Enabling procurement data value chains for economic development, demand management, competitive markets and vendor intelligence

Deliverable D5.3
Procurement APIs and platform release v2

<table>
<thead>
<tr>
<th>Date</th>
<th>29/03/2019</th>
</tr>
</thead>
</table>
| Author(s)  | Francisco Yedro, Oscar Corcho, Carlos Badenes (UPM)  
Brian Elvesæter, Ahmet Soylu, Philip Turk (SINTEF) |
| Dissemination level | PU |
| Work package   | 5 |
| Version        | 1.0 |
Document metadata

Quality assurors and contributors

<table>
<thead>
<tr>
<th>Quality assuror(s)</th>
<th>Divna Djordjevic (CERVED)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contributor(s)</td>
<td>Francisco Yedro, Oscar Corcho, Carlos Badenes (UPM)</td>
</tr>
</tbody>
</table>

Version history

<table>
<thead>
<tr>
<th>Date</th>
<th>Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>15/01/2019</td>
<td>0.1</td>
<td>Initial ToC</td>
</tr>
<tr>
<td>01/02/2019</td>
<td>0.2</td>
<td>Initial version of the API catalogue</td>
</tr>
<tr>
<td>12/02/2019</td>
<td>0.3</td>
<td>Including examples of the methods</td>
</tr>
<tr>
<td>01/03/2019</td>
<td>0.4</td>
<td>OCDS Ontology and some data model added</td>
</tr>
<tr>
<td>15/03/2019</td>
<td>0.5</td>
<td>Conclusions added and internal review done, deliverable ready for external QA</td>
</tr>
<tr>
<td>26/03/2019</td>
<td>0.6</td>
<td>QA comments received and added to the document</td>
</tr>
<tr>
<td>03/04/2019</td>
<td>1.0</td>
<td>Search API documentation included</td>
</tr>
</tbody>
</table>
Executive summary

This document presents Deliverable D5.3 “Procurement APIs and platform release v2” of the TheyBuyForYou project. This deliverable is developed as part of Work Package 5 (WP5) “Standards, best practices, and integration”, task T5.2 “Design principles for procurement APIs”.

This document provides an intermediate version of the architectural diagram for the TheyBuyForYou platform and API, together with a catalogue of the current offering of API requests.

The TheyBuyForYou architecture has been developed following state-of-the-art principles in software development, considering a low decoupling among all the software components, the use of REST principles for data provisioning, and the consideration of a distributed data governance for all the data items that are relevant to procurement APIs.

More specifically, the APIs that are currently described are those that allow dealing with:

- Tenders and their related entities.
- Contracts.
- Contracting-process.
- Organisations.
- Awards.

This API catalogue will continue evolving over the course of the project and documented in upcoming deliverables from WP5. The API will provide access to all the resources that are relevant to provide a full-fledged procurement data platform that can support the business cases developed in the context of WP6, and which will be fully aligned with the TheyBuyForYou public procurement ontology network.
# Table of Contents

**DOCUMENT METADATA** .............................................................................................................. 2

**EXECUTIVE SUMMARY** .................................................................................................................. 3

**TABLE OF FIGURES** .......................................................................................................................... 5

1 **INTRODUCTION** .............................................................................................................................. 6

2 **THEBUYFORYOU ARCHITECTURE** ................................................................................................. 6

3 **THEBUYFORYOU CORE API CATALOGUE** ....................................................................................... 8

   3.1 **THEBUYFORYOU CORE API** .................................................................................................. 8

   3.1.1 General overview ..................................................................................................................... 8

   3.1.2 Data model ............................................................................................................................... 9

   3.1.3 Technology .............................................................................................................................. 11

   3.1.4 OCDS Ontology ...................................................................................................................... 12

   3.1.5 Resources ............................................................................................................................... 14

   3.1.5.1 /contracting-process/ .......................................................................................................... 14

   3.1.5.2 /contracting-process/:id/ .................................................................................................... 15

   3.1.5.3 /contracting-process/:id/award/ ........................................................................................ 19

   3.1.5.4 /contracting-process/:id/contract ....................................................................................... 20

   3.1.5.5 /organisation/:id/ ............................................................................................................... 20

   3.1.5.6 /organisation/:id/award/ .................................................................................................... 21

   3.1.5.7 /organisation/:id/contracting-process/ .............................................................................. 23

   3.1.5.8 /organisation/:id/contracts .............................................................................................. 24

   3.1.5.9 /contract/ ............................................................................................................................ 25

   3.1.5.10 /contract/:id/ ................................................................................................................... 26

   3.1.5.11 /contract/:id/amendment ................................................................................................. 28

   3.1.5.12 /contract/:id/document .................................................................................................... 29

   3.1.5.13 /contract/:id/item/ ........................................................................................................... 29

   3.1.5.14 /award/ ............................................................................................................................. 30

   3.1.5.15 /award/:id/ ....................................................................................................................... 31

   3.1.5.16 /award/:id/amendment ................................................................................................... 33

   3.1.5.17 /award/:id/item/ .............................................................................................................. 34

   3.1.5.18 /award/:id/supplier/ ......................................................................................................... 35

   3.1.5.19 /award/:id/document ....................................................................................................... 35

   3.1.5.20 /tender/ ............................................................................................................................. 36

   3.1.5.21 /tender/:id/ ...................................................................................................................... 37

   3.1.5.22 /tender/:id/item/ .............................................................................................................. 37

   3.1.5.23 /tender/:id/contracting-process/ .................................................................................... 44

   3.1.5.24 /tender/:id/document ...................................................................................................... 46

3.2 **THEBUYFORYOU SEARCH API** ................................................................................................. 54

   3.2.1 General overview ................................................................................................................... 54

   3.2.2 Data Model ............................................................................................................................ 55

   3.2.3 Technology ............................................................................................................................ 55

   3.2.4 Resources ............................................................................................................................. 56

   3.2.4.1 /documents/ ..................................................................................................................... 56

   3.2.4.2 /documents/:id/ .............................................................................................................. 57

   3.2.4.3 /items/ .............................................................................................................................. 57

   3.2.4.4 /items/:id/ ....................................................................................................................... 58

4 **API ACCOUNTS, AUTHENTICATION AND AUTHORISATION** ...................................................... 59

5 **CONCLUSIONS AND FUTURE WORK** ....................................................................................... 60

6 **ANNEX I. THEYBUYFORYOU CORE API CHEATSHEET FOR DEVELOPERS** .......................... 61
Table of Figures

FIGURE 1. THEYBUYFORYOU HIGH-LEVEL ARCHITECTURE ................................................................. 7
FIGURE 2. OVERVIEW OF THE TBFY DATA MODEL (CONTRACT) .................................................. 9
FIGURE 3. OVERVIEW OF THE TBFY DATA MODEL (ORGANISATION) ........................................ 9
FIGURE 4. OVERVIEW OF THE TBFY DATA MODEL (AWARD) ....................................................... 10
FIGURE 5. OVERVIEW OF THE TBFY DATA MODEL (TENDER) ....................................................... 10
FIGURE 6. CONTRACTING PROCESS CLASS SCHEME ................................................................... 12
FIGURE 7. TENDER CLASS SCHEME .................................................................................................. 12
FIGURE 8. AWARD CLASS SCHEME .................................................................................................. 13
FIGURE 9. CONTRACT CLASS SCHEME ............................................................................................... 13
FIGURE 10. OVERVIEW OF THE TBFY SEARCH API DATA MODEL .............................................. 55
1 Introduction

This document provides an intermediate version of the architectural diagram for the TheyBuyForYou platform and API, together with a catalogue of the current offering of API requests.

The document is structured as follows. In section 2, we provide a description of the current high-level architecture of the TheyBuyForYou platform, which allows understanding how data is governed in the context of the project and how it will be made accessible for business cases and external third-party developers. As part of the agile software development methodology that is being applied in the context of the project, as described in deliverable D5.1, the architecture will be evolving, so as to provide an up-to-date overview of the API offering. Such up-to-date version will be always available in the main GitHub repository of the project: http://github.com/tbfy/knowledge-graph-api. Section 3 describes the current catalogue of API requests and access mechanisms to data sources that are currently being offered by the TheyBuyForYou platform. Section 4 discusses on the authentication and authorisation mechanisms that are currently being applied for the existing APIs, which may still evolve in the course of the project. Section 5 discusses on the next steps towards the development of the TheyBuyForYou API.

2 TheyBuyForYou Architecture

The TheyBuyForYou architecture has been developed following state-of-the-art principles in software development, considering a low decoupling among all the software components, the use of REST principles for data provisioning, and the consideration of a distributed data governance for the data items that are relevant to procurement APIs. Our API catalogue is mostly focused on providing access mechanisms to any software developers (in-house or external) that want to make use of the knowledge graph. Therefore, they are mostly focused on providing access to such data through the HTTP GET verb and the API catalogue is organised around the main entities that are relevant for public procurement (and in more general terms, for the handling of the economic information available in public administrations): tenders, agents (companies, organisations, individuals), invoices, etc.

The final characterisation of the specific resources that will need to be handled in our knowledge graph (and hence in our API) will be determined by the development of the public procurement ontologies that is being done in the context of the project, and which is being delivered at the same time as this deliverable.

Figure 1 provides a general high-level overview of the current TheyBuyForYou architecture. On the left-hand side we include the extraction, transformation and load processes that is being done to incorporate data sources into the knowledge graph, mostly via the databases of OpenCorporates (for agents – organisations, companies, individuals -) and OpenOpps (for OCDS\(^1\)-related data). The task of incorporating such data sources has been already done as part of the normal practices of these two organisations, and the work from WP1 and WP2 is improving the coverage and quality of those data sources.

The transformation into the RDF that will be stored in the knowledge graph infrastructure are being done using the DataGraft\(^2\) platform, as well as other ad-hoc scripts when required. Transformations do not always cover the whole set of data available from these APIs, but in many cases just provide some partial views on the data, depending on the needs from the business cases and the characteristics of compulsory and optional attributes identified in the TheyBuyForYou ontology network.

