69], aimed at supporting forestry education scenarios. We tested the application with a selected group of forestry experts, who provided very positive feedback and valuable suggestions, leading to the incorporation of features such as a tutorial, satellite base map, drawing tool for defining data download areas, support for tree nicks and text observations, and tree form improvements to facility data entering. More recently, we expanded the outreach of EducaWood by sharing it with academic contacts and running a pilot with forestry students which is summarized in Section 5.2.

Since traffic on the EducaWood website is tracked with Google Analytics, we can report some figures in the period from July 2023 to January 2025. Table 6 summarizes the collected data; 653 active users have employed EducaWood in 1,471 engaged sessions[15] with an average duration of 2 minutes and 30 seconds. Most users are from Spain (52.5%), while the rest come from Italy (8.9%), the Netherlands (6.4%), Sweden (6.3%), Finland (4.9%), United States (4.9%), and other countries (16.1%). Devices employed include desktop computers (51.0%), mobiles (48.4%), and tablets (0.6%). We also tracked page views (32.3K in total), finding that the map interface route is the most intensively used (77.8% of all page views); activity in the remaining routes range from 7.8% to 1.1% (see Table 6).

We have also analyzed the annotations created in the EducaWood dataset. As of January 2025, the dataset contains 48K triples, corresponding to 1,682 trees and 4.5K tree annotations. Notably, 51 users contributed to content generation, constituting 7.8% of the application user base.

[15] According to Google Analytics, an engaged session is one that lasted longer than 10 seconds, or had a key event, or had two or more screen or page views.

Table 6

Uptake of the test site of EducaWood.

Item	Value
# of active users	653
# of engaged sessions	1,471
Average time per session	2m 30s
# of page views	32.3K
% of landing page views (route R0)	4.3%
% of map page views (route R1)	77.8%
% of tree creation page views (route R2)	7.8%
% of tree page views (route R3)	5.8%
% of last created trees page views (route R4)	1.1%
% of user page views (route R5)	3.2%

The tracked data also includes latencies for `select_content` events, which encompass map updates and tree page visualizations. Although we use an additional `content_type` parameter, Firebase does not support filtering based on this attribute. Nevertheless, the average latency is 0.4s—well below the 1-second threshold [72, ch. 5]—ensuring seamless user interaction without interrupting their workflow. For write operations, tree annotation creation and deletion, the latencies are 1.6s and 1.1s, respectively. These actions take slightly longer due to their non-cacheable nature, but occur far less frequently (especially deletions). Their response times remain reasonable, as they typically mark the completion of a user task rather than being part of an ongoing interaction.

5.2. Pilot study

Context of the study. We have carried out a pilot study of EducaWood within the context of a "Reforestation, Nurseries, and Gardening" course in the third year of the Forestry and Environmental Engineering degree at Universidad de Valladolid. The course has two teachers and 20 enrolled students. The learning scenario corresponds to the one presented in Section 4.2. To achieve its learning goals, and considering that the students did not have prior knowledge about tree management, the teachers structured the scenario in three main stages.

First, a two-hour training session in November 2023 familiarized students with EducaWood through a classroom demonstration, followed by a practical tree annotation session in the campus gardens, and a subsequent verification of the accuracy of tree labeling in the classroom. During the classroom demonstration, the teachers introduced the application by showing its interface, explaining all available options, and providing examples. A first annotation session took place immediately after the classroom demo. During this session, the teachers were available to answer any questions and doubts in relation with the use of EducaWood. Afterwards, the teachers reviewed the students' annotations. Errors detected included missing annotations and unclear observations that were difficult to interpret by other users. A follow-up classroom session was conducted to provide feedback on these mistakes and guide students on how to correct them.

Second, the students were asked to collaboratively create a tree inventory at the Yutera Campus throughout November and December 2023. Each student had to annotate a minimum of 20 trees with EducaWood. The minimum of 20 trees per student was estimated to ensure that the entire tree population on campus would be recorded, while keeping the workload of the students within the course restrictions.

Third, students had to prepare an urban forestry management plan for the Yutera Campus, utilizing the collaboratively created tree inventory. This involved downloading tree data from EducaWood, calculating various variables, and preparing the tree management strategic plan.

While the pilot was ongoing, the resulting tree inventory could be accessed and verified using EducaWood by both teachers and students—Figure 3(d) provides an example at the target location.

Method. We followed a mixed-methods, embedded (concurrent-nested) design with qualitative preference [73], trying to understand, from an interpretive perspective, how EducaWood supported the teachers' practice in terms of pedagogical alignment, assessment affordances, and usability challenges. Two data gathering instruments were employed: students and teachers were asked to fill the standardized System Usability Score (SUS), a validated quantitative survey widely employed in usability studies [74]; and, a semi-structured interview between one of the researchers and the teachers focused on usability issues and pedagogical alignment of EducaWood. Descriptive statistics were employed for analyzing the SUS data, while inductive coding was applied to the transcribed interview, using the technique known as "solo coding" in which the whole research team discusses and agrees upon the emerging codes identified by one of the researchers [75].

It is important to underline that our interpretive approach in this pilot study does not aim to provide statistically generalizable evidence about the educational affordances or usability of EducaWood. Rather, our goal is to gain a sufficiently deep understanding of how the application is used within an authentic, yet specific, learning context—one that exemplifies the challenges identified earlier in the paper.

Results. Each student annotated a similar number of trees, as the scenario design and teachers' estimation of the tree population ensured balanced assignments. Teachers verified that all students completed at least 20 annotations and maintained quality standards. The activity score was based entirely on the strategic forest management plan, and all enrolled students successfully passed the activity. We received 16 responses[16], obtaining an average SUS score of 75.2, with a standard deviation of 11.5. This figure is good, given that SUS scores range from 0 to 100. According to the grading scale interpretation of SUS scores in [76, ch. 8], EducaWood was graded with a B. This indicates a good level of usability. Nevertheless, the teachers also reported some minor usability problems during the semi-structured interview: *"There are some format issues; I had to call [researcher's name] for help. The system exports data as a CSV file, but when converting to Excel, some problems arise".* Another issue raised by the teachers was the need for better documentation and tutorials: *"I missed having tutorials. At a minimum, there should be a video or a tutorial for data collection".*

Also during the semi-structured interview, the teachers provided their feedback about the impact of EducaWood when supporting the described learning scenario. Interestingly, one of them highlighted the potential of EducaWood for supporting forestry engineering educational activities related to tree management, from a more practical perspective: *"What is interesting is that students not only label the trees but also calculate a series of indices that could be incorporated. In urban tree management plans, such as those from Madrid, I ask them to calculate the indices listed in the plan. Similar indices exist in plans from other cities like Seville, Barcelona, and Valencia. These indices help determine age, size, species diversity, pruning needs, and overall tree management. Once students have a database, EducaWood allows them to put numbers to their knowledge, which is often lacking in urban tree management. A major issue in urban tree management is the lack of inventories and tools to create them continuously and easily. EducaWood has enormous potential beyond its role as an educational tool."* Furthermore, the teachers explained that this was the first time they carried out this type of learning scenario, thanks to the affordances of EducaWood: *"[EducaWood] is an easy thing, it allows you to resume the activity when you had stopped before, it allows you to review data, it allows you to check, it allows you to work with the data from others. It gives you a lot of possibilities, a big lot of possibilities. It is a powerful tool, a very powerful tool."* In fact, the teachers pointed out that EducaWood solves one important existing problem when planning educational activities in relation with tree management plans: *"One of the problems we have in relation with urban tree management is that we do not have tree inventories and we do not have tools that allow you to create those inventories, in a sustainable basis. Well, this is a very great potential for EducaWood, very very great. EducaWood is truly an educational tool, but it could even go beyond."* EducaWood has also provided

[16]The SUS scores can be accessed in the EducaWood GitHub repository at https://github.com/guiveg/educawood/blob/main/SUS_educawood.csv.

support during the assessment of the learning activities: *"I have assessed two aspects. One, the inventory that they have created with EducaWood, and, another, the report. Then, the inventory, what I check is that they have 30 trees and that they are completely described, more or less. And this is assessed on the one hand, and then the report they write with the data from the whole database."*

6. Discussion

EducaWood is a LOD-based application for forestry education that successfully meets all the requirements in Table 2. It enables multi-author tree management and geospatial data visualization, integrating diverse data sources through a CRAFTS API. Tree annotation relies on the STA ontology, offering a flexible model for annotating trees and including a conflict resolution mechanism via primary annotations—Section 3 gives multiple examples of the use of this ontology for annotating trees. STA can be extended in different ways and we have already received suggestions from foresters, including: new annotation types like microhabitats (cavities, excrescences, exudates, epiphytics, nests, etc.)—see [77]; additional spatial entities such as down deadwood [47]; and, specialized terminology for urban tree management (e.g. tree pits and pruning).

