|  |  |
| --- | --- |
| /nffgs | Retrieve the set of all nffgs or create a set of new nffgs |
| /nffg | Used to create a new nffg |
| /nffg/{id} | Retrieve an nffg given its name |
| /policies | Retrieve the set of all policies or create a set of new policies |
| /policy/{id} | Retrieve the information of a single policy |
| /policy | Used to create a single new policy or to verify a not stored policy |
|  |  |
|  | \*the way this operations are implemented are described below |
|  |  |
|  |  |

CREATE NFFG WITH NAME

|  |  |
| --- | --- |
| POST | /nffg |
| Content-Type | application/xml |
| Accept | application/xml |
| Body | xsd type: Nffg |
| Response | 200 application /xml  type: xsd:XNffg  Return the same entity we send to the service with a modification: the last update time is now set to the server actual time. |
|  | 403 Forbidden – Resource already existing - Nffg already existing  (I don’t used error 409 - Conflict because it is suggested to use it for PUT requests) |
|  | 500 - Internal Server Error – all problems concerning contacting neo4j |
| In case of service not available | 404 - Problems, service not available – e.g. wrong url, WS not started |

GET all available Nffgs

|  |  |
| --- | --- |
| GET | /nffgs |
| Accept | Application/xml |
| Response | 200 application/xml  Body type: xsd:Nffgs |
|  |  |
|  |  |
| In case of service not available | 404 - Problems, service not available – wrong url ,WS not started |
|  | Typically no other errors can occur as no queries are executed, neither to Neo4J service |

Get NFFG by name

|  |  |
| --- | --- |
| GET | /nffg/{name} |
| Content-Type | Application/xml |
|  |  |
| Response | 200 application/xml  Body type: xsd:XNffg |
|  | 404 Not Found – If the nffg with the requested name is not available in the system |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |
|  |  |
|  | Note: an error 404 is automatically sent back if the name of the requested Nffg does not respect the regular expression on the name ( as explained in the intro.pdf of assignment 1) |
|  |  |

DELETE NFFG by name – remove related policies

|  |  |
| --- | --- |
| DELETE | /nffg/{name}?delpolicy=y  Or  /nffg/{name}  Considering as default delpolicy=y |
| Response | 200 application/xml  Returns the information about the deleted Nffg.  Deletes all the policies relaying on this NFFG. |
|  | 404 – The Nffg you want to delete is not available. Impossible to remove it |
|  | 500 – Internal Server Error – Occurs if you try to delete the Nffg from NEO4J and NEO4J is not reachable. |
|  |  |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |

DELETE NFFG by name – Do not remove related policies

|  |  |
| --- | --- |
| DELETE | /nffg/{name}?delpolicy=n |
| Response | 200 application/xml  Returns the information about the deleted Nffg. Possible only if all policies relative to the nffg have already removed. |
|  | 404 – NotFound The Nffg you want to delete is not available. Impossible to remove it |
|  | 403 – Forbidden The policies referring to the nffg are still available. Remove all related policies and try again |
|  | 500 – Internal Server Error – Occurs if you try to delete the Nffg from NEO4J and NEO4J is not reachable. |
|  |  |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |

**Note: if delpolicy assumes values different from y/n an error:**

**405 – Method Not Allowed is returned. Please provide the query parameter: delpolicy=y or delpolicy=n. Alternatively do not specify the parameter delpolicy (default actions y is assumed)**

CREATE a new Policy overwriting a previously existing one

|  |  |
| --- | --- |
| POST | /policy?overwrite=y |
| Content-Type | application/xml |
| Accept | application/xml |
| Body | xsd:XPolicy |
| Response |  |
|  | 200 – application/xml Policy created, returns the information of the created policy |
|  | 404 – NotFound – The Policy refers to a Nffg that is not existing. |
|  | 500 – Internal Server Error – in case NEO4J is not available or retrieve some errors. |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |
|  |  |