On the right-hand side we provide an overview of the main data storage mechanisms, including a triple store for the RDF-based data and a document store for the documents associated to public procurement (tender notices, award notices, etc.). Access to the pre-existing APIs of OpenCorporates and OpenOpps are also made available as part of the knowledge graph infrastructure (in this case, the underlying infrastructure to serve data will be considered as a black box from the point of view of the TheyBuyForYou project). The knowledge graph will

---

\(^1\) Open Contracting Data Standard

\(^2\) https://datagraft.io/
include owl:sameAs links to the corresponding URIs, so as to facilitate navigation across these different data sources. Documents will be connected to the main entities of the knowledge graph using rdfs:seeAlso or another relevant property, once the ontology design has been stabilised.

The knowledge graph will be accessible via a Linked-Data enabled REST API. This means that the URIs that will be used to identify contracts, tenders, agents, invoices, etc. (that is, the main types of resources handled by the knowledge graph) will be de-referenceable and with content negotiation (at least providing RDF, JSON, JSON-LD, CSV and HTML views of the data). There will be also SPARQL endpoint access, for those developers willing to make ad-hoc queries to the knowledge graph, as well as a range of additional services to enable search over the knowledge graph and document store, and possibly reconciliation services to facilitate third parties the usage of TheyBuyForYou URIs (with the aim of increasing uptake). All these APIs will be available under an authentication/authorisation layer.

Finally, the range of online tools and business case implementations from the project (and also from any other outside contributions) will be based on this data access layer.

In the following section we describe the core API, which is based on the ontology network that is being developed in the context of the project (unlike in the initial version of the API, where we just took the APIs currently available from some of the consortium partners - OpenCorporates, OpenOpps, Zaragoza -). This core API will be extended in further stages of the project with additional added-value services and APIs provided by consortium partners and potentially by other third parties in the future.

---

Figure 1. TheyBuyForYou High-level Architecture
3 TheyBuyForYou Core API Catalogue

3.1 TheyBuyForYou Core API

In this section of the deliverable, as discussed, we present the first catalogue of API requests that we are developing, which we call core API. This core API catalogue is based on the current version of the ontology developed in the context of WP1, which is in its turn based on the Open Contracting Data Standard (OCDS) data model.

More specifically, the core API allows dealing with:

- Tenders and their related entities.
- Contracts.
- Contracting-process.
- Organisations.
- Awards.

3.1.1 General overview

The objective of this section is to provide a general overview of the main data models used by the API as well as the main types of resources and requests that are made available.

Among the main data offered by the core API, we can find information about the tenders, awards and everything related to the contracting process, as well as information from participating entities, such as their legal name, their identifiers and associated information. More details are provided in section 3.1.2. As an example, next we show the response for a sample award, available at http://tbfy.library.linkeddata.es/kg-api/award:

```
[{
  "id": "ocds-213czf-000-00001-award-01",
  "title": "Award of contract to build new cycle lanes in the center of town.",
  "description": "AnyCorp Ltd has been awarded the contract to build new cycle lanes in the center of town.",
  "datetime": {
    "date": "2010-05-10T10:30:00Z",
    "time": 0
  }
},
{
  "id": "ocds-213czf-000-00001-award-02",
  "title": "Award of contract to build new cycle lanes in the center of town.",
  "description": "AnyCorp Ltd has been awarded the contract to build new cycle lanes in the center of town.",
  "datetime": {
    "date": "2010-05-10T10:30:00Z",
    "time": 0
  }
}
]
```

In order to improve the understanding of the core API, we have developed a full-fledged mock-up using Postman. The current endpoint for this mock-up API is http://tbfy.library.linkeddata.es/kg-api and, for the time being, the API does not support versioning. The current YAML implementation for the core API is available at https://github.com/TBFY/knowledge-graph-API/blob/master/API_description.yaml
3.1.2 Data model

In this section we provide a brief description of the main elements in the data model of the API, which is graphically described in Figure 2, Figure 3, Figure 4 and Figure 5.

Figure 2. Overview of the TBFY Data Model (contract)

Figure 3. Overview of the TBFY Data Model (organisation)
Figure 4. Overview of the TBFY Data Model (award)

Figure 5. Overview of the TBFY Data Model (tender)
3.1.3 Technology

As aforementioned, we have already developed a mock-up of the core API so as to facilitate consortium partners as well as third parties the understanding of the main elements and types of requests involved in it, and so as to facilitate discussions around the API design. This is only intended to be a tool for discussion and for initial demos, and in the next stages of the project we will focus on the actual development of the API.

In this section we describe the technology that will be used to implement it. The API will be based on RESTful API technology which is an application program interface (API) that uses HTTP requests to GET (used to retrieve a resource’s representation), PUT (allows a client to modify a resource state), POST (used to create resources) and DELETE (requests a resource to be deleted) data. In this initial version of the API only GET requests will be implemented.

REST, which is considered as a standard for building web services, is aligned with the precepts of the web and provides many benefits that provide lot of features such as suitable actions (GET, POST, PUT, DELETE, HEAD, PATCH…), caching, security, redirection and forwarding.

So, building on top of HTTP, REST APIs provide the means to build flexible APIs that can:

- Support backward compatibility
- Evolvable APIs
- Scalable services
- Securable services
- A spectrum of stateless to stateful services

To carry out this task, a typical technology will be used in the development of APIs from Java, SPRING technology so, all the REST services will be built with SPRING\(^3\).

SPRING has become the de-facto standard for building Java based applications. SPRING is a java programming language framework with the aim of standardising the work and management of the problems that allows to develop applications in an easy and quickly way. The Spring Framework is open source.

---

\(^3\) https://spring.io/
3.1.4 OCDS Ontology

In this section we describe the ontology on which the API is based. This ontology is available at https://github.com/TBFY/ocds-ontology/. The ontology has been built on the basis of the original OCDS data model. It has four main classes, which are described below:

Contracting Process

The contracting process class represents a single contracting process at a particular point in time and contains information about new tenders, awards, contracts and other updates.

![Contracting Process class scheme](image)

Tender

The Tender class provides data regarding the tender process, which involves publicly inviting prospective contractors to submit bids for evaluation and selecting a winner or winners.

![Tender class scheme](image)
Award

The Award class corresponds to an award for the given procurement process. There may be more than one award per contracting process e.g. because the contract is split among different providers, or because it is a standing offer.

![Figure 8. Award class scheme](image)

Contract

The Contract class provides information regarding the signed contract between the buyer and supplier(s).

![Figure 9. Contract class scheme](image)

An in-depth description of the OCDS ontology and the TBFY Knowledge Graph will be provided in Deliverable 2.2: "Knowledge Graph Publication" (Month 18).
3.1.5 Resources

In this subsection we provide details of the methods implemented in the TBFY core API, related to the previous data model. A complete summary of these methods is provided in Annex I. All of these methods may use the HTTP verbs GET, PUT, POST and DELETE. The current mock-up only provides support for the verb GET.

3.1.5.1 /contracting-process/

This returns the list of contracting processes. To limit the results of the list, this call will receive two parameters to define the date. These parameters are:

- Before: string (ISO 8601)
- After: string (ISO 8601)

This release call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>ocid</td>
<td>A globally unique identifier for this Open Contracting Process. Composed of a publisher prefix and an identifier for the contracting process</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>id</td>
<td>An identifier for this particular release of information. A release identifier must be unique within the scope of its related contracting process (defined by a common ocid), and unique within any release package it appears in. A release identifier must not contain the # character.</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>name</td>
<td>A common name for this organization or other participant in the contracting process. The identifier object provides a space for the formal legal name, and so this may either repeat that value, or could provide the common name by which this organization or entity is known. This field may also include details of the department or sub-unit involved in this contracting process.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>datetime</td>
<td>The date this information was first released or published. Composed by two fields: Date: string (ISO 8601) Time: Integer</td>
<td>String</td>
<td>Date-time</td>
<td>True</td>
</tr>
</tbody>
</table>

Example:

```
http://tbfy.library.linkeddata.es/kg-api/contracting-process?before=2010-05-31T10:30:00Z&after=2010-05-01T10:30:00Z
```

```json
[
  {
    "ocid": "ocds-213czf-000-00001",
    "id": "GB-COH-1111111",
    "name": "AnyCorp Ltd",
    "datetime": {
      "date": "2010-05-10T10:30:00Z",
      "time": 0
    }
  },
  {
    "ocid": "ocds-213czf-000-00001",
  }
]
```
### 3.1.5.2 /contracting-process/:id

This returns the core data for the given contracting process. This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>ocid</td>
<td>A globally unique identifier for this Open Contracting Process. Composed of a publisher prefix and an identifier for the contracting process.</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>id</td>
<td>An identifier for this particular release of information. A release identifier must be unique within the scope of its related contracting process (defined by a common ocid), and unique within any release package it appears in. A release identifier must not contain the # character.</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>datetime</td>
<td>The date this information was first released or published. Composed by two fields: Date: string (ISO 8601) Time: Integer</td>
<td>String</td>
<td>Date-time</td>
<td>True</td>
</tr>
<tr>
<td>tag</td>
<td>One or more values from the releaseTag codelist. Tags may be used to filter release and to understand the kind of information that a release might contain.</td>
<td>array[</td>
<td></td>
<td></td>
</tr>
<tr>
<td>initationType</td>
<td>String specifying the type of initiation process used for this contract, taken from the initiationType codelist. Currently only tender is supported.</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>parties</td>
<td>Information on the parties (organizations, economic operators and other participants) who are involved in the contracting process and their roles, e.g. buyer, procuring entity, supplier etc. Organization references elsewhere in the schema are used to refer back to these entries in this list (see Organization).</td>
<td>array[</td>
<td></td>
<td></td>
</tr>
<tr>
<td>buyer</td>
<td>A buyer is an entity whose budget will be used to pay for goods, works or services related to a contract. This may be different from the procuring entity who may be specified in the tender data (see OrganizationReference).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>planning</td>
<td>Information from the planning phase of the contracting process. This includes information related to the process of deciding what to contract, when and how (see Planning).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tender</td>
<td>The activities undertaken in order to enter into a contract (see Tender).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>------------------------------------------------------------------------</td>
<td>--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>awards</td>
<td>Information from the award phase of the contracting process. There may be more than one award per contracting process e.g. because the contract is split among different providers, or because it is a standing offer (see Award).</td>
<td>array[Award]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>contracts</td>
<td>Information from the contract creation phase of the procurement process (see Contract).</td>
<td>array[Contract]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>language</td>
<td>Specifies the default language of the data using either two-letter ISO639-1, or extended BCP47 language tags. The use of lowercase two-letter codes from ISO639-1 is strongly recommended.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>relatedProcesses</td>
<td>If this process follows on from one or more prior process, represented under a separate open contracting identifier (ocid) then details of the related process can be provided here. This is commonly used to relate mini-competitions to their parent frameworks, full tenders to a pre-qualification phase, or individual tenders to a broad planning process (see RelatedProcess).</td>
<td>array[RelatedProcess]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

