

Project

ONTO-APP-MODELS

USER DOCUMENTATION

Starting the application in the application generator

The application generator has the frontend and the backend part. It functions as a client server application. The end user opens the frontend part and has a dashboard of the OntoUI application generator. The user has an option to upload its own Form Application Model (FAM) or to choose a model from the list of models which are already stored on the server.

OntoUI

List server models

Durchsuchen... ...

Upload model to server

Run server model

Export output file

UI model is not loaded

Model instance:
{}

Output graph instance:
Empty

The selected model is loaded on the server when the user clicks “Load model” for the corresponding problem.

OntoUI

List server models

Durchsuchen... ...

Upload model to server

Run server model

Export output file

List of application models on the server

Filename	Model	Description	Action
restaurant_model_ttl1.ttl	http://example.org/logicinterface/restaurant/restaurant_model	The application model to insert main information about a restaurant.	Load model
enter-string-form.ttl	http://example.org/logicinterface/testing/one_simple_form_model	The form consists of a single input field, submit and cancel button.	Load model
simple_form_shacl_1.ttl	http://example.org/logicinterface/testing/restaurant_model	The application model of a form with two elements using shacl.	Load model

Model instance:
{}

Output graph instance:
Empty

We can choose a basic use case with a single form input field and two buttons “Submit” and “Cancel”.

OntoUI

The following model is loaded.

<input type="button" value="List server models"/>	
<input type="button" value="Durchsuchen..."/>	
<input type="button" value="Upload model to server"/>	File: loadedModelInfo.filename
<input type="button" value="Run server model"/>	Description: loadedModelInfo.description
	Action: <input type="button" value="Run the model"/>
<input type="button" value="Export output file"/>	

Model instance:
{}

Output graph instance:
Empty

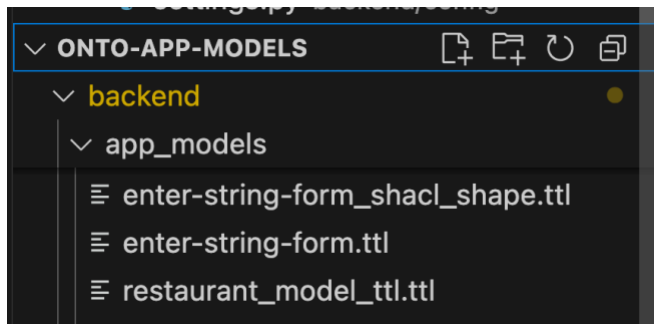
After the model is loaded. There is an option to “Run the model”, which starts the corresponding application. We have chosen the following shacl shape that has a single property which is a v1:legalName of the v1:BusinessEntity in GoodRelations.

```
@prefix v1: <http://purl.org/goodrelations/v1#> .
@prefix shacl: <http://www.w3.org/ns/shacl#> .
@prefix schema: <http://schema.org/> .
@base <http://example.org/logicinterface/testing/> .

### http://example.org/logicinterface/testing/shacl_shape_1
:shacl_shape_1 rdf:type owl:NamedIndividual ,
                shacl:NodeShape ;
                shacl:property _:genid1 ;
                shacl:targetClass v1:BusinessEntity .

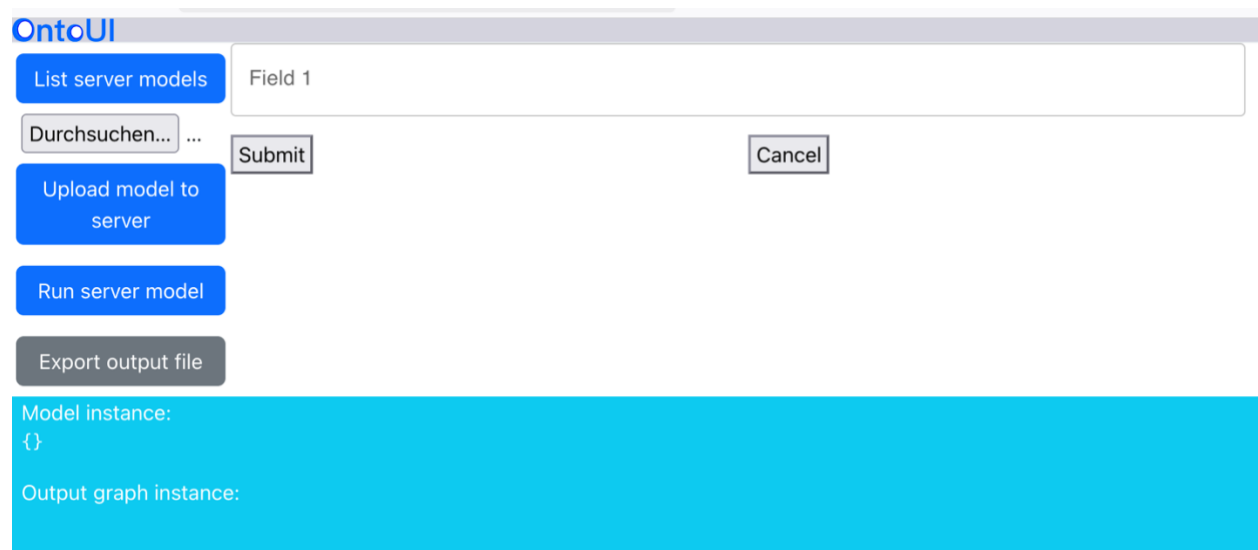
_:genid1 shacl:datatype xsd:string ;
          shacl:path v1:legalName ;
          shacl:maxCount 1 .
```

All models on the server are stored in the directory `app_models`:



There can be seen that we have `enter_string_shape.ttl` and `enter-string-form.ttl` files. They are the SHACL file with shapes and extended application model respectively.

After the user runs the model it is generated user interface and in our example it is a form:



When the user enters a string in the input field and proceeds with the submit, the data is transferred to the server where new triples are added to the in-memory triple store.

The next screen of the application shows in the green area below what triples are inserted in the output knowledge graph. This window offers to the user option to start validation using a pySHACL library in the next step.

OntoUI

List server models

The data have been stored in the database.

Durchsuchen... ...

Validate

Upload model to
server

Run server model

Export output file

Model instance:

```
{}
```

Output graph instance:

```
@prefix ns1: <http://purl.org/goodrelations/v1#> .
```

```
<http://example.org/logicinterface/testing/instance/ea721fa6-7146-4c52-baf2-ce32b363332c> a <http://purl.obolibrary.org/obo/ro:0000000> ;  
    ns1:BusinessEntity ;  
    ns1:legalName "Car Parts GmbH" .
```

After the validation is initiated, we got the message that the created knowledge graph conforms to the SHACL shape above.

OntoUI

List server models

Validation Report Conforms: True

Durchsuchen... ...

OK

Upload model to
server

Run server model

Export output file

Model instance:

```
{}
```

Output graph instance:

```
@prefix ns1: <http://purl.org/goodrelations/v1#> .
```

```
<http://example.org/logicinterface/testing/instance/701cd4a3-686f-47c8-9ac2-7001665e7b18> a ns1:BusinessEntity ;  
    ns1:legalName "Car Parts GmbH" .
```