CREATE a new Policy without overwriting a previously existing one

|  |  |
| --- | --- |
| POST | /policy?overwrite=n |
| Content-Type | application/xml |
| Accept | application/xml |
| Body | xsd:XPolicy |
| Response |  |
|  | 200 – application/xml Policy created, returns the information of the created policy |
|  | 404 – NotFound – The Policy refers to a Nffg that is not existing. |
|  | 403 – Forbidden. The policy can’t be created because another policy with the same name is present. |
|  | 500 – Internal Server Error – in case NEO4J is not available or retrieve some errors. |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |
|  |  |

If overwrite assumes values different from y or n an error:

**Error 403 – Forbidden is returned. Please provide the query parameter: overwrite=y or overwrite=n. Alternatively do not specify the parameter overwrite (default actions y is assumed).**

**NB IS IT A PROBLEM TO HAVE THE SAME STATUS CODE FOR DIFFERENT PROBLEMS?!**

GET all policies

|  |  |
| --- | --- |
| GET | **/policies** |
| Content-Type | application/xml |
|  |  |
| Response | 200 application/xml  xsd:XPolicies  Contains the set of all policies stored in the service with verification results |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |
|  |  |

CREATE a set of policies

|  |  |
| --- | --- |
| POST | **/policies** |
| Query Parameter | ?overwrite=y  ?overwrite=n |
| Content-Type | application/xml |
| Accept | Application/xml |
| Response | 200 application/xml  xsd:XPolicies  Contains the set of all policies just created |
|  | 404 – NotFound – At least one policy refers to a Nffg that is not existing. |
|  | \*403 – Forbidden. The set of policies can’t be created because at least one policy witis present.  \*Only if ?overwrite=n is specifieid |
|  | \*\*403 – Forbidden – The parameter overwrite assumes a not allowed value (only y/n or no parameter are allowed) |
|  | 500 – Internal Server Error – in case NEO4J is not available or retrieve some errors. |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |
|  |  |

GET Policy given its name

|  |  |
| --- | --- |
| GET | **/policy/{name}** |
| Content-Type | application/xml |
| Response | 200 application/xml  xsd:XPolicy  Contains all the information of the given Policy |
|  | 200 / Body: xsd type: Policy |
|  | 404 – NotFound – In case the policy with the requested name is not stored in the database |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |
|  |  |

DELETE Policy given its name

|  |  |
| --- | --- |
| DELETE | **/policy/{name}** |
| Content-Type | application/xml |
| Body |  |
| Response | 200 application/xml  xsd:XPolicy  Contains the information of the deleted policy. |
|  | 404 – NotFound – The policy you want to delete is not stored in the database |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |
|  |  |

DELETE all stored policies

|  |  |
| --- | --- |
| DELETE | **/policies** |
|  |  |
|  |  |
| Response | 204 - No content |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |

UPDATE a Policy

|  |  |
| --- | --- |
| PUT | **/policy/{name}** |
| Content-Type | application/xml |
| Accept | application/xml  xsd:XPolicy |
| Response | 200 application/xml  xsd:XPolicy  Contains the information of the updated policy |
|  | 403 – Forbidden – The Nffg the policy refers to does not exist. |
|  | 404 – NotFound – The policy you want to update does not exists. |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |

**NOTE: The policy name cannot be modified; it identifies in a unique way the policy itself! All the others parameters, including the nffg the policy refers to, can be modified.**

VERIFICATION OF POLICIY

|  |  |
| --- | --- |
| POST | **/policy/{name}** |
| Content-Type | application/xml |
| Response | 200 application/xml  xsd:XPolicy  Contains the information of the verified policy, including all verification informations |
|  | 404 – NotFound – The policy you want to verify does not exists. |
|  | 500 – Internal Server Error – Problems while contacting Neo4j |
| In case of service not available | 404 - Problems, service not available – wrong URL, WS not started |