http://tbfy.library.linkeddata.es/kg-api/contracting-process/ocds-213czf-000-00001

```json
[
  {
    "ocid": "ocds-213czf-000-00001",
    "id": "ocds-213czf-000-00001-02-tender",
    "datetime": {
      "date": "2010-05-10T10:30:00Z",
      "time": 0
    },
    "tag": ["releasetag": "tender" ],
    "initiationtype": "tender",
    "parties": [
      {
        "id": 0,
        "legalname": "string"
      }
    ],
    "buyer": {
      "identifier": {
        "scheme": "GB-LAC",
        "id": "E09000003",
        "legalname": "London Borough of Barnet",
        "uri": "http://www.barnet.gov.uk/
      }
    }
  }
]```
"name": "London Borough of Barnet",
"address": {
  "streetAddress": "A, North London Business Park, Oakleigh Rd S",
  "locality": "London",
  "region": "London",
  "postalCode": "N11 1NP",
  "countryName": "United Kingdom"
},
"contactPoint": {
  "name": "Procurement Team",
  "email": "procurement-team@example.com",
  "telephone": "01234 345 346",
  "faxNumber": "01234 345 345",
  "url": "http://example.com/contact/
}
},
"planning": {},
"tender": {
  "id": "ocds-213czf-000-00001-01-tender",
  "title": "Planned cycle lane improvements",
  "description": "Tenders solicited for work to build new cycle lanes in the center of town.",
  "status": "active",
  "items": [
    {
      "id": "0001",
      "description": "string",
      "classification": {
        "scheme": "CPV",
        "id": "45233130",
        "description": "Construction work for highways",
        "uri": "http://cpv.data.ac.uk/code-45233130"
      },
      "additionalClassifications": [
        {
          "scheme": "CPV",
          "id": "45233162-2",
          "description": "Cycle path construction work",
          "uri": "http://cpv.data.ac.uk/code-45233162.html"
        }
      ],
      "quantity": 8,
      "unit": {
        "name": "Miles",
        "value": {
          "amount": 120000,
          "currency": "GBP"
        }
      }
    }
  ],
  "minValue": {
    "amount": 600000,
    "currency": "GBP"
  },
  "value": {
    "amount": 1100000,
    "currency": "GBP"
  },
  "procurementMethod": "open",
  "procurementMethodRationale": "An open competitive tender is required by EU Rules",
  "awardCriteria": "bestProposal",
  "awardCriteriaDetails": "The best proposal, subject to value for money requirements, will be accepted.",
  "submissionMethod": [
    "electronicSubmission"
  ]
}
"submissionMethodDetails": "Submit through the online portal at http://example.com/submissions/ocds-213czf-000-00001-01/",
"enquiryPeriod": {
  "startDate": "2010-03-01T09:00:00Z",
  "endDate": "2010-03-14T17:30:00Z"
},
"hasEnquiries": false,
"tenderPeriod": {
  "startDate": "2010-03-01T09:00:00Z",
  "endDate": "2011-04-01T18:00:00Z"
},
"awardPeriod": {
  "startDate": "2010-06-01T00:00:00Z",
  "endDate": "2011-08-01T23:59:59Z"
},
"procuringEntity": {
  "identifier": {
    "scheme": "GB-LAC",
    "id": "E09000003",
    "legalName": "London Borough of Barnet",
    "uri": "http://www.barnet.gov.uk/"
  },
  "name": "London Borough of Barnet",
  "address": {
    "streetAddress": "4, North London Business Park, Oakleigh Rd S",
    "locality": "London",
    "region": "London",
    "postalCode": "N11 1NP",
    "countryName": "United Kingdom"
  },
  "contactPoint": {
    "name": "Procurement Team",
    "email": "procurement-team@example.com",
    "telephone": "01234 345 346",
    "faxNumber": "01234 345 345",
    "url": "http://example.com/contact/"
  }
},
"documents": [{
  "id": "0005",
  "documentType": "notice",
  "title": "Tender Notice",
  "description": "Official tender notice."
},
"language": "en"
},
"awards": [{
  "id": 0,
  "title": "string"
}]
"contracts": [{
  "id": 0,
  "title": "string"
}]
"language": "en",
"relatedProcesses": [
3.1.5.3 /contracting-process/:id/award

This returns the list of awards in a particular contracting process. This call will receive three parameters to define the contracting process. These parameters are:

- id: string
- before: string (ISO 8601)
- after: string (ISO 8601)

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier for this award. It must be unique and cannot change within the Open Contracting Process it is part of (defined by a single ocid).</td>
<td>[string]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>Award title</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Award description</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>datetime</td>
<td>The date this information was first released or published. Composed by two fields:</td>
<td>String</td>
<td>Date-time</td>
<td>True</td>
</tr>
<tr>
<td></td>
<td>• Date: string (ISO 8601)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Time: Integer</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

http://tbfy.library.linkeddata.es/kg-api/contracting-process/ocds-213czf-000-00001/award?before=2010-05-31T10:30:00Z&after=2010-05-01T10:30:00Z

[{
   "id": "ocds-213czf-000-00001-award-0",
   "title": "Award of contract to build new cycle lanes in the center of town.",
   "description": "AnyCorp Ltd has been awarded the contract to build new cycle lanes in the center of town",
   "datetime": {
      "date": "2010-05-10T09:30:00Z",
      "time": 0
   }
}]


3.1.5.4 /contracting-process/:id/contract

This returns the list of contracts in a particular contracting process. This call will receive three parameters to define the contracting process. These parameters are:

- id: string
- before: string (ISO 8601)
- after: string (ISO 8601)

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier for this contract. It must be unique and cannot change within its Open Contracting Process (defined by a single ocid). See the identifier guidance for further details.</td>
<td>[string, integer]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>Contract title.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Contract description.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:


```json
[
  {
    "id": "ocds-213czf-000-00001-contract-01",
    "title": "Contract to build new cycle lanes in the center of town.",
    "description": "A contract has been signed between the Council and AnyCorp Ltd for construction of new cycle lanes in the center of town."
  },
  {
    "id": "ocds-213czf-000-00001-contract-02",
    "title": "Contract to build new cycle lanes in the center of town.",
    "description": "Physical progress report for cycle path construction."
  }
]
```

3.1.5.5 /organisation/

This returns the list of organisations. No parameters are defined in this call. This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Organization identifiers should be drawn from an existing organization identifier list. The scheme field is used to indicate the list or register from which the identifier is drawn. This value should be drawn from the Organization Identifier Scheme codelist.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>legalname</td>
<td>The legally registered name of the organization.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**uri**

A URI to identify the organization, such as those provided by Open Corporates or some other relevant URI provider. This is not for listing the website of the organization: that can be done through the URL field of the Organization contact point.

[string, null]  

**Example:**

http://tbfy.library.linkeddata.es/kg-api/organisation/

```json
[
  {
    "id": "17087984",
    "legalname": "Vermogensconsultancy Borst B.V.",
    "uri": "https://www.kvk.nl/zoeken/handelsregister/%E1%80%A0uitgebreid-zoeken&handelsnaam=%E1%80%A0kvknummer=17087984&straat=%E1%80%A0postcode=%E1%80%A0huisnummer=%E1%80%A0plaats=%E1%80%A0hoofdvestiging=true&rechtspersoon=true&nevenvestiging=false&zoekvervallen=1&zoekuitgeschreven=1&start=0&initial=0&searchfield=uitgebreidzoeken"
  },
  {
    "id": "17087985",
    "legalname": "Bover B.V.",
    "uri": "https://www.kvk.nl/zoeken/handelsregister/%E1%80%A0uitgebreid-zoeken&handelsnaam=%E1%80%A0kvknummer=17087985&straat=%E1%80%A0postcode=%E1%80%A0huisnummer=%E1%80%A0plaats=%E1%80%A0hoofdvestiging=true&rechtspersoon=true&nevenvestiging=false&zoekvervallen=1&zoekuitgeschreven=1&start=0&initial=0&searchfield=uitgebreidzoeken"
  },
  {
    "id": "17087986",
    "legalname": "Smolders Holding B.V.",
    "uri": "https://server.db.kvk.nl/TST-BIN/BU/TSWS001@?BUTT=17087986"
  }
]
```

### 3.1.5.6 /organisation/:id

Returns all the information about a single organisation, the core data for the given organisation. This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>A common name for this organization or other participant in the contracting process. The identifier object provides a space for the formal legal name, and so this may either repeat that value, or could provide the common name by which this organization or entity is known. This field may also include details of the department or sub-unit involved in this contracting process.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>The ID used for cross-referencing to this party from other sections of the release. This field may be built with the following structure {identifier.scheme}-{identifier.id}(-{department-identifier}).</td>
<td>[string]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>identifier</td>
<td>The primary identifier for this organization or participant. Identifiers that uniquely pick out a legal entity should be preferred. Consult the organization identifier guidance for the</td>
<td>object</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### additionalIdentifiers
A list of additional / supplemental identifiers for the organization or participant, using the organization identifier guidance. This could be used to provide an internally used identifier for this organization in addition to the primary legal entity identifier (see identifier).

<table>
<thead>
<tr>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>array[Identifier]</td>
</tr>
</tbody>
</table>

### address
An address. This may be the legally registered address of the organisation or may be a correspondence address for this particular contracting process (see Address).

<table>
<thead>
<tr>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
</tr>
</tbody>
</table>

### contactPoint
Contact details that can be used for this party (see ContactPoint).

<table>
<thead>
<tr>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
</tr>
</tbody>
</table>

### roles
The party’s role(s) in the contracting process. Role(s) should be taken from the partyRole codelist. Values from the provided codelist should be used wherever possible, though extended values can be provided if the codelist does not have a relevant code.

<table>
<thead>
<tr>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>[array, null]</td>
</tr>
</tbody>
</table>

### details
Additional classification information about parties can be provided using partyDetail extensions that define particular properties and classification schemes.

<table>
<thead>
<tr>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>[object, null]</td>
</tr>
</tbody>
</table>

**Example:**