EducaWood addresses three major challenges in handling semantic data for forestry education. Regarding CHALLENGE #1, human-computer interaction with the Semantic Web, the user interface of EducaWood follows good practices in web application design to simplify interaction with Semantic Web technologies. It provides an interactive map to visualize tree data at various zoom levels, complemented by form-based interfaces for both viewing and authoring trees. Page URLs are designed to encapsulate all application state—check the routes in Table 4—ensuring that a URL will produce the same view regardless of the device employed and allowing users to safely bookmark and share EducaWood URLs. Findings from the pilot indicate that this user interface design effectively addresses two key objectives: (1) concealing the intricacies of Semantic Web technologies, and (2) facilitating user tasks. This is supported by the good SUS score and the successful creation of a tree inventory with EducaWood—see Section 5.2. The application streamlined the tree annotation workflow, which traditionally involves collecting data on paper, transferring it to a computer, and then sharing it. With EducaWood, students uploaded data *in situ* in a consistent format, reducing positioning errors and eliminating the need for manual data transfer. This also significantly reduced teachers' workload, as they no longer needed to collect, aggregate, and redistribute annotations. It is noteworthy that neither pilot students nor teachers have a background in Semantic Web or databases, highlighting the challenge of user interaction with Semantic Web technologies [6–8]. The use of form-based interfaces for semantic annotation aligns with established good practices in usability and has proven effective, as demonstrated in EducaWood and other systems such as Wikidata.

Accessing data in EducaWood can be demanding due to the mixture of write and read operations across multiple data sources (CHALLENGE #2). Nevertheless, the utilization of a CRAFTS API significantly streamlines this process by providing a centralized access point for all data operations. This required a thorough authoring of the configuration file in Appendix A to support the different features of EducaWood—Table 5 gives a good overview of the API calls used in the application. Template queries are primarily employed during the bootstrapping routine and map exploration, with careful attention given to meeting latency requirements, as elaborated in Section 3. In this regard, we employ client-side caching along a user session to avoid duplicated requests to the API, as well as exploiting geospatial relations among cells to derive new information without making further API calls. Tree management essentially involves the use of model `educatree` with the appropriate HTTP methods (GET, PUT, PATCH, DELETE) for retrieving, creating, updating, and deleting trees. All in all, the application only sees JSON data and REST API calls; CRAFTS automatically makes the translation of API requests into SPARQL queries.

When addressing CHALLENGE #3, EducaWood employs various techniques to efficiently handle semantic geospatial data. Our grid of cells for requesting tree data is inspired by tiled web maps [78], a prevalent strategy for enhancing the cacheability of web maps. By dividing a map into a grid, EducaWood ensures that identical API calls are made for data within the same cell by different users, optimizing server caching at CRAFTS. To manage cells with varying tree densities, EducaWood uses a procedure that limits data requests when numerous trees are present. Moreover, EducaWood also exploits geospatial relations among cells to reduce the number of API calls (see Section 3.4). We have tested this approach with the 1.4 million trees in the IFN dataset across various regions of Spain and at different zoom levels. The application processes them efficiently, maintaining response times below one second without significant delays. These techniques hold broader applicability to scenarios involving semantic geospatial data. For instance, they can be applied in Forest Explorer [39] to improve the handling of forestry data. We are currently refining Forest Explorer with techniques from EducaWood, including a CRAFTS API, a grid of cells, exploitation of geospatial cell relationships, and URL redesign to facilitate their sharing. While Forest Explorer and EducaWood share some similarities, they are fundamentally different in scope and functionality. Forest Explorer does not support collaborative tree annotation (requirements FR0–3 in Table 2) and is limited to Spain and Portugal, so world-wide tree exploration (FR 4) is not supported. Additionally, EducaWood focuses exclusively on trees, whereas Forest Explorer also includes administrative regions, land cover patches, and inventory plots. Given such differences, it is not possible to make a quantitative comparison between the two applications.

Thus far, EducaWood has been tested by more than 650 users, with 7.8% of them actively contributing content. This creator-to-consumer ratio surpasses the 1% rule of thumb often observed in Internet communities [79], although collected data in EducaWood is still limited. To moderate its emerging community, we have defined several roles within the application: normal users can create trees and annotations, with the ability to delete their own contributions only; superusers can delete any content and ban normal users; while banned users are restricted from authoring. Each annotation includes its creator, facilitating swift action against vandalism.

While EducaWood presents significant advancements, it also has some limitations. First, the application has been tested primarily in a pilot study in a University course. While the results are promising, broader testing across different educational contexts and user groups is needed to validate its effectiveness. Second, while the STA ontology is flexible, it may require further refinement and expansion based on user feedback and evolving educational needs, as discussed earlier. Lastly, EducaWood could benefit from explicit support for tree annotation tasks that teachers can design, thereby better aligning the application with specific educational objectives, similarly to CHEST [26] in the Cultural Heritage domain.

7. Conclusion and future work

EducaWood emerges as a versatile educational tool poised to enhance environmental education across various educational levels, spanning from secondary to university master's programs. The learning objectives of EducaWood can encompass a broad spectrum, aiming to cultivate various skills and knowledge among students, depending on the activity designed by teachers. This paper has illustrated how the annotation capabilities of EducaWood can be used in one particular authentic learning scenario in the context of forestry engineering education: tree management. However, other interesting learning scenarios might include, for example, the differentiation of main groups of forest species, fostering a deeper understanding of ecosystem diversity, and igniting a greater interest in nature among learners. Also, interdisciplinary learning can be favored by incorporating mathematical concepts such as calculating structural diversity indices and carbon sequestration rates, thereby enhancing students' quantitative reasoning skills. So EducaWood promotes collaborative learning experiences, nurturing teamwork and communication skills essential for effective problem solving and group dynamics. Nevertheless, and in order to better understand the educational affordances of EducaWood, we plan future research lines aimed at defining different personas of educational user types and linking their requirements with the application functionalities more explicitly, on top of those more specific to tree annotation. For example, while teachers and students are the primary user types of EducaWood, we can take a step further by identifying distinct teacher personas based on their experience, goals, behaviors, and needs. This allows us to assess how well EducaWood supports a diverse range of teaching approaches. Some teachers may use EducaWood solely as an annotation tool—as illustrated in the study presented in this paper—whereas others might design more complex learning activities centered around specific trees of interest. Indeed, we are currently working on adding to EducaWood a teacher interface for the creation of different types of geolocalized learning tasks such as multiple-choice questions, comparing trees, etc. This is an approach that we have explored in the domain of Cultural Heritage education (see, e.g., [26]).

By bridging classroom learning with real-world experiences, EducaWood extends the educational landscape beyond traditional confines, fostering active and contextualized learning. Moreover, it amplifies ecological awareness by spotlighting forests' pivotal role in climate change mitigation and biodiversity conservation. Its innovative features, such as collaborative annotation functionalities, not only facilitate remote learning but also enable students from diverse backgrounds to engage with forest ecosystems regardless of geographical constraints. This adaptability is very valuable, especially in navigating challenges such as those posed by the COVID-19 pandemic, where traditional in-person educational activities may be impractical. Our future work includes new pilots in forestry education to gather feedback and further improve EducaWood, thereby bolstering its utility for environmental education. We also plan to conduct additional tests to evaluate scalability, as well as to improve EducaWood with the possibility of combining annotations and data analysis at different layers (e.g., highlighting a big amount of dead trees in an area).

Acknowledgements

This work has been partially funded by the European Commission through "Small4Good" (101135517), by the Spanish Research Agency through "LOD.For.Trees" (TED2021-130667B-I00), "H2O" (PID2020-112584RB-C32), and "GENIELearn" (PID2023-146692OB-C32) projects, by the Junta de Castilla y León through project "iuFOR Institute Unit of Excellence" (CLU-2019-01) of Universidad de Valladolid and co-financed by the European Regional Development Fund (ERDF "Europe drives our growth"). The authors also want to thank the participants in the pilot study.

