Software Requirements Specifications Document

From Standards to Ontologies - A Web-based tool to semantify/ontologize the knowledge of a standards with semantic technologies

Team:
Aleksandr Korovin
Shinho Kang
Alexey Karpov
Omar Gutiérrez

Supervisor
Irlán Grangel

Project overview

Description

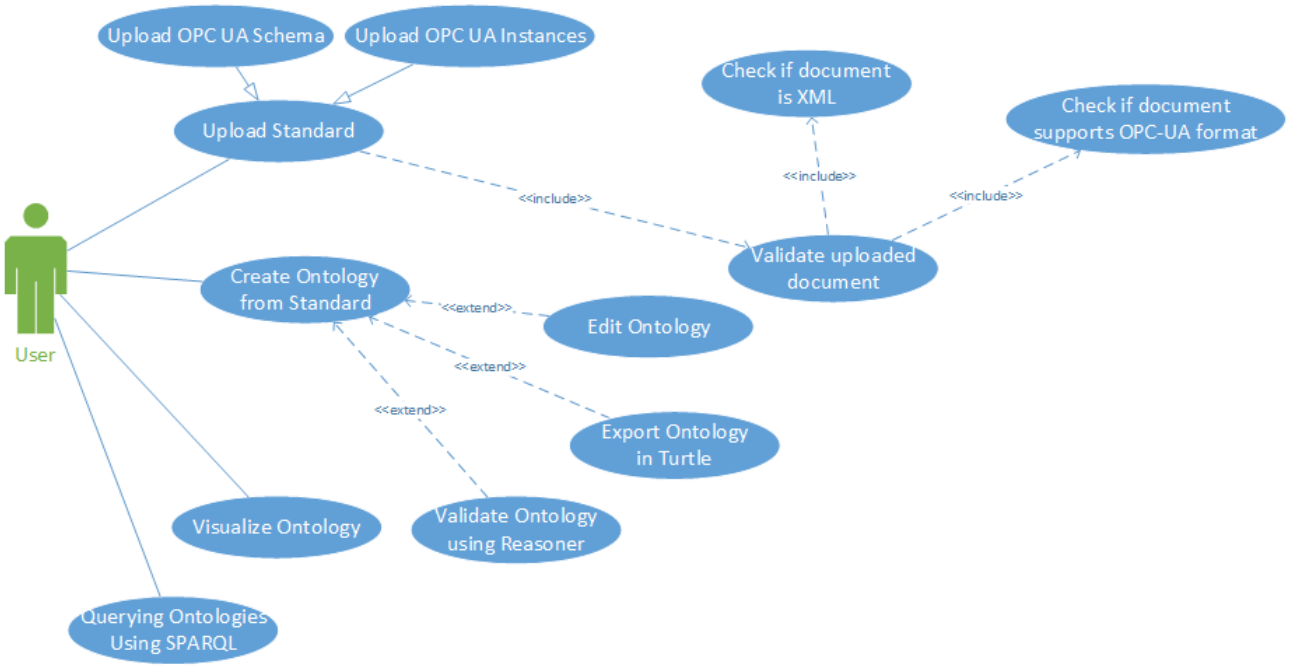
A web tool will be developed to read and analyze OPC UA standard specification documents, these will be imported and the system will provide the functionality of parsing to create a vocabulary allowing the user the edition of this vocabulary.

The development is intended to be done using web technologies such as Node.js, Express, React.js, etc.

Software requirements

Requirements	Description	Priority
Functional		
Translation of OPC UA Schema files into OWL vocabularies/ontologies	Load and read OPC UA Schema files and translate it into OWL vocabularies/ontologies. The load of the file could be done specifying the URL of the file or directly uploading the file.	Very high
Visualization of ontologies	The translated file will be visualized graphically in a form of graph/tree	High
Edition of vocabularies/ontologies	Once it has been obtained the ontology the user will be able to modify and update it using a visual editor	High
Management and storage of vocabularies	The user will be able to store and handle the vocabularies, having functionalities such as save, delete, save as, etc.	High
Identify Malformed XML files	The system will be able to detect XML files that not comply with the SOA OPC schema and alert the user about it	Low
SPARQL queries support	Each of the ontologies it may be explored using the SPARQL query language	Very low
Non-functional		
Handle different format files	Manage different format files: • Turtle (.ttl) • RDF (.rdf)	Medium
Handle large files	The system will be able to load large SOA OPC files and efficiently handle and visualize them keeping a constant time performance.	Medium
Responsive interface	The system will be able to be visualized in different devices such as smart-phones, tablets and computers	Low

Use cases


Upload OPC UA Schema
Upload OPC UA Instances
Upload Standard
Check if document is XML
Check if document supports OPC-UA format
<<include>>
<<include>>
<<include>>
Validate uploaded document
User
Create Ontology from Standard
<<extend>>
Edit Ontology
<<extend>>
<<extend>>
Export Ontology in Turtle
Visualize Ontology
Validate Ontology using Reasoner
Querying Ontologies Using SPARQL

Use case templates

Title	**Create Ontology**
ID	UC1
Description	Self-explained
Precondition	The user is logged into the system
Postcondition	A personalized ontology is obtained
Main course	1. The system shows an interface with the graph of the ontology. 2. The user modify the ontology with help of an editor. 3. The user press the button “Save” to store the ontology. 4. The system validates the ontology. 5. The ontology is saved.
Alternate course (a)	1. The system shows an interface with the graph of the ontology. 2. The user modify the ontology with help of an editor. 3. The user press the button “Save” to store the ontology. 4. The system validates the ontology. 5. The system shows the next error message to the user: “The ontology is not valid”.
Alternate course (b)	1. The system shows an interface with the graph of the ontology. 2. The user press the button “Export” to store the ontology. 3. The system exports the ontology to Turtle

Title	**Upload Standard**
ID	UC2
Description	Self-explained
Precondition	The user is logged into the system
Postcondition	A graph visualizing the ontology is obtained
Main course	1. User selects the option “Upload OPC UA Schema” or “Upload OPC UA Instance” from the interface. 2. The system validates the uploaded document and create an ontology from the Standard file 3. The system shows an interface with the graph of the correspondent ontology
Alternate course	1. User selects the option “Upload OPC UA Schema” or “Upload OPC UA Instance” from the interface. 2. The system validates the uploaded file and shows an error message to the user stating that the file is not a valid schema file.

Title	**Query Ontology**
ID	UC3
Description	Self-explained
Precondition	The user is logged into the system
Postcondition	The system returns the result in form of queries
Main course	1. The system shows an interface with a text box giving the user the possibility to introduce SPARQL queries 2. The user writes the SPARQL query and press the button “Execute” 3. The system executes the query 4. The system returns the result in form of triples
Alternate course	1. The system shows an interface with a text box giving the user the possibility to introduce SPARQL queries 2. The user writes the SPARQL query and press the button “Execute” 3. The system executes the query 4. The system shows the next error “the query is not valid”