```
http://tbfy.library.linkeddata.es/kg-api/organisation/17087984
```

```json
{
  "id": "17087984",
  "name": "London Borough of Barnet",
  "identifier": {
    "id": "E09000003",
    "legalName": "London Borough of Barnet",
    "scheme": "GB-LAC",
    "uri": "http://www.barnet.gov.uk/"}
},
  "address": {
    "countryName": "United Kingdom",
    "locality": "London",
    "postalCode": "N11 1NP",
    "region": "London",
    "streetAddress": "4, North London Business Park, Oakleigh Rd S"
  },
  "contactPoint": {
    "email": "procurement-team@example.com",
    "faxNumber": "01234 345 345",
    "name": "Procurement Team",
    "telephone": "01234 345 346",
    "url": "http://example.com/contact/"
  },
  "roles": [
```
3.1.5.7 /organisation/:id/contracting-process

This returns the list of contracting process in a particular organisation. This call will receive one parameter to define the organisation. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>ocid</td>
<td>A globally unique identifier for this Open Contracting Process. Composed of a publisher prefix and an identifier for the contracting process</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>id</td>
<td>An identifier for this particular release of information. A release identifier must be unique within the scope of its related contracting process (defined by a common ocid), and unique within any release package it appears in. A release identifier must not contain the # character.</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>name</td>
<td>A common name for this organization or other participant in the contracting process. The identifier object provides a space for the formal legal name, and so this may either repeat that value, or could provide the common name by which this organization or entity is known. This field may also include details of the department or sub-unit involved in this contracting process.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>datetime</td>
<td>The date this information was first released or published. Composed by two fields:</td>
<td>String</td>
<td>Date-time</td>
<td>True</td>
</tr>
<tr>
<td></td>
<td>• Date: string (ISO 8601)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Time: Integer</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

http://tbfy.library.linkeddata.es/kg-api/organisation/17087984/contracting-process

```json
[
  {
    "ocid": "ocds-213c7f-000-00001",
    "id": "GB-COH-11111111",
    "name": "AnyCorp Ltd",
    "datetime": {
      "date": "2010-05-10T10:30:00Z",
      "time": 0
    }
  },
  {
    "ocid": "ocds-213c7f-000-00001",
    "id": "GB-LAC-E00000003",
    "name": "London Borough of Barnet",
  }
]```
3.1.5.8 /organisation/:id/awards

This returns the list of awards in a particular organisation. This call will receive one parameter to define the organisation. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier for this award. It must be unique and cannot change within the Open Contracting Process it is part of (defined by a single ocid).</td>
<td>[string]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>Award title</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Award description</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| datetime   | The date this information was first released or published. Composed by two fields:  
  - Date: string (ISO 8601)  
  - Time: Integer              | String     | Date-time      | True     |

Example:

`http://tbfy.library.linkeddata.es/kg-api/organisation/17087984/award`

```json
{
  "id": "ocds-213czf-000-00001-award-01",
  "title": "Award of contract to build new cycle lanes in the center of town.",
  "description": "AnyCorp Ltd has been awarded the contract to build new cycle lanes in the center of town.",
  "datetime": {
    "date": "2010-05-10T10:30:00Z",
    "time": 0
  }
}
```
3.1.5.9  /contract/

This returns the list of contracts. To limit the results of the list, this call will receive three parameters to define the date. These parameters are:

- Status: query about the status of the contract (pending, active, cancelled, terminated)
- Before: string (ISO 8601)
- After: string (ISO 8601)

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier for this contract. It must be unique and cannot change within its Open Contracting Process (defined by a single ocid).</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>title</td>
<td>Contract title.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Contract description.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```
http://tbfy.library.linkeddata.es/kg-api/contract?status=active&before=2010-05-31T10:30:00Z&after=2010-05-01T10:30:00Z
```

```
[
  {
    "id": "ocds-213czf-000-00001-contract-01",
    "title": "Contract to build new cycle lanes in the center of town.",
    "description": "A contract has been signed between the Council and AnyCorp Ltd for construction of new cycle lanes in the center of town."
  }
]
```
3.1.5.10 /contract/:id

Returns all the information about a single contract, the core data for the given contract. This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier for this contract. It must be unique and cannot change within its Open Contracting Process (defined by a single ocid).</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>awardID</td>
<td>The award.id against which this contract is being issued.</td>
<td>[string, integer]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>Contract title.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Contract description.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>The current status of the contract. Drawn from the contractStatus codelist (pending, active, cancelled, terminated).</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>period</td>
<td>The start and end date for the contract (see Period).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>value</td>
<td>The total value of this contract. A negative value indicates that the contract will involve payments from the supplier to the buyer (commonly used in concession contracts) (see Value).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>items</td>
<td>The goods, services, and any intangible outcomes in this contract. Note: If the items are the same as the award do not repeat (see Item).</td>
<td>array[Item]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| dateSigned | The date the contract was signed. In the case of multiple signatures, the date of the last signature. Composed by two fields:  
- Date: string (ISO 8601)  
- Time: Integer | [string, null] | date-time |          |
| documents | All documents and attachments related to the contract, including any notices (see Document). | array[Document] |           |          |
| implementation | Information related to the implementation of the contract in accordance with the obligations laid out therein (see Implementation). | object |           |          |
| relatedProcesses | If this process is followed by one or more contracting processes, represented under a separate open contracting identifier (ocid) then details of the related process can be provided here. This is commonly used to point to subcontracts, or to renewal and replacement | array[Related Process] |           |          |
D5.3 - Procurement API's and platform release v2

<table>
<thead>
<tr>
<th>milestone</th>
<th>A list of milestones associated with the finalization of this contract (see Milestone).</th>
<th>array[Milestone]</th>
</tr>
</thead>
<tbody>
<tr>
<td>amendment</td>
<td>A contract amendment is a formal change to, or extension of, a contract, and generally involves the publication of a new contract notice/release, or some other documents detailing the change. The rationale and a description of the changes made can be provided here (see Amendment).</td>
<td>array[Amendment]</td>
</tr>
</tbody>
</table>

Example:

http://tbfy.library.linkeddata.es/kg-api/contract/ocds-213czf-000-00001-contract-01

```json
{
  "id": "ocds-213czf-000-00001-contract-01",
  "title": "Contract to build new cycle lanes in the center of town.",
  "award": {
    "id": "ocds-213czf-000-00001-award-01",
    "uri": ""
  },
  "description": "A contract has been signed between the Council and AnyCorp Ltd for construction of new cycle lanes in the center of town.",
  "status": "active",
  "period": {
    "start": {
      "date": "2010-07-01T00:00:00Z",
      "time": 0
    },
    "end": {
      "date": "2011-08-01T23:59:00Z",
      "time": 0
    }
  },
  "value": {
    "amount": 11000000,
    "currency": "GBP"
  },
  "items": [
    {
      "id": "0001",
      "description": "string"
    }
  ],
  "dateSigned": {
    "date": "2015-06-10T14:23:12Z",
    "time": 0
  },
  "documents": [
    {
      "id": "0008",
      "title": "Signed Contract"
    }
  ],
  "milestones": [
    
```
3.1.5.11 /contract/:id/amendment

This returns the list of amendments in a particular contract. This call will receive three parameters to define the contracting process. These parameters are:

- id: string
- before: string (ISO 8601)
- after: string (ISO 8601)

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(see Amendment).</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```
```

[
  {
    "id": "1.0",
    "datetime": {
      "date": "2010-03-20T09:45:00Z",
      "time": 0
    },
    "rationale": "The tender period has been extended in accordance with council guidelines",
    "description": "The closing date for tender submissions has been extended to 2017-04-08",
    "amendsReleaseID": "ocds-213czf-000-00001-02-tender",
    "releaseID": "ocds-213czf-000-00001-03-tenderAmendment",
    "changes": [
      {
        "name": ""
      }
    ]
  },
  {
    "id": "update-1",
    "datetime": {
      "date": "2010-03-20T09:45:00Z",
      "time": 0
    },
    "rationale": "Update following enquiries."
  }
]
"description": "Following the enquiry period, enquiries were received and responses to questions asked have been published. No changes to the overall tender details were made.",
  amendsReleaseID": "ocds-213czf-000-00001-02-tender",
  "releaseID": "ocds-213czf-000-00001-03-tenderAmendment",
  "changes": [
    
  ]
}
]

3.1.5.12 /contract/:id/document

This returns the list of documents in a particular contract. This call will receive one parameter to define the contracting process. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(see Document).</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```plaintext
http://tbfy.library.linkeddata.es/kg-api/contract/ocds-213czf-000-00001-contract-01/document
```

```json
[  
  {  
    "id": "0007",
    "documentType": "notice",
    "title": "Award notice",
    "description": "Award of contract to build new cycle lanes to AnyCorp Ltd.",
    "datepublished": {  
      "date": "2010-05-10T10:30:00Z",
      "time": 0,
      "format": "text/html",
      "language": "en"
    }
  }
]
```

3.1.5.13 /contract/:id/item

This returns the list of items in a particular contract. This call will receive one parameter to define the contracting process. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(see Item).</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example:

http://tbfy.library.linkeddata.es/kg-api/contract/ocds-213czf-000-00001-contract-01/item