Appendix A. CRAFTS API configuration of EducaWood

We include here the CRAFTS API configuration file employed in EducaWood. For further information on the use of CRAFTS, refer to [12].

```
{
  "apiId": "educawood",
  "endpoints": [
    {
      "id": "crossforest",
      "sparqlURI": "https://crossforest.gsic.uva.es/sparql",
      "graphURI": "http://crossforest.eu",
      "httpMethod": "GET"
    },
    {
      "id": "educawood",
      "sparqlURI": "https://semanticforest.gsic.uva.es/sparql",
      "graphURI": "http://educawood.gsic.uva.es",
      "httpMethod": "GET",
      "sparqlUpdate": {
        "sparqlURI": "https://semanticforest.gsic.uva.es/sparql-auth",
        "authInfo": {
          "user": "NOT SHOWN",
          "password": "NOT SHOWN",
          "type": "digest"
        }
      }
    },
    {
      "id": "dbpedia",
      "sparqlURI": "http://dbpedia.org/sparql",
      "graphURI": "http://dbpedia.org",
```

```
        "httpMethod": "GET"
      },
      {
        "id": "wikidata",
        "sparqlURI": "https://query.wikidata.org/sparql",
        "httpMethod": "GET"
      }
    ],
    "model": [
      {
        "id": "Tree",
        "oprops": [ ],
        "dprops": [
          {
            "label": "dbh1mm",
            "iri": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasDBH1InMillimeters",
            "endpoint": "crossforest"
          },
          {
            "label": "dbh2mm",
            "iri": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasDBH2InMillimeters",
            "endpoint": "crossforest"
          },
          {
            "label": "heightM",
            "iri": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasTotalHeightInMeters",
            "endpoint": "crossforest"
          }
        ],
        "types": [
          {
            "label": "species",
            "targetId": "Species",
            "restrictions": [
              "?type a/rdfs:subClassOf <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Taxon> ."
            ],
            "embed": true,
            "endpoint": "crossforest"
          }
        ]
      },
      {
        "id": "Person",
        "oprops": [
        ],
        "dprops": [
          {
            "label": "created",
            "iri": "http://purl.org/dc/terms/creator>/<http://purl.org/dc/terms/created",
            "inv": true,
            "orderBy": "?value",
            "limit": 1,
            "endpoint": "educawood"
          },
          {
            "label": "nick",
            "iri": "http://xmlns.com/foaf/0.1/nick",
            "endpoint": "educawood"
          },
          {
            "label": "isMasterAnnotator",
            "iri": "http://educawood.gsic.uva.es/sta/ontology/isMasterAnnotator",
            "endpoint": "educawood"
          },
          {
            "label": "cannotAnnotate",
            "iri": "http://educawood.gsic.uva.es/sta/ontology/cannotAnnotate",
            "endpoint": "educawood"
          },
          {
            "label": "numberOfAnnotations",
            "iri": "http://purl.org/dc/terms/creator",
            "inv": true,
            "altResult": "count(distinct ?value) as ?nanns",
            "altVariable": "nanns",
            "endpoint": "educawood"
```

```
          },
          {
            "label": "numberOfEducatrees",
            "iri": "http://purl.org/dc/terms/creator",
            "inv": true,
            "restrictions": [ "?value a <http://educawood.gsic.uva.es/sta/ontology/Tree> ." ],
            "altResult": "count(distinct ?value) as ?nets",
            "altVariable": "nets",
            "endpoint": "educawood"
          }
        ],
        "types": [
          {
            "label": "types",
            "endpoint": "educawood",
            "writeonly": true
          }
        ]
      },
      {
        "id": "BasicEducaTree",
        "oprops": [
          {
            "label": "creator",
            "targetId": "Person",
            "iri": "http://purl.org/dc/terms/creator",
            "embed": true,
            "endpoint": "educawood"
          },
          {
            "label": "species",
            "targetId": "Species",
            "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimarySpecies>/<http://educawood.gsic.uva.es/sta/ontology/
                 hasTaxon",
            "embed": false,
            "endpoint": "educawood"
          },
          {
            "label": "images",
            "iri": "http://educawood.gsic.uva.es/sta/ontology/hasImageAnnotation>/<http://educawood.gsic.uva.es/sta/ontology/
                 hasImage>/<http://educawood.gsic.uva.es/sta/ontology/imageURL",
            "endpoint": "educawood"
          },
          {
            "label": "treeStatus",
            "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryTreeStatus>/<http://www.w3.org/1999/02/22-rdf-syntax-
                 ns#type",
            "endpoint": "educawood"
          }
        ],
        "dprops": [
          {
            "label": "created",
            "iri": "http://purl.org/dc/terms/created",
            "endpoint": "educawood"
          },
          {
            "label": "nick",
            "iri": "http://xmlns.com/foaf/0.1/nick",
            "endpoint": "educawood"
          },
          {
            "label": "lat",
            "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryPosition>/<http://www.w3.org/2003/01/geo/wgs84_pos#
                 lat",
            "endpoint": "educawood"
          },
          {
            "label": "lng",
            "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryPosition>/<http://www.w3.org/2003/01/geo/wgs84_pos#
                 long",
            "endpoint": "educawood"
          },
          {
            "label": "dbh",
```

```
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryDiameter>/<http://educawood.gsic.uva.es/sta/ontology/
             hasDiameterInMillimeters",
        "endpoint": "educawood"
      },
      {
        "label": "height",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryHeight>/<http://educawood.gsic.uva.es/sta/ontology/
             hasHeightInMeters",
        "endpoint": "educawood"
      },
      {
        "label": "observations",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasObservationAnnotation>/<http://educawood.gsic.uva.es/sta/
             ontology/observationText",
        "endpoint": "educawood"
      }
    ],
    "types": [ ]
  },
  {
    "id": "EducaTree",
    "oprops": [
      {
        "label": "creator",
        "targetId": "Person",
        "iri": "http://purl.org/dc/terms/creator",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "position",
        "targetId": "PositionAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryPosition",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "positionAnnotations",
        "targetId": "PositionAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPositionAnnotation",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "species",
        "targetId": "SpeciesAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimarySpecies",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "speciesAnnotations",
        "targetId": "SpeciesAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasSpeciesAnnotation",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "diameter",
        "targetId": "DiameterAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryDiameter",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "diameterAnnotations",
        "targetId": "DiameterAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasDiameterAnnotation",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "height",
        "targetId": "HeightAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryHeight",
        "embed": true,
```

```
        "endpoint": "educawood"
      },
      {
        "label": "heightAnnotations",
        "targetId": "HeightAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasHeightAnnotation",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "observations",
        "targetId": "ObservationAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasObservationAnnotation",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "treeStatus",
        "targetId": "TreeStatusAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasPrimaryTreeStatus",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "treeStatusAnnotations",
        "targetId": "TreeStatusAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasTreeStatusAnnotation",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "imageAnnotations",
        "targetId": "ImageAnnotation",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasImageAnnotation",
        "embed": true,
        "endpoint": "educawood"
      }
    ],
    "dprops": [
      {
        "label": "created",
        "iri": "http://purl.org/dc/terms/created",
        "endpoint": "educawood"
      },
      {
        "label": "nick",
        "iri": "http://xmlns.com/foaf/0.1/nick",
        "endpoint": "educawood"
      }
    ],
    "types": [
      {
        "label": "types",
        "endpoint": "educawood",
        "writeonly": true
      }
    ]
  },
  {
    "id": "PositionAnnotation",
    "oprops": [
      {
        "label": "creator",
        "targetId": "Person",
        "iri": "http://purl.org/dc/terms/creator",
        "embed": true,
        "endpoint": "educawood"
      }
    ],
    "dprops": [
      {
        "label": "created",
        "iri": "http://purl.org/dc/terms/created",
        "endpoint": "educawood"
      },
      {
```

```
        "label": "latWGS84",
        "iri": "http://www.w3.org/2003/01/geo/wgs84_pos#lat",
        "endpoint": "educawood"
      },
      {
        "label": "lngWGS84",
        "iri": "http://www.w3.org/2003/01/geo/wgs84_pos#long",
        "endpoint": "educawood"
      }
    ],
    "types": [
      {
        "label": "types",
        "endpoint": "educawood",
        "writeonly": true
      }
    ]
  },
  {
    "id": "DiameterAnnotation",
    "oprops": [
      {
        "label": "creator",
        "targetId": "Person",
        "iri": "http://purl.org/dc/terms/creator",
        "embed": true,