```json
[
  {
    "id": "0001",
    "description": "string",
    "classification": {
      "scheme": "CPV",
      "id": "45233130",
      "description": "Construction work for highways",
      "uri": "http://cpv.data.ac.uk/code-45233130"
    },
    "quantity": 8,
    "unit": {
      "name": "Miles",
      "value": {
        "amount": 137000,
        "currency": "GBP"
      }
    }
  }
]
```

3.1.5.14 /award/

This returns the list of awards. To limit the results of the list, this call will receive two parameters to define the date. These parameters are:

- Before: string (ISO 8601)
- After: string (ISO 8601)

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier for this award. It must be unique and cannot change within the Open Contracting Process it is part of (defined by a single ocid).</td>
<td>[string]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>Award title</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Award description</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>datetime</td>
<td>The date this information was first released or published. Composed by two fields: Date: string (ISO 8601) Time: Integer</td>
<td>String</td>
<td>Date-time</td>
<td>True</td>
</tr>
</tbody>
</table>
Example:

```
http://tbfy.library.linkeddata.es/kg-api/award?before=2010-05-31T10:30:00Z&after=2010-05-01T10:30:00Z
```

```
[
  {
    "id": "ocds-213czf-000-00001-award-01",
    "title": "Award of contract to build new cycle lanes in the center of town.",
    "description": "AnyCorp Ltd has been awarded the contract to build new cycle lanes in the center of town.",
    "datetime": {
      "date": "2010-05-10T10:30:00Z",
      "time": 0
    }
  },
  {
    "id": "ocds-213czf-000-00001-award-02",
    "title": "Award of contract to build new cycle lanes in the center of town.",
    "description": "AnyCorp Ltd has been awarded the contract to build new cycle lanes in the center of town.",
    "datetime": {
      "date": "2010-05-10T10:30:00Z",
      "time": 0
    }
  }
]
```

### 3.1.5.15 /award/:id/

Returns all the information about a single award, the core data for the given award.

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier for this award. It must be unique and cannot change within the Open Contracting Process it is part of (defined by a single ocid).</td>
<td>[string, integer]</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>title</td>
<td>Award title.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Award description.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>The current status of the award drawn from the awardStatus codelist.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>date</td>
<td>The date of the contract award. This is usually the date on which a decision to award was made. Composed by two fields: Date: string (ISO 8601) Time: Integer</td>
<td>[string, null]</td>
<td>date-time</td>
<td></td>
</tr>
<tr>
<td>value</td>
<td>The total value of this award. In the case of a framework contract this may be the total estimated lifetime value, or maximum value, of the agreement. There may be more than one award per procurement. A negative value indicates that the award may involve payments</td>
<td>object</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
from the supplier to the buyer (commonly used in concession contracts) (see Value).

**suppliers**  
The suppliers awarded this award. If different suppliers have been awarded different items or values, these should be split into separate award blocks (see OrganizationReference).

**items**  
The goods and services awarded in this award, broken into line items wherever possible. Items should not be duplicated, but the quantity specified instead (see Item).

**contractPeriod**  
The period for which the contract has been awarded (see Period).

**documents**  
All documents and attachments related to the award, including any notices (see Document).

**amendments**  
An award amendment is a formal change to the details of the award, and generally involves the publication of a new award notice/release. The rationale and a description of the changes made can be provided here (see Amendment).

Example:

```
hhttp://tbfy.library.linkeddata.es/kg-api/award/ocds-213czf-000-00001-award-01
```

```
[  
  {  
    "id": "ocds-213czf-000-00001-award-01",
    "title": "Award of contract to build new cycle lanes in the center of town.",
    "description": "AnyCorp Ltd has been awarded the contract to build new cycle lanes in the center of town.",
    "status": "active",
    "datetime": {  
      "date": "2010-05-10T10:30:00Z",
      "time": 0
    },
    "value": {  
      "amount": 11000000,
      "currency": "GBP"
    },
    "suppliers": [  
      {  
        "suppliers": "AnyCorp Cycle Provision"
      }
    ],
    "items": {  
      "id": "1.0",
      "description": "Cycle lane improvements",
      "classification": {  
        "scheme": "CPV",
        "id": "45233130.0",
        "description": "Construction work for highways",
        "uri": "http://cpv.data.ac.uk/code-45233130"
      },
      "quantity": 8,
      "unit": "SMI"
    }
  }
]```
3.1.5.16 /award/:id/amendment

This returns the list of amendments in a particular award. This call will receive one parameter to define the award. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(see Amendment).</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```http
http://tbfy.library.linkeddata.es/kg-api/award/ocds-213czf-000-00001-award-01/amendment
```

```json
[

  {
    "id": "1.0",
    "datetime": {
      "date": "2010-03-20T09:45:00Z",
      "time": 0
    },
    "rationale": "The tender period has been extended in accordance with council guidelines",
    "description": "The closing date for tender submissions has been extended to 2017-04-08",
  }
]
```
3.1.5.17 /award/:id/item

This returns the list of items in a particular award. This call will receive one parameter to define the award. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(see Item).</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

http://tbfy.library.linkeddata.es/kg-api/award/ocds-213czf-000-00001-award-01/item/

```json
[
  {
    "id": "0001",
    "description": "string",
    "classification": {
      "scheme": "CPV",
      "id": "45233130",
      "description": "Construction work for highways",
      "uri": "http://cpv.data.ac.uk/code-45233130"
    },
    "quantity": 6,
    "unit": "SMI"
  },
  {
    "id": "1.0",
    "description": "Cycle lane improvements",
    "classification": {
      "scheme": "CPV",
      "id": "45233130",
      "description": "Construction work for highways",
      "uri": "http://cpv.data.ac.uk/code-45233130"
    },
    "quantity": 6,
    "unit": "SMI"
  }
]```
3.1.5.18 /award/:id/supplier/

This returns the list of suppliers in a particular award. This call will receive one parameter to define the award. This parameter is:
- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>The identifier of the supplier in the selected scheme.</td>
<td>[string,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>legalname</td>
<td>The legally registered name of the supplier.</td>
<td>[string,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uri</td>
<td>A URI to identify the organization, such as those provided by Open Corporates or some other relevant URI provider. This is not for listing the website of the organization: that can be done through the URL field of the Organization contact point.</td>
<td>[string,</td>
<td>uri</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```json
[{
    "id": "GB-COH-11111111",
    "legalname": "AnyCorp Cycle Provision",
    "uri": ""
}]
```

3.1.5.19 /award/:id/document

This returns the list of documents in a particular award. This call will receive one parameter to define the contracting process. This parameter is:
- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>string</td>
</tr>
<tr>
<td>uri</td>
<td>string</td>
</tr>
</tbody>
</table>
Example:

http://tbfy.library.linkeddata.es/kg-api/award/ocds-213czf-000-00001-award-01/document/

{
  "id": "7.0",
  "documenttype": "awardNotice",
  "title": "Award notice",
  "description": "Award of contract to build new cycle lanes in the center of town to AnyCorp Ltd.",
  "datepublished": {
    "date": "2010-05-10T10:30:00Z",
    "time": 0
  },
  "datemodified": {
    "date": "string",
    "time": 0
  },
  "format": "text/html",
  "language": "en"
}

3.1.5.20 /tender/

This returns the list of tenders. To limit the results of the list, this call will receive one parameter according to the state of the tender. This parameter is:

- Status: query about the status of the contract (pending, active, cancelled, terminated)

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>An identifier for this tender process. This may be the same as the ocid or may be drawn from an internally held identifier for this tender.</td>
<td>[string, integer]</td>
<td>True</td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>A title for this tender. This will often be used by applications as a headline to attract interest, and to help analysts understand the nature of this procurement.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>A summary description of the tender. This should complement structured information provided using the items array. Descriptions should be short and easy to read. Avoid using ALL CAPS.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example:

http://tbfy.library.linkeddata.es/kg-api/tender?status=active

```
[
  {
    "id": "ocds-213czf-000-00001-01-tender",
    "title": "Planned cycle lane improvements",
    "description": "Tenders solicited for work to build new cycle lanes in the center of town."
  },
  {
    "id": "ocds-213czf-000-00001-02-tender",
    "title": "New closing date",
    "description": "The closing date for tender submissions has been extended to 2017-04-08"
  }
]
```

3.1.5.21 /tender/:id/

Returns all the information about a single tender, the core data for the given tender.

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>An identifier for this tender process. This may be the same as the ocid or may be drawn from an internally held identifier for this tender.</td>
<td>[string, integer]</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>title</td>
<td>A title for this tender. This will often be used by applications as a headline to attract interest, and to help analysts understand the nature of this procurement.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>A summary description of the tender. This should complement structured information provided using the items array. Descriptions should be short and easy to read. Avoid using ALL CAPS.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>The current status of the award drawn from the awardStatus codelist.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>procuringEntity</td>
<td>The entity managing the procurement. This may be different from the buyer who pays for, or uses, the items being procured (see OrganizationReference).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>items</td>
<td>The goods and services to be purchased, broken into line items wherever possible. Items should not be duplicated, but a quantity of 2 specified instead (see Item).</td>
<td>array[Item]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>value</td>
<td>The total upper estimated value of the procurement. A negative value indicates that the contracting process may involve payments from the supplier to the buyer (commonly used in concession contracts) (see Value).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>minValue</strong></td>
<td>The minimum estimated value of the procurement. A negative value indicates that the contracting process may involve payments from the supplier to the buyer (commonly used in concession contracts) (see Value).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>procurementMethod</strong></td>
<td>Specify tendering method using the method codelist. This is a closed codelist. Local method types should be mapped to this list.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>procurementMethodDetails</strong></td>
<td>Additional detail on the procurement method used. This field may be used to provide the local name of the particular procurement method used.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>procurementMethodRationale</strong></td>
<td>Rationale for the chosen procurement method. This is especially important to provide a justification in the case of limited tenders or direct awards.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>mainProcurementCategory</strong></td>
<td>The primary category describing the main object of this contracting process from the procurementCategory codelist. This is a closed codelist. Local classifications should be mapped to this list.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>additionalProcurementCategories</strong></td>
<td>Any additional categories which describe the objects of this contracting process, from the extendedProcurementCategory codelist. This is an open codelist. Local categories can be included in this list.</td>
<td>[array, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>awardCriteria</strong></td>
<td>Specify the award criteria for the procurement, using the award criteria codelist</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>awardCriteriaDetails</strong></td>
<td>Any detailed or further information on the award or selection criteria.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>submissionMethod</strong></td>
<td>Specify the method by which bids must be submitted, in person, written, or electronic auction. Using the submission method codelist.</td>
<td>[array, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>submissionMethodDetails</strong></td>
<td>Any detailed or further information on the submission method. This may include the address, e-mail address or online service to which bids should be submitted, and any special requirements to be followed for submissions.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>tenderPeriod</strong></td>
<td>The period when the tender is open for submissions. The end date is the closing date for tender submissions (see Period).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>enquiryPeriod</strong></td>
<td>The period during which potential bidders may submit questions and requests for clarification to the entity managing procurement. Details of how to submit enquiries should be provided in attached notices, or in</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td>Type</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>submissionMethodDetails</td>
<td>Structured dates for when responses to questions will be made can be provided using tender milestones (see Period).</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>hasEnquiries</td>
<td>A true/false field to indicate whether any enquiries were received during the tender process. Structured information on enquiries that were received, and responses to them, can be provided using the enquiries extension.</td>
<td>boolean, null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eligibilityCriteria</td>
<td>A description of any eligibility criteria for potential suppliers.</td>
<td>string, null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>awardPeriod</td>
<td>The period for decision making regarding the contract award. The end date should be the date on which an award decision is due to be finalized. The start date is optional and may be used to indicate the start of an evaluation period (see Period).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>contractPeriod</td>
<td>The period over which the contract is estimated or required to be active. If the tender does not specify explicit dates, the duration field may be used (see Period).</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>numberOfTenderers</td>
<td>The number of parties who submit a bid.</td>
<td>integer, null</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tenderers</td>
<td>All parties who submit a bid on a tender. More detailed information on bids and the bidding organization can be provided using the optional bid extension (see OrganizationReference).</td>
<td>array[OrganizationReference]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>documents</td>
<td>All documents and attachments related to the tender, including any notices. See the documentType codelist for details of potential documents to include. Common documents include official legal notices of tender, technical specifications, evaluation criteria, and, as a tender process progresses, clarifications and replies to queries (see Document).</td>
<td>array[Document]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>milestones</td>
<td>A list of milestones associated with the tender (see Milestone).</td>
<td>array[Milestone]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>amendments</td>
<td>A tender amendment is a formal change to the tender, and generally involves the publication of a new tender notice/release. The rationale and a description of the changes made can be provided here (see Amendment).</td>
<td>array[Amendment]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example:

```json
http://tbfy.library.linkeddata.es/kg-api/tender/ocds-213czf-000-00001-01-tender/

[
  {
    "id": "ocds-213czf-000-00001-01-tender",
    "title": "Planned cycle lane improvements",
    "description": "Tenders solicited for work to build new cycle lanes in the center of town.",
    "status": "completed",
    "procuringEntity": {
      "id": "GB-LAC-E0900003",
      "legalname": "London Borough of Barnet",
      "uri": ""
    },
    "items": [
      {
        "id": "1.0",
        "description": "Cycle lane improvements"
      }
    ],
    "value": {
      "amount": 1100000,
      "currency": "GBP"
    },
    "minValue": {
      "amount": 600000,
      "currency": "GBP"
    },
    "procurementMethod": "open",
    "procurementMethodDetails": "In open procedures, any interested economic operator may submit a tender in response to a contract notice.",
    "procurementMethodRationale": "An open competitive tender is required by EU Rules",
    "mainProcurementCategory": "string",
    "additionalProcurementCategories": [
      "goods"
    ],
    "awardCriteria": "bestProposal",
    "awardCriteriaDetails": "The best proposal, subject to value for money requirements, will be accepted.",
    "submissionMethod": [
      "electronicSubmission"
    ],
    "submissionMethodDetails": "Submit through the online portal at http://example.com/submissions/ocds-213czf-000-00001-01/",
    "tenderPeriod": {
      "startDate": {
        "date": "2010-03-01T09:00:00Z",
        "time": 0
      },
      "endDate": {
        "date": "2011-04-01T18:00:00Z",
        "time": 0
      }
    },
    "maxExtendedDate": {
      "date": "string",
      "time": 0
    },
    "durationInDays": 31
  },
  "enquiryPeriod": {
    "startDate": {
      "date": "2010-03-01T09:00:00Z",
      "time": 0
    }
  }
]
```
"time": 0 },
  "endDate": {
    "date": "2010-03-14T17:30:00Z",
    "time": 0
  },
  "maxExtendedDate": {
    "date": "string",
    "time": 0
  },
  "durationInDays": 14
},
  "hasEnquiries": true,
  "eligibilityCriteria": "string",
  "awardPeriod": {
    "startDate": {
      "date": "2010-06-01T00:00:00Z",
      "time": 0
    },
    "endDate": {
      "date": "2011-08-01T23:59:59Z",
      "time": 0
    },
    "maxExtendedDate": {
      "date": "string",
      "time": 0
    },
    "durationInDays": 0
  },
  "contractPeriod": {
    "startDate": {
      "date": "2010-07-01T00:00:00Z",
      "time": 0
    },
    "endDate": {
      "date": "2011-06-30T23:59:59Z",
      "time": 0
    },
    "maxExtendedDate": {
      "date": "",
      "time": 0
    },
    "durationInDays": 365
  },
  "numberOfTenderers": 0,
  "tenderers": [
    {
      "id": "GB-COH-1111111",
      "name": "AnyCorp Cycle Provision",
      "identifier": {
        "scheme": "string",
        "id": "string",
        "legalName": "string",
        "uri": "string"
      },
      "additionalIdentifiers": {
        "scheme": "string",
        "id": "string",
        "name": "string",
        "uri": "string"
      },
      "address": {
        "streetAddress": "string",
        "postalCode": "string",
        "addressName": "string",
        "addressType": "string",
        "geo": {
          "lat": 11.111111111111111,
          "long": -22.222222222222222
        }
      },
      "isJointBidOrCollaboration": false,
      "isMainContractor": false,
      "isSpecialContractor": false,
      "isEconomicOperator": true,
      "isWithdrawn": false,
      "isInterested": true,
      "isInAward": false,
      "isInTender": true,
      "isInProcurementPeriod": true,
      "isInAwardPeriod": false,
      "isInContractPeriod": false
    }
  ]
"locality": "string",
"region": "string",
"postalcode": "string",
"countryname": "string"
},
"contactpoint": {
"name": "string",
"email": "string",
"telephone": "string",
"faxnumber": "string",
"url": "string"
},
"roles": [
{
"partyroles": "string"
}
],
"details": {}
},
{
"id": "GB-COH-22222222",
"name": "Beta Corp Construction",
"identifier": {
"scheme": "string",
"id": "string",
"legalname": "string",
"uri": "string"
},
"additionalIdentifiers": {
"scheme": "string",
"id": "string",
"name": "string",
"uri": "string"
},
"address": {
"streetaddress": "string",
"locality": "string",
"region": "string",
"postalcode": "string",
"countryname": "string"
},
"contactpoint": {
"name": "string",
"email": "string",
"telephone": "string",
"faxnumber": "string",
"url": "string"
},
"roles": [
{
"partyroles": "string"
}
],
"details": {}
},
"documents": [
{
"id": "3.0",
"documenttype": "x_consultationDocument",
"title": "Consultation on cycle provision",
"description": "A consultation document inviting citizen input into cycle provision."
}
"datepublished": {
  "date": "2010-02-15T00:00:00Z",
  "time": 0
},
"datemodified": {
  "date": "string",
  "time": 0
},
"format": "text/html",
"language": "en"
},
{
  "id": "4.0",
  "documenttype": "x_map",
  "title": "Map of affected areas",
  "description": "A consultation document inviting citizen input into cycle provision.",
  "url": "http://example.com/consultations/cycle-provision/",
  "datepublished": {
    "date": "2010-02-15T00:00:00Z",
    "time": 0
  },
  "datemodified": {
    "date": "string",
    "time": 0
  },
  "format": "offline/print",
  "language": "en"
}
],
"milestones": [
  {
    "id": "string",
    "title": "string",
    "type": "string",
    "description": "string",
    "code": "string",
    "dueDate": {
      "date": "string",
      "time": 0
    },
    "dateMet": {
      "date": "string",
      "time": 0
    },
    "dateModified": {
      "date": "string",
      "time": 0
    },
    "status": "pending",
    "documents": [
      {
        "id": "string",
        "documenttype": "string",
        "title": "string",
        "description": "string",
        "url": "string",
        "datepublished": {
          "date": "string",
          "time": 0
        },
        "datemodified": {
          "date": "string",
          "time": 0
        }
      }
    ]
  }
]


3.1.5.22 /tender/:id/item

This returns the list of items in a particular tender. This call will receive one parameter to define the award. This parameter is:

- **id**: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(see Item)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example:

```
http://tbfy.library.linkeddata.es/kg-api/tender/ocds-213czf-000-00001-01-tender/item
```

```
[
  {
    "id": "1.0",
    "description": "Cycle lane improvements",
    "classification": {
      "scheme": "CPV",
      "id": "45233130.0",
      "description": "Construction work for highways",
      "uri": "http://cpv.data.ac.uk/code-45233130"
    },
    "quantity": 8,
    "unit": "SMI"
  }
]
```

### 3.1.5.23 /tender/:id/contracting-process

This returns the list of contracting process in a particular tender. This call will receive one parameter to define the organisation. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>ocid</td>
<td>A globally unique identifier for this Open Contracting Process. Composed of a publisher prefix and an identifier for the contracting process</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>id</td>
<td>An identifier for this particular release of information. A release identifier must be unique within the scope of its related contracting process (defined by a common ocid), and unique within any release package it appears in. A release identifier must not contain the # character.</td>
<td>String</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>name</td>
<td>A common name for this organization or other participant in the contracting process. The identifier object provides a space for the formal legal name, and so this may either repeat that value, or could provide the common name by which this organization or entity is known. This field may also include details of the department or sub-unit involved in this contracting process.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| datetime   | The date this information was first released or published. Composed by two fields:
- Date: string (ISO 8601)
- Time: Integer | String      | Date-time  | True     |
D5.3 - Procurement API's and platform release v2

Example:

http://tbfy.library.linkeddata.es/kg-api/tender/ocds-213czf-000-00001-01-tender/contracting-process