        "endpoint": "educawood"
      }
    ],
    "dprops": [
      {
        "label": "created",
        "iri": "http://purl.org/dc/terms/created",
        "endpoint": "educawood"
      },
      {
        "label": "millimeters",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasDiameterInMillimeters",
        "endpoint": "educawood"
      }
    ],
    "types": [
      {
        "label": "types",
        "endpoint": "educawood",
        "writeonly": true
      }
    ]
  },
  {
    "id": "HeightAnnotation",
    "oprops": [
      {
        "label": "creator",
        "targetId": "Person",
        "iri": "http://purl.org/dc/terms/creator",
        "embed": true,
        "endpoint": "educawood"
      }
    ],
    "dprops": [
      {
        "label": "created",
        "iri": "http://purl.org/dc/terms/created",
        "endpoint": "educawood"
      },
      {
        "label": "meters",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasHeightInMeters",
        "endpoint": "educawood"
      }
    ],
    "types": [
      {
        "label": "types",
        "endpoint": "educawood",
```

```
        "writeonly": true
      }
    ]
  },
  {
    "id": "ObservationAnnotation",
    "oprops": [
      {
        "label": "creator",
        "targetId": "Person",
        "iri": "http://purl.org/dc/terms/creator",
        "embed": true,
        "endpoint": "educawood"
      }
    ],
    "dprops": [
      {
        "label": "created",
        "iri": "http://purl.org/dc/terms/created",
        "endpoint": "educawood"
      },
      {
        "label": "text",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/observationText",
        "endpoint": "educawood"
      }
    ],
    "types": [
      {
        "label": "types",
        "endpoint": "educawood",
        "writeonly": true
      }
    ]
  },
  {
    "id": "TreeStatusAnnotation",
    "oprops": [
      {
        "label": "creator",
        "targetId": "Person",
        "iri": "http://purl.org/dc/terms/creator",
        "embed": true,
        "endpoint": "educawood"
      }
    ],
    "dprops": [
      {
        "label": "created",
        "iri": "http://purl.org/dc/terms/created",
        "endpoint": "educawood"
      }
    ],
    "types": [
      {
        "label": "treeStatus",
        "endpoint": "educawood"
      }
    ]
  },
  {
    "id": "ImageAnnotation",
    "oprops": [
      {
        "label": "creator",
        "targetId": "Person",
        "iri": "http://purl.org/dc/terms/creator",
        "embed": true,
        "endpoint": "educawood"
      },
      {
        "label": "image",
        "targetId": "Image",
        "iri": "http://educawood.gsic.uva.es/sta/ontology/hasImage",
        "embed": true,
        "endpoint": "educawood"
```

```
    }
  ],
  "dprops": [
    {
      "label": "created",
      "iri": "http://purl.org/dc/terms/created",
      "endpoint": "educawood"
    }
  ],
  "types": [
    {
      "label": "types",
      "endpoint": "educawood",
      "writeonly": true
    }
  ]
},
{
  "id": "Image",
  "oprops": [
    {
      "label": "imageURL",
      "iri": "http://educawood.gsic.uva.es/sta/ontology/imageURL",
      "endpoint": "educawood"
    }
  ],
  "dprops": [
    {
      "label": "firebasePath",
      "iri": "http://educawood.gsic.uva.es/sta/ontology/firebasePath",
      "endpoint": "educawood"
    }
  ],
  "types": [
    {
      "label": "plantPart",
      "restrictions": [
        "FILTER (?type NOT IN ( <http://educawood.gsic.uva.es/sta/ontology/Image> ))"
      ],
      "endpoint": "educawood"
    },
    {
      "label": "types",
      "endpoint": "educawood",
      "writeonly": true
    }
  ]
},
{
  "id": "SpeciesAnnotation",
  "oprops": [
    {
      "label": "creator",
      "targetId": "Person",
      "iri": "http://purl.org/dc/terms/creator",
      "embed": true,
      "endpoint": "educawood"
    },
    {
      "label": "species",
      "targetId": "Species",
      "iri": "http://educawood.gsic.uva.es/sta/ontology/hasTaxon",
      "embed": true,
      "endpoint": "educawood"
    }
  ],
  "dprops": [
    {
      "label": "created",
      "iri": "http://purl.org/dc/terms/created",
      "endpoint": "educawood"
    }
  ],
  "types": [
    {
      "label": "types",
```

```
            "endpoint": "educawood",
            "writeonly": true
          }
        ]
      },
      {
        "id": "Species",
        "oprops": [
          {
            "label": "wikidata",
            "targetId": "WikidataTaxonBasic",
            "iri": "http://www.w3.org/2000/01/rdf-schema#subClassOf",
            "restrictions": [
              "FILTER strstarts(str(?value), \"http://www.wikidata.org/entity/\" )"
            ],
            "embed": true,
            "endpoint": "educawood"
          },
          {
            "label": "subclasses",
            "iri": "http://www.w3.org/2000/01/rdf-schema#subClassOf",
            "endpoint": "educawood",
            "writeonly": true
          },
          {
            "label": "creator",
            "targetId": "Person",
            "iri": "http://purl.org/dc/terms/creator",
            "endpoint": "educawood",
            "writeonly": true
          }
        ],
        "dprops": [
          {
            "label": "scientificName",
            "iri": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/acceptedNameWithoutAuthor",
            "endpoint": "educawood"
          },
          {
            "label": "vulgarName",
            "iri": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/vulgarName",
            "restrictions": [
              "FILTER(LANG(?value) = 'en' || LANG(?value) = 'es')"
            ],
            "endpoint": "educawood"
          },
          {
            "label": "wikipediaPage",
            "iri": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasWikipediaPage",
            "endpoint": "educawood"
          },
          {
            "label": "wikispeciesPage",
            "iri": "https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasWikispeciesPage",
            "endpoint": "educawood"
          },
          {
            "label": "created",
            "iri": "http://purl.org/dc/terms/created",
            "endpoint": "educawood",
            "writeonly": true
          }
        ],
        "types": [
          {
            "label": "types",
            "endpoint": "educawood",
            "writeonly": true
          }
        ]
      },
      {
        "id": "WikidataTaxonBasic",
        "oprops": [
          {
            "label": "image",
```

```
        "iri": "http://www.wikidata.org/prop/P18>/<http://www.wikidata.org/prop/statement/P18",
        "endpoint": "wikidata"
      },
      {
        "label": "gbifPage",
        "iri": "http://www.wikidata.org/prop/direct/P846",
        "restrictions": [ "BIND(URI(CONCAT(\"https://www.gbif.org/species/\", ?value)) AS ?gbif)" ],
        "altResult": "?gbif",
        "altVariable": "gbif",
        "endpoint": "wikidata"
      }
    ],
    "dprops": [
      {
        "label": "sitelinks",
        "iri": "http://wikiba.se/ontology#sitelinks",
        "restrictions": [ "hint:Query hint:optimizer \"None\"." ],
        "endpoint": "wikidata"
      },
      {
        "label": "statements",
        "iri": "http://wikiba.se/ontology#statements",
        "restrictions": [ "hint:Query hint:optimizer \"None\"." ],
        "endpoint": "wikidata"
      },
      {
        "label": "comment",
        "iri": "http://www.w3.org/2002/07/owl#sameAs>/<http://www.w3.org/2000/01/rdf-schema#comment",
        "inv": true,
        "restrictions": [
          "FILTER(LANG(?value) = 'en' || LANG(?value) = 'es')"
        ],
        "endpoint": "dbpedia"
      }
    ],
    "types": []
  },
  {
    "id": "WikidataTaxon",
    "oprops": [
      {
        "label": "genus",
        "targetId": "WikidataTaxon",
        "iri": "http://www.wikidata.org/prop/direct/P171",
        "altResult": "?genus",
        "altVariable": "genus",
        "restrictions": [
              "?iri <http://www.wikidata.org/prop/direct/P171>+ ?genus .",
              "?genus <http://www.wikidata.org/prop/direct/P105> <http://www.wikidata.org/entity/Q34740> ."
        ],
        "embed": false,
        "endpoint": "wikidata"
      },
      {
        "label": "family",
        "targetId": "WikidataTaxon",
        "iri": "http://www.wikidata.org/prop/direct/P171",
        "altResult": "?family",
        "altVariable": "family",
        "restrictions": [
              "?iri <http://www.wikidata.org/prop/direct/P171>+ ?family .",
              "?family <http://www.wikidata.org/prop/direct/P105> <http://www.wikidata.org/entity/Q35409> ."
        ],
        "embed": false,
        "endpoint": "wikidata"
      },
      {
        "label": "image",
        "iri": "http://www.wikidata.org/prop/P18>/<http://www.wikidata.org/prop/statement/P18",
        "endpoint": "wikidata"
      },
      {
        "label": "wikipediaPage",
        "iri": "http://schema.org/about",
        "inv": true,
        "restrictions": [
```

```
          "?value <http://schema.org/inLanguage> \"en\" ;\n  <http://schema.org/isPartOf> <https://en.wikipedia.org/> ."
        ],
        "endpoint": "wikidata"
      },
      {
        "label": "wikispeciesPage",
        "iri": "http://schema.org/about",
        "inv": true,
        "restrictions": [
          "?value  <http://schema.org/inLanguage> \"en\" ;\n  <http://schema.org/isPartOf> <https://species.wikimedia.org
               /> ."
        ],
        "endpoint": "wikidata"
      },
      {
        "label": "gbifPage",
        "iri": "http://www.wikidata.org/prop/direct/P846",
        "restrictions": [ "BIND(URI(CONCAT(\"https://www.gbif.org/species/\", ?value)) AS ?gbif)" ],
        "altResult": "?gbif",
        "altVariable": "gbif",
        "endpoint": "wikidata"
      }
    ],
    "dprops": [
      {
        "label": "scientificName",
        "iri": "http://www.wikidata.org/prop/direct/P225",
        "endpoint": "wikidata"
      },
      {
        "label": "vulgarName",
        "iri": "http://www.wikidata.org/prop/P1843>/<http://www.wikidata.org/prop/statement/P1843",
        "restrictions": [
          "FILTER(LANG(?value) = 'en' || LANG(?value) = 'es')"
        ],
        "endpoint": "wikidata"
      },
      {
        "label": "comment",
        "iri": "http://www.w3.org/2002/07/owl#sameAs>/<http://www.w3.org/2000/01/rdf-schema#comment",
        "inv": true,
        "restrictions": [
          "FILTER(LANG(?value) = 'en' || LANG(?value) = 'es')"
        ],
        "endpoint": "dbpedia"
      },
      {
        "label": "isSpecies",
        "iri": "http://www.wikidata.org/prop/direct/P105",
        "altResult": "?isSpecies",
        "altVariable": "isSpecies",
        "restrictions": [
          "BIND ( sameTerm(?value, <http://www.wikidata.org/entity/Q7432>) as ?isSpecies )"
        ],
        "endpoint": "wikidata"
      },
      {
        "label": "isConifer",
        "iri": "http://www.wikidata.org/prop/direct/P171",
        "altResult": "(true as ?isConifer)",
        "altVariable": "isConifer",
        "restrictions": [
          "?value <http://www.wikidata.org/prop/direct/P171>+ <http://www.wikidata.org/entity/Q133712> ."
        ],
        "endpoint": "wikidata"
      }
    ],
    "types": []
  }
],
"queryTemplates": [
  {
    "id": "subclasses",
    "description": "Obtain subclass relations between pairs of clases (variables \"sup\" and \"sub\") from an ancestor
         class (parameter \"ancestor\")",
    "template": "SELECT DISTINCT ?sup ?sub WHERE {
```

```
          ?sup rdfs:subClassOf* <{{{ancestor}}}> .
          ?sub rdfs:subClassOf ?sup . }",
    "variables": [
      "sup",
      "sub"
    ],
    "parameters": [
      {
        "label": "ancestor",
        "type": "iri"
      }
    ],
    "endpoint": "educawood"
  },
  {
    "id": "infoClasses",
    "description": "Obtain info about all the classes (variable \"class\") from an ancestor class (parameter \"ancestor
        \")",
    "template": "SELECT DISTINCT ?class ?parent ?labes ?laben ?comes ?comen WHERE {
       ?class rdfs:subClassOf* <{{{ancestor}}}> ;
         rdfs:subClassOf ?parent .
        OPTIONAL {
          ?class rdfs:label ?labes .
          FILTER (lang(?labes) = \"es\")
         }
         OPTIONAL {
           ?class rdfs:label ?laben .
           FILTER (lang(?laben) = \"en\")
          }
         OPTIONAL {
           ?class rdfs:comment ?comes .
           FILTER (lang(?comes) = \"es\")
          }
          OPTIONAL {
            ?class rdfs:comment ?comen
            FILTER (lang(?comen) = \"en\")
          } }",
    "variables": [
      "class",
      "parent",
      "labes",
      "laben",
      "comes",
      "comen"
    ],
    "parameters": [
      {
        "label": "ancestor",
        "type": "iri"
      }
    ],
    "endpoint": "educawood"
  },
  {
    "id": "counttreesinbox",
    "description": "Count the number of trees (variable \"count\") of an optional species (variable \"species\") in a
        bounding box with GPS coordinates \"latsouth\", \"latsouth\", \"latsouth\", and \"latsouth\"",
    "template": "SELECT COUNT(distinct ?tree) AS ?count WHERE {
      ?tree a <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Tree> ;
          <http://crossforest.eu/position/ontology/hasPosition> ?pos .
      {{#species}}?tree a/rdfs:subClassOf*<{{{species}}}> .{{/species}}
      ?pos <http://crossforest.eu/position/ontology/hasCoordinateReferenceSystem> <http://epsg.w3id.org/data/crs/4326> ;
          <http://epsg.w3id.org/ontology/axis/106> ?lat ;
          <http://epsg.w3id.org/ontology/axis/107> ?lng .
       {{#latnorth}}  FILTER (?lat <= {{latnorth}}) .{{/latnorth}}
       {{^latnorth}}  FILTER (?lat <= 0) .{{/latnorth}}
       {{#latsouth}}  FILTER (?lat > {{latsouth}}) .{{/latsouth}}
       {{^latsouth}}  FILTER (?lat > 0) .{{/latsouth}}
       {{#lngeast}}  FILTER (?lng <= {{lngeast}}) .{{/lngeast}}
       {{^lngeast}}  FILTER (?lng <= 0) .{{/lngeast}}
       {{#lngwest}}  FILTER (?lng > {{lngwest}}) .{{/lngwest}}
       {{^lngwest}}  FILTER (?lng > 0) .{{/lngwest}} }",
    "variables": [
      "count"
    ],
    "parameters": [
```

```
          {
            "label": "lngwest",
            "type": "number",
            "optional": true
          },
          {
            "label": "lngeast",
            "type": "number",
            "optional": true
          },
          {
            "label": "latnorth",
            "type": "number",
            "optional": true
          },
          {
            "label": "latsouth",
            "type": "number",
            "optional": true
          },
          {
            "label": "species",
            "type": "iri",
            "optional": true
          }
        ],
        "endpoint": "crossforest"
      },
      {
        "id": "treesinbox",
        "description": "Obtain trees (variable \"tree\") of an optional species (parameter \"species\") along with their GPS
             coordinates (variables \"lat\" and \"lng\"), diameters (\"dbh1mm\" and \"dbh2mm\"), height (\"heightM\") and
             species (\"type\") a in a bounding box with GPS coordinates \"latsouth\", \"latsouth\", \"latsouth\", and \"
             latsouth\". This template query can be paginated with the optional parameters \"limit\" and \"offset\"",
        "template": "SELECT DISTINCT ?tree ?lat ?lng ?dbh1mm ?dbh2mm ?heightM ?type WHERE {
           ?tree a <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Tree> ;
               <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasDBH1InMillimeters> ?dbh1mm ;
               <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasDBH2InMillimeters> ?dbh2mm ;
               <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/hasTotalHeightInMeters> ?heightM ;
                <http://crossforest.eu/position/ontology/hasPosition> ?pos .
           {{#species}} ?tree a/rdfs:subClassOf* <{{{species}}}> . {{/species}}
           OPTIONAL {
               ?tree a ?type .
               ?type a/rdfs:subClassOf <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Taxon> .
            }
            ?pos <http://crossforest.eu/position/ontology/hasCoordinateReferenceSystem> <http://epsg.w3id.org/data/crs/4326>
                 ;
                <http://epsg.w3id.org/ontology/axis/106> ?lat ;
                <http://epsg.w3id.org/ontology/axis/107> ?lng .
            {{#latnorth}}  FILTER (?lat <= {{latnorth}}) .{{/latnorth}}
            {{^latnorth}}  FILTER (?lat <= 0) .{{/latnorth}}
            {{#latsouth}}  FILTER (?lat > {{latsouth}}) .{{/latsouth}}
            {{^latsouth}}  FILTER (?lat > 0) .{{/latsouth}}
            {{#lngeast}}  FILTER (?lng <= {{lngeast}}) .{{/lngeast}}
            {{^lngeast}}  FILTER (?lng <= 0) .{{/lngeast}}
            {{#lngwest}}  FILTER (?lng > {{lngwest}}) .{{/lngwest}}
            {{^lngwest}}  FILTER (?lng > 0) .{{/lngwest}}
          }
          {{#limit}}LIMIT {{limit}}{{/limit}}{{^limit}}LIMIT 100{{/limit}}
          {{#offset}}OFFSET {{offset}}{{/offset}}",
        "variables": [
          "tree",
          "lat",
          "lng",
          "dbh1mm",
          "dbh2mm",
          "heightM",
          "type"
        ],
        "parameters": [
          {
            "label": "species",
            "type": "iri",
            "optional": true
          },
          {
```

```
        "label": "lngwest",
        "type": "number",
        "optional": true
      },
      {
        "label": "lngeast",
        "type": "number",
        "optional": true
      },
      {
        "label": "latnorth",
        "type": "number",
        "optional": true
      },
      {
        "label": "latsouth",
        "type": "number",
        "optional": true
      },
      {
        "label": "limit",
        "type": "integer",
        "optional": true
      },
      {
        "label": "offset",
        "type": "integer",
        "optional": true
      }
    ],
    "endpoint": "crossforest"
  },
  {
    "id": "treesinboxbasico",
    "description": "Obtain trees (variable \"tree\") of an optional species (parameter \"species\") and their GPS
        coordinates (variables \"lat\" and \"lng\") in a bounding box with GPS coordinates \"latsouth\", \"latsouth\",
        \"latsouth\", and \"latsouth\". This template query can be paginated with the optional parameters \"limit\" and
        \"offset\"",
    "template": "SELECT DISTINCT ?tree ?lat ?lng WHERE {
      ?tree a <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Tree> ;
          <http://crossforest.eu/position/ontology/hasPosition> ?pos .
      {{#species}} ?tree a/rdfs:subClassOf* <{{{species}}}> .{{/species}}
      ?pos <http://crossforest.eu/position/ontology/hasCoordinateReferenceSystem> <http://epsg.w3id.org/data/crs/4326> ;
          <http://epsg.w3id.org/ontology/axis/106> ?lat ;
          <http://epsg.w3id.org/ontology/axis/107> ?lng .
       {{#latnorth}}  FILTER (?lat <= {{latnorth}}) .{{/latnorth}}
       {{^latnorth}}  FILTER (?lat <= 0) .{{/latnorth}}
       {{#latsouth}}  FILTER (?lat > {{latsouth}}) .{{/latsouth}}
       {{^latsouth}}  FILTER (?lat > 0) .{{/latsouth}}
       {{#lngeast}}  FILTER (?lng <= {{lngeast}}) .{{/lngeast}}
       {{^lngeast}}  FILTER (?lng <= 0) .{{/lngeast}}
       {{#lngwest}}  FILTER (?lng > {{lngwest}}) .{{/lngwest}}
       {{^lngwest}}  FILTER (?lng > 0) .{{/lngwest}}
      }
      {{#limit}}LIMIT {{limit}}{{/limit}}{{^limit}}LIMIT 100{{/limit}}
      {{#offset}}OFFSET {{offset}}{{/offset}}",
    "variables": [
      "tree",
      "lat",
      "lng"
    ],
    "parameters": [
      {
        "label": "species",
        "type": "iri",
        "optional": true
      },
      {
        "label": "lngwest",
        "type": "number",
        "optional": true
      },
      {
        "label": "lngeast",
        "type": "number",
        "optional": true
```

```
        },
        {
          "label": "latnorth",
          "type": "number",
          "optional": true
        },
        {
          "label": "latsouth",
          "type": "number",
          "optional": true
        },
        {
          "label": "limit",
          "type": "integer",
          "optional": true
        },
        {
          "label": "offset",
          "type": "integer",
          "optional": true