```
[
  {
    "ocid": "ocds-213czf-000-00001",
    "id": "ocds-213czf-000-00001-06-implementation",
    "name": "string",
    "datetime": {
      "date": "2011-01-10T09:30:00Z",
      "time": 0
    }
  }
]
```

3.1.5.24 /tender/:id/document

This returns the list of documents in a particular tender. This call will receive one parameter to define the contracting process. This parameter is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>(see Document)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

http://tbfy.library.linkeddata.es/kg-api/tender/ocds-213czf-000-00001-01-tender/document

```
[
  {
    "id": "1.0",
    "documenttype": "procurementPlan",
    "title": "Area Wide Cycle Improvements - Procurement Plan",
    "description": "The overall strategic framework for procurement to enhance cycle provision.",
    "datepublished": {
      "date": "2009-01-05T00:00:00Z",
      "time": 0
    },
    "datemodified": {
      "date": "string",
      "time": 0
    },
    "format": "application/pdf",
    "language": "en"
  },
  {
    "id": "2.0",
    "documenttype": "procurementPlan",
    "title": "Cycle provision - Needs Assessment",
    "description": "Needs assessment for provision for cyclists in the centerstring of town.",
    "datepublished": {
      "date": "2009-01-15T00:00:00Z",
      "time": 0
    }
  }
]```
The following building block has been commonly re-used throughout the standard in the document:

**Identifier**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>scheme</td>
<td>Organization identifiers should be drawn from an existing organization identifier list. The scheme field is used to indicate the list or register from which the identifier is drawn. This value should be drawn from the Organization Identifier Scheme codelist.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>The identifier of the organization in the selected scheme.</td>
<td>[string, integer, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>legalName</td>
<td>The legally registered name of the organization.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uri</td>
<td>A URI to identify the organization</td>
<td>[string, null]</td>
<td>uri</td>
<td></td>
</tr>
</tbody>
</table>

**Address**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>streetAddress</td>
<td>The street address. For example, 1600 Amphitheatre Pkwy.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>locality</td>
<td>The locality. For example, Mountain View.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>region</td>
<td>The region. For example, CA.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>postalCode</td>
<td>The postal code. For example, 94043.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>countryName</td>
<td>The country name. For example, United States.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### ContactPoint

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>The name of the contact person, department, or contact point, for correspondence relating to this contracting process.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>email</td>
<td>The e-mail address of the contact point/person.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>telephone</td>
<td>The telephone number of the contact point/person. This should include the international dialing code.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>faxNumber</td>
<td>The fax number of the contact point/person. This should include the international dialing code.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>uri</td>
<td>A URI to identify the organization</td>
<td>[string, null]</td>
<td>uri</td>
<td></td>
</tr>
</tbody>
</table>

### Period

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>startDate</td>
<td>The start date for the period. When known, a precise start date must always be provided.</td>
<td>[string, null]</td>
<td>Date-time</td>
<td></td>
</tr>
<tr>
<td>endDate</td>
<td>The end date for the period. When known, a precise end date must always be provided.</td>
<td>[string, null]</td>
<td>Date-time</td>
<td></td>
</tr>
<tr>
<td>maxExtentDate</td>
<td>The period cannot be extended beyond this date. This field is optional and can be used to express the maximum available data for extension or renewal of this period.</td>
<td>[string, null]</td>
<td>Date-time</td>
<td></td>
</tr>
<tr>
<td>durationInDays</td>
<td>The maximum duration of this period in days. A user interface may wish to collect or display this data in months or years as appropriate but should convert it into days when completing this field. This field can be used when exact dates are not known. Where a startDate and endDate are given, this field is optional, and should reflect the difference between those two days. Where a startDate and maxExtentDate are given, this field is optional, and should reflect the difference between startDate and maxExtentDate.</td>
<td>[integer, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Value

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>amount</td>
<td>Amount as a number.</td>
<td>[number,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>currency</td>
<td>The currency for each amount should always be specified using the</td>
<td>[string,</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>uppercase 3-letter currency code from ISO4217.</td>
<td>null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Item

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>A local identifier to reference and merge the items by. Must be unique within a given array of items.</td>
<td>[string, integer]</td>
<td>Date-time</td>
<td>True</td>
</tr>
<tr>
<td>description</td>
<td>A description of the goods, services to be provided.</td>
<td>[string, null]</td>
<td>Date-time</td>
<td></td>
</tr>
<tr>
<td>classification</td>
<td>The primary classification for the item.</td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td>additional</td>
<td>An array of additional classifications for the item.</td>
<td>array[Classification]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>classification</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>quantity</td>
<td>The number of units required.</td>
<td>[number, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>unit</td>
<td>A description of the unit in which the supplies, services or works are</td>
<td>[object, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>provided (e.g. hours, kilograms) and the unit-price. For comparability, an established list of units can be used.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### RelatedProcess

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>A local identifier for this relationship, unique within this array.</td>
<td>[string]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>relationship</td>
<td>Specify the type of relationship using the related process codelist.</td>
<td>[array, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>The title of the related process, where referencing an open contracting process, this field should match the tender/title field in the related process.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>scheme</td>
<td>The identification scheme used by this cross-reference from the related process scheme codelist. When cross-referencing information also published using OCDS, an Open Contracting ID (ocid) should be used.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>A local identifier for this milestone, unique within this block. This field is used to keep track of multiple revisions of a milestone through the compilation from release to record mechanism.</td>
<td>[string, integer]</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>title</td>
<td>Milestone title.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>type</td>
<td>The type of milestone, drawn from an extended milestoneType codelist.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>A description of the milestone.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>code</td>
<td>Milestone codes can be used to track specific events that take place for a particular kind of contracting process. For example, a code of “approvalLetter” could be used to allow applications to understand this milestone represents the date an approvalLetter is due or signed. Milestone codes is an open codelist, and codes should be agreed among data producers and the applications using that data.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dueDate</td>
<td>The date the milestone is due.</td>
<td>[string, null]</td>
<td>date-time</td>
<td></td>
</tr>
<tr>
<td>dateMet</td>
<td>The date on which the milestone was met.</td>
<td>[string, null]</td>
<td>date-time</td>
<td></td>
</tr>
<tr>
<td>dateModified</td>
<td>The date the milestone was last reviewed or modified, and the status was altered or confirmed to still be correct.</td>
<td>[string, null]</td>
<td>date-time</td>
<td></td>
</tr>
<tr>
<td>status</td>
<td>The status that was realized on the date provided in dateModified, drawn from the milestoneStatus codelist.</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Document

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>A local, unique identifier for this document. This field is used to keep track of multiple revisions of a document through the compilation from release to record mechanism.</td>
<td>[string, integer]</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>documentType</td>
<td>A classification of the document described taken from the documentType codelist. Values from the provided codelist should be used wherever possible, though extended values can be provided if the codelist does not have a relevant code.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>title</td>
<td>The document title.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>A short description of the document. We recommend descriptions do not exceed 250 words. In the event the document is not accessible online, the description field can be used to describe arrangements for obtaining a copy of the document.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>url</td>
<td>Direct link to the document or attachment. The server providing access to this document should be configured to correctly report the document mime type.</td>
<td>[string, null]</td>
<td>uri</td>
<td></td>
</tr>
<tr>
<td>datePublished</td>
<td>The date on which the document was first published. This is particularly important for legally important documents such as notices of a tender.</td>
<td>[string, null]</td>
<td>date-time</td>
<td></td>
</tr>
<tr>
<td>dateModified</td>
<td>Date that the document was last modified</td>
<td>[string, null]</td>
<td>date-time</td>
<td></td>
</tr>
<tr>
<td>format</td>
<td>The format of the document taken from the IANA Media Types codelist, with the addition of one extra value for “offline/print”, used when this document entry is being used to describe the offline publication of a document. Use values from the template column. Links to web pages should be tagged “text/html”.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>language</td>
<td>Specifies the language of the linked document using either two-letter ISO639-1, or extended BCP47 language tags. The use of lowercase two-letter codes from ISO639-1 is strongly recommended unless there is a clear user need for distinguishing the language subtype.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Amendment

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>An identifier for this amendment: often the amendment number.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>date</td>
<td>The date of this amendment.</td>
<td>[string, null]</td>
<td>Date-time</td>
<td></td>
</tr>
<tr>
<td>rationale</td>
<td>An explanation for the amendment.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>A free text, or semi-structured, description of the changes made in this amendment.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>amendsReleaseID</td>
<td>Provide the identifier (release.id) of the OCDS release (from this contracting process) that provides the values for this contracting process before the amendment was made.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>releaseID</td>
<td>Provide the identifier (release.id) of the OCDS release (from this contracting process) that provides the values for this contracting process after the amendment was made.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Implementation

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>transactions</td>
<td>A list of the spending transactions made against this contract.</td>
<td>array[Transaction information]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>milestones</td>
<td>As milestones are completed, milestone completions should be documented (see Milestones).</td>
<td>array[Milestones information]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>documents</td>
<td>Documents and reports that are part of the implementation phase e.g. audit and evaluation reports (see Documents).</td>
<td>array[Documents information]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Planning

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>rationale</td>
<td>The rationale for the procurement provided in free text. More detail can be provided in an attached document.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>milestones</td>
<td>A list of milestones associated with the planning stage (see Milestones).</td>
<td>array[Milestones information]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>documents</td>
<td>A list of documents related to the planning process (see Documents).</td>
<td>array[Documents information]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### OrganizationReference

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Used to cross-reference the entry in the parties section that contains full information on this organization or entity.</td>
<td>[string, null]</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>name</td>
<td>A name field that repeats the name given in the parties section, provided for the convenience of users viewing the data, and to support detection of mistakes in cross-referencing.</td>
<td>[string, null]</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
3.2 TheyBuyForYou Search API

In this section of the deliverable we present the first catalogue of the search API requests that we are developing as part of the added value services described in the TBFY architecture. This search API catalogue is based on the work that is being developed within WP3.

More specifically, the Search API allows dealing with:

- Documents.
- Items.