        }
      ],
      "endpoint": "crossforest"
    },
    {
      "id": "counteducatreesinbox",
      "description": "Count the number of educatreestrees (variable \"count\") of an optional species (variable \"species
           \") in a bounding box with GPS coordinates \"latsouth\", \"latsouth\", \"latsouth\", and \"latsouth\"",
      "template": "SELECT COUNT(distinct ?tree) AS ?count WHERE {
        ?tree a <http://educawood.gsic.uva.es/sta/ontology/Tree> ;
            <http://educawood.gsic.uva.es/sta/ontology/hasPrimaryPosition> ?pos .
        {{#species}}
          ?tree <http://educawood.gsic.uva.es/sta/ontology/hasPrimarySpecies> ?ps .
          ?ps <http://educawood.gsic.uva.es/sta/ontology/hasTaxon>/rdfs:subClassOf* <{{{species}}}> .
        {{/species}}
        ?pos  <http://www.w3.org/2003/01/geo/wgs84_pos#lat> ?lat ;
           <http://www.w3.org/2003/01/geo/wgs84_pos#long> ?lng .
         {{#latnorth}}  FILTER (?lat <= {{latnorth}}) .{{/latnorth}}
         {{^latnorth}}  FILTER (?lat <= 0) .{{/latnorth}}
         {{#latsouth}}  FILTER (?lat > {{latsouth}}) .{{/latsouth}}
         {{^latsouth}}  FILTER (?lat > 0) .{{/latsouth}}
         {{#lngeast}}  FILTER (?lng <= {{lngeast}}) .{{/lngeast}}
         {{^lngeast}}  FILTER (?lng <= 0) .{{/lngeast}}
         {{#lngwest}}  FILTER (?lng > {{lngwest}}) .{{/lngwest}}
         {{^lngwest}}  FILTER (?lng > 0) .{{/lngwest}} }",
      "variables": [
        "count"
      ],
      "parameters": [
        {
          "label": "lngwest",
          "type": "number",
          "optional": true
        },
        {
          "label": "lngeast",
          "type": "number",
          "optional": true
        },
        {
          "label": "latnorth",
          "type": "number",
          "optional": true
        },
        {
          "label": "latsouth",
          "type": "number",
          "optional": true
        },
        {
          "label": "species",
          "type": "iri",
          "optional": true
        }
      ],
      "endpoint": "educawood"
```

```
},
{
  "id": "educatreesinbox",
  "description": "Obtain educatrees (variable \"tree\") of an optional primary species (parameter \"species\") and
       their GPS coordinates corresponding to their primary location (variables \"lat\" and \"lng\") in a bounding box
       with GPS coordinates \"latsouth\", \"latsouth\", \"latsouth\", and \"latsouth\". This template query can be
       paginated with the optional parameters \"limit\" and \"offset\"",
  "template": "SELECT DISTINCT ?tree ?lat ?lng WHERE {
    ?tree a <http://educawood.gsic.uva.es/sta/ontology/Tree> ;
        <http://educawood.gsic.uva.es/sta/ontology/hasPrimaryPosition> ?pos .
    {{#species}}
      ?tree <http://educawood.gsic.uva.es/sta/ontology/hasPrimarySpecies> ?ps .
      ?ps <http://educawood.gsic.uva.es/sta/ontology/hasTaxon>/rdfs:subClassOf* <{{{species}}}> .
    {{/species}}
    ?pos  <http://www.w3.org/2003/01/geo/wgs84_pos#lat> ?lat ;
       <http://www.w3.org/2003/01/geo/wgs84_pos#long> ?lng .
     {{#latnorth}}  FILTER (?lat <= {{latnorth}}) .{{/latnorth}}
     {{^latnorth}}  FILTER (?lat <= 0) .{{/latnorth}}
     {{#latsouth}}  FILTER (?lat > {{latsouth}}) .{{/latsouth}}
     {{^latsouth}}  FILTER (?lat > 0) .{{/latsouth}}
     {{#lngeast}}  FILTER (?lng <= {{lngeast}}) .{{/lngeast}}
     {{^lngeast}}  FILTER (?lng <= 0) .{{/lngeast}}
     {{#lngwest}}  FILTER (?lng > {{lngwest}}) .{{/lngwest}}
     {{^lngwest}}  FILTER (?lng > 0) .{{/lngwest}}
    }
    {{#limit}}LIMIT {{limit}}{{/limit}}{{^limit}}LIMIT 100{{/limit}}
    {{#offset}}OFFSET {{offset}}{{/offset}}",
  "variables": [
    "tree",
    "lat",
    "lng"
  ],
  "parameters": [
    {
      "label": "species",
      "type": "iri",
      "optional": true
    },
    {
      "label": "lngwest",
      "type": "number",
      "optional": true
    },
    {
      "label": "lngeast",
      "type": "number",
      "optional": true
    },
    {
      "label": "latnorth",
      "type": "number",
      "optional": true
    },
    {
      "label": "latsouth",
      "type": "number",
      "optional": true
    },
    {
      "label": "limit",
      "type": "integer",
      "optional": true
    },
    {
      "label": "offset",
      "type": "integer",
      "optional": true
    }
  ],
  "endpoint": "educawood"
},
{
  "id": "usernick",
  "description": "Obtain the user (variable \"user\") with a specific nick (parameter \"nick\")",
  "template": "SELECT DISTINCT ?user WHERE {
      ?user <http://xmlns.com/foaf/0.1/nick> \"{{nick}}\" }",
```

```
    "variables": [
      "user"
    ],
    "parameters": [
      {
        "label": "nick",
        "type": "string",
        "optional": false
      }
    ],
    "endpoint": "educawood"
  },
  {
    "id": "mostRecentEducatrees",
    "description": "Obtain the most recent educatrees (variable \"tree\") with their creation date (variable \"date\"),
         tree nick (if exists with variable \"tnick\"), creator (variable \"creator\"), and nick (if exists with variable
          \"unick\"), that have been optionally created by a user (parameter \"user\"). This template query can be
         paginated with the optional parameters \"limit\" and \"offset\"",
    "template": "SELECT DISTINCT ?tree ?date ?tnick ?creator ?unick WHERE {
       {{#user}}  VALUES ?creator { <{{{.}}}> }{{/user}}
       ?tree a <http://educawood.gsic.uva.es/sta/ontology/Tree> ;
           <http://purl.org/dc/terms/creator> ?creator ;
           <http://purl.org/dc/terms/created> ?date .
        OPTIONAL { ?tree <http://xmlns.com/foaf/0.1/nick> ?tnick }
        OPTIONAL {?creator <http://xmlns.com/foaf/0.1/nick> ?unick}
      }
      ORDER BY DESC(?date)
      {{#limit}}LIMIT {{.}}{{/limit}}{{^limit}}LIMIT 10{{/limit}}
      {{#offset}}OFFSET {{.}}{{/offset}}",
    "variables": [
      "tree",
      "date",
      "tnick",
      "creator",
      "unick"
    ],
    "parameters": [
      {
        "label": "user",
        "type": "iri",
        "optional": true
      },
      {
        "label": "limit",
        "type": "integer",
        "optional": true
      },
      {
        "label": "offset",
        "type": "integer",
        "optional": true
      }
    ],
    "endpoint": "educawood"
  },
  {
    "id": "mostRecentAnnotations",
    "description": "Obtain the most recently annotated educatrees (variable \"tree\") with their annotation type (
         variable \"annType\"), date (variable \"date\"), tree nick (if exists with variable \"tnick\"), annotator (
         variable \"annotator\") and nick (if exists with variable \"unick\"), that have been optionally annotated by a
         user (parameter \"user\"). This template query can be paginated with the optional parameters \"limit\" and \"
         offset\"",
    "template": "SELECT DISTINCT ?tree ?annType ?date ?tnick ?annotator ?unick WHERE {
        {{#user}} VALUES ?annotator { <{{{.}}}> }{{/user}}
        ?tree a <http://educawood.gsic.uva.es/sta/ontology/Tree> ;
            ?pr ?ann .
        ?pr rdfs:subPropertyOf <http://educawood.gsic.uva.es/sta/ontology/hasAnnotation> .
        ?ann a ?annType ;
            <http://purl.org/dc/terms/creator> ?annotator ;
            <http://purl.org/dc/terms/created> ?date .
         OPTIONAL {?tree <http://xmlns.com/foaf/0.1/nick> ?tnick}
         OPTIONAL {?annotator <http://xmlns.com/foaf/0.1/nick> ?unick}
       }
       ORDER BY DESC(?date)
       {{#limit}}LIMIT {{.}}{{/limit}}{{^limit}}LIMIT 10{{/limit}}
       {{#offset}}OFFSET {{.}}{{/offset}}",
```

```
      "variables": [
        "tree",
        "annType",
        "date",
        "tnick",
        "annotator",
        "unick"
      ],
      "parameters": [
        {
          "label": "user",
          "type": "iri",
          "optional": true
        },
        {
          "label": "limit",
          "type": "integer",
          "optional": true
        },
        {
          "label": "offset",
          "type": "integer",
          "optional": true
        }
      ],
      "endpoint": "educawood"
    },
    {
      "id": "validTaxons",
      "description": "Obtain the subset of the Wikidata entities (parameter \"eiri\") which are valid tree species, genera
           or families. Additionally, the sitelinks (variable \"sitelinks\") and statements (variable \"statements\") of
           the valid taxons are extracted.",
      "template": "SELECT ?taxon ?sitelinks ?statements WHERE {
         VALUES ?taxon { {{#eiri}} <{{{.}}}>{{/eiri}} }
         VALUES ?ranks {<http://www.wikidata.org/entity/Q7432> <http://www.wikidata.org/entity/Q34740> <http://www.wikidata
              .org/entity/Q35409>}
         VALUES ?plantclasses { <http://www.wikidata.org/entity/Q25314> <http://www.wikidata.org/entity/Q133712>}
         ?taxon wdt:P31 <http://www.wikidata.org/entity/Q16521> ;
             wdt:P105 ?ranks ;
             wdt:P171+ ?plantclasses ;
          <http://wikiba.se/ontology#sitelinks> ?sitelinks ;
          <http://wikiba.se/ontology#statements> ?statements .}",
      "variables": [
        "taxon",
        "sitelinks",
        "statements"
      ],
      "parameters": [
        {
          "label": "eiri",
          "type": "iri[]"
        }
      ],
      "endpoint": "wikidata"
    },
    {
      "id": "existingWikidataTaxons",
      "description": "Obtain the subset of the Wikidata taxons (parameter and variable \"wdiri\") with a corresponding
           taxon (variable \"txiri\") in the dataset. This is for checking if a taxon is already imported.",
      "template": "SELECT DISTINCT ?wdiri ?txiri WHERE {
        VALUES ?wdiri { {{#wdiri}} <{{{.}}}>{{/wdiri}} }
        VALUES ?plantclasses { <https://datos.iepnb.es/def/sector-publico/medio-ambiente/ifn/Class2> <https://datos.iepnb.
             es/def/sector-publico/medio-ambiente/ifn/Class1> }
        ?txiri rdfs:subClassOf* ?plantclasses .
        ?txiri rdfs:subClassOf ?wdiri . }",
      "variables": [
        "txiri",
        "wdiri"
      ],
      "parameters": [
        {
          "label": "wdiri",
          "type": "iri[]"
        }
      ],
      "endpoint": "educawood"
```

```
    }
  ]
}
```

References

[1] O. Derevenskaia, Active Learning Methods in Environmental Education of Students, *Procedia - Social and Behavioral Sciences* **131** (2014). doi:10.1016/j.sbspro.2014.04.086.

[2] S.C. Cheng, G.J. Hwang and C.H. Chen, From reflective observation to active learning: A mobile experiential learning approach for environmental science education, *British Journal of Educational Technology* **50** (2019). doi:10.1111/bjet.12845.

[3] German Federal Ministry of Food and Agriculture, Integrate+ - M-learning tools, Last visited September 2025. http://www.integrateplus.org/m-learning-tools.html.

[4] Observation International, Welcome - observation.org, 2025, Last visited September 2025. https://observation.org/.

[5] iNaturalist, A Community for Naturalists · iNaturalist, Last visited September 2025. https://www.inaturalist.org/.

[6] C.C. Charalampidis and E.A. Keramopoulos, Semantic Web user interfaces – A model and a review, *Data & Knowledge Engineering* **115** (2018), 214–227. doi:10.1016/j.datak.2018.04.003.

[7] A.-S. Dadzie and M. Rowe, Approaches to visualising Linked Data: a survey, *Semantic Web* **2**(2) (2011), 89–124. doi:10.3233/SW-2011-0037.

[8] T. Heath, J. Domingue and P. Shabajee, User interaction and uptake challenges to successfully deploying Semantic Web technologies, in: *Proceedings of the 3rd International Semantic Web User Interaction Workshop (SWUI2006), co-located with the 5th International Semantic Web Conference*, Athens, GA, USA, 2006.

[9] R. Verborgh, M. Vander Sande, O. Hartig, J. Van Herwegen, L. De Vocht, B. De Meester, G. Haesendonck and P. Colpaert, Triple Pattern Fragments: a Low-cost Knowledge Graph Interface for the Web, *Journal of Web Semantics* **37–38** (2016), 184–206. doi:10.1016/j.websem.2016.03.003.

[10] A.-S. Dadzie and E. Pietriga, Visualisation of Linked Data – Reprise, *Semantic Web* **8**(1) (2017), 1–21. doi:10.3233/SW-160249.

[11] L.V.D. Brink, P. Barnaghi, J. Tandy, G. Atemezing, R. Atkinson, B. Cochrane, Y. Fathy, R. García Castro, A. Haller, A. Harth, K. Janowicz, S. Kolozali, B.V. Leeuwen, M. Lefrançois, J. Lieberman, A. Perego, D. Le-Phuoc, B. Roberts, K. Taylor and R. Troncy, Best Practices for Publishing, Retrieving, and Using Spatial Data on the Web, *Semantic Web* **10** (2019), 95–114. doi:10.3233/SW-180305.

[12] G. Vega-Gorgojo, CRAFTS: Configurable REST APIs for triple stores, *IEEE Access* **10** (2022), 32426–32441. doi:10.1109/ACCESS.2022.3160610.

[13] Ministerio para la Transición Ecológica y el Reto Demográfico, Gobierno de España, Inventario Forestal Nacional, Last visited September 2025. https://www.miteco.gob.es/es/biodiversidad/temas/inventarios-nacionales/inventario-forestal-nacional.html.

[14] Wikidata, Wikidata, 2023, Last visited September 2025. https://www.wikidata.org/.

[15] DBpedia Association, Home - DBpedia Association, 2025, Last visited September 2025. https://www.dbpedia.org/.

[16] European Forest Institute, European Forest Institute, 2025, Last visited September 2025. https://efi.int.

[17] A. Bravo-Oviedo, M. Marchi, D. Travaglini, F. Pelleri, M.C. Manetti, P. Corona, F. Cruz, F. Bravo and S. Nocentini, Adoption of new silvicultural methods in Mediterranean forests: the influence of educational background and sociodemographic factors on marker decisions, *Annals of Forest Science* **77** (2020), 48, DOI: 10.1007/s13595-020-00947-z.

[18] G. Segalina, C. Dang and R. Grado, thinning scenarios to reconcile biodiversity conservation and socio-economic cobenefits in protected forest of Vietnam: effects on habitat value and timber yield, *Asian Journal of Forestry* **4** (2020), 22–35. doi:10.13057/asianjfor/r040105.

[19] C. Herodotou, N. Ismail, A.I. Benavides-Lahnstein, M. Aristeidou, A.N. Young, R.F. Johnson, L.M. Higgins, M. Ghadiri-Khanaposhtani, L.D. Robinson and H.L. Ballard, Young people in iNaturalist: a blended learning framework for biodiversity monitoring, *International Journal of Science Education, Part B* **0**(0) (2023), 1–28. doi:10.1080/21548455.2023.2217472.

[20] S. Unger, M. Rollins, A. Tietz and H. Dumais, iNaturalist as an engaging tool for identifying organisms in outdoor activities, *Journal of Biological Education* **55**(5) (2021), 537–547. doi:10.1080/00219266.2020.1739114.

[21] V. Ivanova, P. Lambrix, S. Lohmann and C. Pesquita, Visualization and interaction for ontologies and linked data – Editorial, *Journal of Web Semantics* **55** (2019), 145–149. doi:10.1016/j.websem.2018.10.001.

[22] E. Hyvönen, Digital humanities on the semantic web: Sampo model and portal series, *Semantic Web* **14**(4) (2023), 729–744. doi:10.3233/SW-223034.

[23] G. Vega-Gorgojo, LOD4Culture: Easy exploration of cultural heritage linked open data, *Semantic Web* **15** (2024), 1563–1592. doi:10.3233/SW-233358.

[24] T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer and D. Sheets, Tabulator: Exploring and analyzing Linked Data on the Semantic Web, in: *Proceedings of the 3rd International Semantic Web User Interaction Workshop*, Athens, Georgia, 2006.

[25] P. Andrews, I. Zaihrayeu and J. Pane, A Classification of Semantic Annotation Systems, *Semantic Web* **3**(3) (2012), 223–248. doi:10.3233/SW-2011-0056.

[26] P. García-Zarza, M.L. Bote-Lorenzo, G. Vega-Gorgojo and J.I. Asensio-Pérez, CHEST: A Linked Open Data-based Application to Annotate and Carry Out Learning Tasks About Cultural Heritage, in: *Educating for a New Future: Making Sense of Technology-Enhanced Learning Adoption*, I. Hilliger, P.J. Muñoz-Merino, T.D. Laet, A. Ortega-Arranz and T. Farrell, eds, Springer International Publishing, 2022, pp. 441–447. ISBN 978-3-031-16290-9. doi:10.1007/978-3-031-16290-9_34.

[27] E. Koutsiana, G.M.R. Amaral, N. Reeves, A. Meroño-Peñuela and E. Simperl, An analysis of discussions in collaborative knowledge engineering through the lens of Wikidata, *Journal of Web Semantics* **78** (2023). doi:10.1016/j.websem.2023.100799.

[28] G. Kellogg, P.A. Champin and D. Longley, JSON-LD 1.1: A JSON-based Serialization for Linked Data, Recommendation, W3C, 2020, URL: https://www.w3.org/TR/2020/REC-json-ld11-20200716/, last visited July 2024.

[29] P. Lisena, A. Meroño-Peñuela, T. Kuhn and R. Troncy, Easy Web API Development with SPARQL Transformer, in: *Proceedings of the 18th International Semantic Web Conference (ISWC 2019)*, C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. Cruz, A. Hogan, J. Song, M. Lefrançois and F. Gandon, eds, LNCS, Vol. 11779, Springer, Cham, Switzerland, 2019, pp. 454–470.

[30] M. Daquino, I. Heibi, S. Peroni and D. Shotton, Creating Restful APIs over SPARQL endpoints with RAMOSE, *Semantic Web* **13**(2) (2022), 195–213. doi:10.3233/SW-210439.

[31] C. Badenes-Olmedo, P. Espinoza-Arias and O. Corcho, RESTful-API for RDF data (R4R), in: *Proceedings of the ISWC 2021 Posters & Demonstrations and Industry Tracks co-located with 20th International Semantic Web Conference (ISWC 2021)*, O. Seneviratne, C. Pesquita, J. Sequeda and L. Etcheverry, eds, CEUR Workshop Proceedings, Vol. 2980, Aachen, Germany, 2021.

[32] D. Garijo and M. Osorio, OBA: An Ontology-Based Framework for Creating REST APIs for Knowledge Graphs, in: *Proceedings of the 19th International Semantic Web Conference (ISWC 2020)*, J.Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne and L. Kagal, eds, LNCS, Vol. 12507, Springer, Cham, Switzerland, 2020, pp. 48–64.