3.2.1 General overview

The objective of this section is to provide a general overview of the main data models used by the search API as well as the resources and requests that are made available.

Among the main data offered by the search API, we can find information about the documents and items involved within contracting-processes. More details are provided in section 3.2.2. As an example, next we show the response for a list of documents, available at http://tbfy.library.linkeddata.es/search-api/documents:

```json
[
  {
    "id": "00001",
    "name": "Declaration concerning subcontractors"
  },
  {
    "id": "00002",
    "name": "Instructions for staff of external companies working at the JRC-GEEL site"
  }
]
```

In order to improve the understanding of the search API, we have developed a full-fledged mock-up using Postman. The current endpoint for this mock-up API is http://tbfy.library.linkeddata.es/search-api.
3.2.2 Data Model

In this section we provide a brief description of the main elements in the data model of the API, which is graphically described in Figure 10.

![Data Model Diagram]

Figure 10. Overview of the TBFY Search API Data Model

3.2.3 Technology

As aforementioned in the case of the core API, we have already developed a mock-up of the search API so as to facilitate consortium partners as well as third parties the understanding of the main elements and types of requests involved in it, and so as to facilitate discussions around the API design. The search API will be based on RESTful API technology which is an application program interface (API) that uses HTTP requests to GET (used to retrieve a resource’s representation), PUT (allows a client to modify a resource state), POST (used to create resources) and DELETE (requests a resource to be deleted) data. In this initial version of the search API only GET and POST requests will be implemented.

REST, which is considered as a standard for building web services, is aligned with the precepts of the web and provides many benefits that provide lot of features such as suitable actions (GET, POST, PUT, DELETE, HEAD, PATCH…), caching, security, redirection and forwarding.

So, building on top of HTTP, REST APIs provide the means to build flexible APIs that can:

- Support backward compatibility
- Evolvable APIs
- Scalable services
- Securable services
- A spectrum of stateless to stateful services
To carry out this task, a typical technology will be used in the development of APIs from Java, SPRING technology so, all the REST services will be built with SPRING\(^4\).

SPRING has become the de-facto standard for building Java based applications. SPRING is a java programming language framework with the aim of standardising the work and management of the problems that allows to develop applications in an easy and quickly way. The Spring Framework is open source.

### 3.2.4 Resources

In this subsection we provide details of the methods implemented in the TBFY search API, related to the previous data model. A complete summary of these methods is provided in Annex I along with the core API methods. All of these methods may use the HTTP verbs GET, PUT, POST and DELETE. The current mock-up only provides support for the verbs GET and POST.

#### 3.2.4.1 /documents

This returns the list of documents according to the chosen parameters. This call will receive seven parameters to define the documents list. These parameters are:

- id: string
- lang: string
- name: string
- offset: integer
- size: integer
- term: string
- type: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Identifier of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the document</td>
<td>string</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```
http://tbfy.library.linkeddata.es/search-api/documents
```

```
[
  {
    "id": "00001",
    "name": "Declaration concerning subcontractors"
  },
  {
    "id": "00002",
    "name": "Instructions for staff of external companies working at the JRC-GEEl site"
  }
]
```

\(^4\) [https://spring.io/](https://spring.io/)
### 3.2.4.2 /documents/:id

This returns the whole information of a given document. This call will receive one parameter to define the document. This parameters is:

- id: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td>Data related to the document not considered in the other fields.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>format</td>
<td>Format of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>id</td>
<td>Identifier of the document.</td>
<td>string</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>label</td>
<td>List of labels of the document</td>
<td>array[string]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>language</td>
<td>Language of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>text</td>
<td>Text of the document</td>
<td>string</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```
http://tbfy.librairy.linkeddata.es/search-api/documents/id1
```

```
[
  {
    id: "00001",
    name: "Declaration concerning subcontractors",
    text: "sample name",
    labels: [ ],
    data: { },
    format: "doc",
    language: en
  }
]
```

### 3.2.4.3 /items

This returns the list of similar documents according to a given text. This call will receive a message in JSON format that is part of the body of the call. This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Identifier of the document to consult.</td>
<td>string</td>
<td></td>
<td>True</td>
</tr>
<tr>
<td>lang</td>
<td>Language of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>score</td>
<td>Numerical value that represents the level of similarity.</td>
<td>integer</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example:

```
http://tbfy.library.linkeddata.es/search-api/items
```

```
[
  {
    "id": "00001",
    "name": "Declaration concerning subcontractors",
    "lang": "en",
    "score": 20.8
  },
  {
    "id": "00002",
    "name": "Instructions for staff of external companies working at the JRC-GEEl site",
    "lang": "en",
    "score": 6.5
  }
]
```

3.2.4.4 /items/:id

This returns similar documents to a given one. This call will receive one parameter to obtain the whole list. This parameters is:

- **id**: string

This call will show the following information:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Type</th>
<th>Format</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Identifier of the document to consult.</td>
<td>string</td>
<td>True</td>
<td></td>
</tr>
<tr>
<td>lang</td>
<td>Language of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>name</td>
<td>Name of the document.</td>
<td>string</td>
<td></td>
<td></td>
</tr>
<tr>
<td>score</td>
<td>Numerical value that represents the level of similarity.</td>
<td>integer</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example:

```
http://tbfy.library.linkeddata.es/search-api/items/id1
```

```
[
  {
    "id": "00001",
    "name": "Declaration concerning subcontractors",
    "lang": "en",
    "score": 20.8
  },
  {
    "id": "00002",
    "name": "Instructions for staff of external companies working at the JRC-GEEl site",
    "lang": "en",
    "score": 6.5
  }
]
```
4 API accounts, authentication and authorisation

The current mock-up does not provide, obviously, any type of authentication or authorisation mechanisms to access data. However, the core API to be implemented will provide basic authentication and authorisation mechanisms that will be based on Spring Security, given the technology stack that has been proposed for the API development.

More specifically, we will give support to OAuth2.
5 Conclusions and future work

As discussed in the introduction, the main objective of this document was to describe the evolution of the API from the first set of basic principles that were exposed in the initial deliverable (D5.2) into the current status of the API design that we have done once that the ontology network for TheyBuyForYou has been developed, and hence the structure of the data to be maintained in the TheyBuyForYou knowledge graph has been established.

We have concentrated our description in the description of the core API, also paying attention to the additional added-value APIs that will be built on top of it (e.g., search, reconciliation, online tools, etc.). In this document a small initial version of the search API has been included and it is expected that it will evolve according to the received feedback from business cases that will start discussing on how they would be implementing their required functionalities on top of this core API.

The services offered by TBFY are built on different levels of information. The first level provides data as documents available from a document-oriented database. Above this level, the information is organised according to the concepts and relationships described in the OCDS ontology to offer them via RDF through a SPARQL endpoint. At the next level, the Core API structures the data following the REST principles and offers them through HTTP queries in multiple formats.

This technology stack supports added value services that extend the original data with new information and provide it through their interfaces. The Search API, for example, expands the exploration offered in the first level of information by establishing cross-lingual similarity between documents. This service uses the document-oriented database to annotate and identify the content-based relationships, and the Core API to enrich the information of each resource. In turn, the search service will partially support a web application aims to analyse, visualise and detect anomalies in public spending data and public procurement data.

As already described in D5.2, following our agile methodology for software development, and our principles of “release early”, as described in deliverable D5.1, it is expected that the architecture will still be evolving over the course of the project, so as to give support to those needs that will be arising from the user stories derived from the needs of the online tools or mostly from the business cases.

While D5.2 focused on providing an overview of the current offering already provided by some of the most prominent data sources that we are using in the context of the project, this deliverable has focused on providing a description of the current work on the TheyBuyForYou API implementation, which is currently ongoing and will be delivered in a phased approach, especially providing access to the RDF-based data that is being generated in the context of workpackages 1 and 2.

All developments will continue to be regularly updated in the GitHub repositories of the GitHub organisation that represents the project: http://github.com/tbfy.
## ANNEX I. TheyBuyForYou Core API cheatsheet for developers

<table>
<thead>
<tr>
<th>HTTP Method</th>
<th>Endpoint Path</th>
<th>Description</th>
<th>API Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/contracting-process/</td>
<td>List of contracting processes</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contracting-process/:id</td>
<td>Core data for the given contracting process</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contracting-process/:id/award</td>
<td>List of awards in a particular contracting process</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contracting-process/:id/contract</td>
<td>List of contracts in a particular contracting process</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/organisation/</td>
<td>List of organisations</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/organisation/:id</td>
<td>Core data for the given organisation</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/organisation/:id/contracting-process</td>
<td>List of contracting process in a particular organisation.</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/organisation/:id/awards</td>
<td>List of awards in a particular organisation</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contract/</td>
<td>List of contracts</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contract/:id</td>
<td>Core data for the given contract</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contract/:id/amendment</td>
<td>List of amendments in a particular contract</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contract/:id/document</td>
<td>List of documents in a particular contract</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/contract/:id/item</td>
<td>List of items in a particular contract</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/award/</td>
<td>List of awards</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/award/:id/</td>
<td>Core data for the given award</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/award/:id/amendment</td>
<td>List of amendments in a particular award</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/award/:id/item</td>
<td>List of items in a particular award</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/award/:id/supplier/</td>
<td>List of suppliers in a particular award</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/award/:id/document</td>
<td>List of documents in a particular award</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/tender/</td>
<td>List of tenders</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/tender/:id/</td>
<td>Core data for the given tender</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/tender/:id/item</td>
<td>List of items in a particular tender</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/tender/:id/contracting-process</td>
<td>List of contracting process in a particular tender.</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/tender/:id/document</td>
<td>List of documents in a particular tender</td>
<td>Core API</td>
</tr>
<tr>
<td>GET</td>
<td>/document</td>
<td>List of documents</td>
<td>Search API</td>
</tr>
<tr>
<td>GET</td>
<td>/document/:id</td>
<td>Core data for the given document</td>
<td>Search API</td>
</tr>
<tr>
<td>POST</td>
<td>/item</td>
<td>List of documents</td>
<td>Search API</td>
</tr>
<tr>
<td>GET</td>
<td>/item/:id</td>
<td>List of documents</td>
<td>Search API</td>
</tr>
</tbody>
</table>