[33] A. Meroño-Peñuela and R. Hoekstra, grlc makes GitHub taste like linked data APIs, in: *Proceedings of the 13th European Semantic Web Conference (ESWC 2016)*, H. Sack, G. Rizzo, N. Steinmetz, D. Mladenić, S. Auer and C. Lange, eds, LNCS, Vol. 9989, Springer, Cham, Switzerland, 2016, pp. 342–353.

[34] E. Daga, L. Panziera and C. Pedrinaci, A BASILar approach for building web APIs on top of SPARQL endpoints, in: *Proceedings of the Third Workshop on Services and Applications over Linked APIs and Data (SALAD2015)*, Vol. 1359, Portoroz, Slovenia, 2015, pp. 22–32, co-located with the 12th European Semantic Web Conference (ESWC 2015).

[35] L. Dusseault, L. Lab and J. Snell, PATCH Method for HTTP, Standards Track, RFC 5789, Internet Engineering Task Force (IETF), 2010, URL: https://datatracker.ietf.org/doc/html/rfc5789, last visited July 2024.

[36] W. Beek, E. Folmer, L. Rietveld and J. Walker, GeoYASGUI: The GeoSPARQL query editor and result set visualizer, *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* **42** (2017), 39–42. doi:10.5194/isprs-archives-XLII-4-W2-39-2017.

[37] C. Nikolaou, K. Dogani, K. Bereta, G. Garbis, M. Karpathiotakis, K. Kyzirakos and M. Koubarakis, Sextant: Visualizing time-evolving linked geospatial data, *Journal of Web Semantics* **35** (2015), 35–52. doi:10.1016/j.websem.2015.09.004.

[38] M. Gaigg, *Designing Map Interfaces: Patterns for Building Effective Map Apps*, Esri Press, 2023.

[39] G. Vega-Gorgojo, J.M. Giménez-García, C. Ordóñez and F. Bravo, Pioneering easy-to-use forestry data with Forest Explorer, *Semantic Web* **13**(2) (2022), 147–162. doi:10.3233/SW-210430.

[40] J.M. Giménez-García, G. Vega-Gorgojo, C. Ordóñez, N. Crespo-Lera and F. Bravo, Improving availability and utilization of forest inventory and land use map data using Linked Open Data, *Frontiers in Forests and Global Change* **7** (2024).

[41] C. Stadler, J. Lehmann, K. Höffner and S. Auer, LinkedGeoData: A core for a web of spatial open data, *Semantic Web* **3**(4) (2012), 333–354. doi:10.3233/SW-2011-0052.

[42] A. de León, F. Wisniewski, B. Villazón-Terrazas and O. Corcho, Map4rdf – Faceted browser for geospatial datasets, in: *Proceedings of the First International Workshop on Open Data (WOD-2012)*, Nantes, France, 2012.

[43] D. Brickley, Basic Geo (WGS84 lat/long) Vocabulary, Technical Report, W3C Semantic Web Interest Group, 2006, URL: https://www.w3.org/2003/01/geo/, last visited September 2025.

[44] D. Brickley and L. Miller, FOAF Vocabulary Specification, Technical Report, 2014, Paddington Edition. URL: http://xmlns.com/foaf/spec/20140114.html, last visited September 2025.

[45] DCMI Usage Board, DCMI Metadata Terms, Recommendation, 2020, URL: https://www.dublincore.org/specifications/dublin-core/dcmi-terms/, last visited September 2025.

[46] S. Lohmann, S. Negru, F. Haag and T. Ertl, Visualizing ontologies with VOWL, *Semantic Web* **7**(4) (2016), 399–419. doi:10.3233/SW-150200.

[47] C. Maser, R.G. Anderson, K. Cromack, J.T. Williams and R.E. Martin, Dead and Down Woody Material, in: *Wildlife Habitats in Managed Forests the Blue Mountains of Oregon and Washington*, J.W. Thomas, ed., Agriculture Handbook, Vol. 553, U.S. Department of Agriculture, Forest Service., Washington, USA, 1979.
[48] M.L. Hunter (ed.), *Maintaning biodiversity in Forest Ecosystems*, Cambridge University Press, 2006.
[49] Esri, World Imagery, 2025, Last visited September 2025. https://www.arcgis.com/home/item.html?id=10df2279f9684e4a9f6a7f08febac2a9.
[50] E.A. Scott, *SPA Design and Architecture: Understanding single-page web applications*, Manning Publications, 2015.
[51] P. Gearon, A. Passant and A. Polleres, SPARQL 1.1 Update, Recommendation, W3C, 2013, URL: http://www.w3.org/TR/2013/REC-sparql11-update-20130321/, last visited July 2024.
[52] Cross-Forest, Home - Cross Forest, 2020, Last visited September 2025. https://crossforest.eu/.
[53] Wikispecies, Wikispecies, free species directory, 2025, Last visited September 2025. https://species.wikimedia.org.
[54] Wikimedia Foundation, Inc., Wikipedia, 2025, Last visited September 2025. https://www.wikipedia.org/.
[55] Global Biodiversity Information Facility, GBIF, 2025, Last visited September 2025. https://www.gbif.org.
[56] P. Bryan, JavaScript Object Notation (JSON) Patch, Standards Track, RFC 6902, Internet Engineering Task Force (IETF), 2013, URL: https://datatracker.ietf.org/doc/html/rfc6902, last visited July 2024.
[57] Mozilla Foundation, JavaScript modules - JavaScript | MDN, 2025, Last visited September 2025. https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Modules.
[58] npm, Inc., npm | Home, Last visited September 2025. https://www.npmjs.com/.
[59] D. Govett and Parcel Contributors, Parcel, 2025, Last visited September 2025. https://parceljs.org/.
[60] V. Agafonkin, Leaflet - a JavaScript library for interactive maps, 2025, Last visited September 2025. https://leafletjs.com/.
[61] OpenStreetMap contributors, OpenStreetMap, 2025, Last visited September 2025. https://www.openstreetmap.org.
[62] Bootstrap team, Bootstrap - The most popular HTML, CSS, and JS library in the world., Last visited September 2025. https://getbootstrap.com/.
[63] C. Wanstrath, {{ mustache }}, Last visited September 2025. https://mustache.github.io/.
[64] J. Ashkenas and Underscore.js Contributors, Underscore.js, 2024, Last visited September 2025. https://underscorejs.org/.
[65] Google, Firebase | Google's Mobile and Web App Development Platform, 2025, Last visited September 2025. https://firebase.google.com/.
[66] M. Jones and D. Hardt, The OAuth 2.0 Authorization Framework: Bearer Token Usage, Standards Track, RFC 6750, Internet Engineering Task Force (IETF), 2012, URL: https://datatracker.ietf.org/doc/html/rfc6750, last visited July 2024.
[67] The Apache Software Foundation, Welcome to Apache Solr - Apache Solr, 2025, Last visited September 2025. https://solr.apache.org/.
[68] G. Vega-Gorgojo, GitHub - EducaWood: Your social application for forestry education, 2025, Last visited September 2025. https://github.com/guiveg/educawood.
[69] G. Vega-Gorgojo, Home - EducaWood, 2023, Last visited September 2025. http://educawood.gsic.uva.es.
[70] J. Andrade-Hoz, G. Vega-Gorgojo, I. Ruano-Benito, M.L. Bote-Lorenzo, J.I. Asensio-Pérez, F. Bravo and C. Ordóñez, EducaWood: a Socio-Semantic Annotation System for Environmental Education, in: *Proceedings of the 16th European Conference on Technology Enhanced Learning (EC-TEL 2021)*, LNCS, Springer, Cham, Switzerland, 2021, pp. 368–372. doi:10.1007/978-3-030-86436-1_37.
[71] Ministerio para la Transformación Digital y de la Función Pública, Gobierno de España, UniversiData-Lab, Proyecto MIP y EducaWood ganan el III Desafío Aporta | datos.gob.es, 2021, Last visited September 2025. https://datos.gob.es/es/noticia/universidata-lab-proyecto-mip-y-educawood-ganan-el-iii-desafio-aporta.
[72] J. Nielsen, *Usability engineering*, Morgan Kaufman, 1993.
[73] J.W. Creswell and J.D. Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*, Sage publications, 2017.
[74] J. Brooke et al., SUS-A quick and dirty usability scale, *Usability evaluation in industry* **189**(194) (1996), 4–7.
[75] J. Saldaña, *The coding manual for qualitative researchers*, Sage publications, 2021.
[76] J. Sauro and J.R. Lewis, *Quantifying the user experience: Practical statistics for user research*, Morgan-Kaufmann, Amsterdam, Netherlands, 2012.
[77] L. Larrieu, Y. Paillet, S. Winter, R. Bütler, D. Kraus, F. Krumm, T. Lachat, A.K. Michel, B. Regnery and K. Vandekerkhove, Tree related microhabitats in temperate and Mediterranean European forests: A hierarchical typology for inventory standardization, *Ecological Indicators* **84** (2018), 194–207. doi:10.1016/j.ecolind.2017.08.051.
[78] J.T. Sample and E. Ioup, *Tile-based geospatial information systems: principles and practices*, Springer, 2010.
[79] C. Arthur, What is the 1% rule?, 2006, URL: https://www.theguardian.com/technology/2006/jul/20/guardianweeklytechnologysection2, last visited July 2024.