1. Add vocabulary

URL:	api/vocabulary/add
HTTP Method:	POST
Description:	Add new vocabulary to triple store, namespace need to be unique

Inputs:	{
	"name" : "<vocabulary name>",
	"namespace" : "<vocabulary namespace>",
	"path" : "<vocabulary path>"
	}

Response	{
	"message": "success or error message"
	}

example:
Request:
Request URL: http://localhost:8080/ManageVocabulary/api/vocabulary/add
Request Method: POST
Request Payload: {
"name": "foaf",
"namespace":"http://xmlns.com/foaf/0.1/",
"path":"http://xmlns.com/foaf/spec/index.rdf"}

Respond:
Response header:
Content-Length: 70
Content-Type: application/json
Date: Mon, 15 Jun 2015 07:13:25 GMT
Server: Apache-Coyote/1.1

{"message":"Vocabulary has been added successfully"}

2. Delete vocabulary

URL:	api/vocabulary/delete
HTTP Method:	POST
Description:	Delete existing vocabulary based on name and namespace.

Inputs:	{
	"name" : "<vocabulary name>",
	"namespace" : "<vocabulary namespace>",
	}

Response	{
	"message": "success or error message"
	}

example:

Request:
Remote Address: [::1]:8080
Request URL: http://localhost:8080/ManageVocabulary/api/vocabulary/delete
Request Method: POST
Status Code: 200 OK
Request Payload:
{name: "dcterms", namespace: "http://purl.org/dc/terms/"}

Response:
Response Header:
Content-Length: 72
Content-Type: application/json
Date: Mon, 15 Jun 2015 07:35:20 GMT
Server: Apache-Coyote/1.1

{"message":"Vocabulary has been deleted successfully"}

3. Search

URL:	api/vocabulary/search/{keyword}
HTTP Method:	GET
Description:	Searching classes and properties in vocabulary based on keyword matching.
Inputs:	keyword

Response	{
	"message": "<error message>"
	"result": {
	"value": "<class name or property name>"
	}
	}

example:

Request:
Remote Address: [::1]:8080
Request URL: http://localhost:8080/ManageVocabulary/api/vocabulary/search/name
Request Method: GET
Status Code: 200 OK

Respond:
Response header:
Content-Length: 299
Content-Type: application/json
Date: Mon, 15 Jun 2015 07:27:02 GMT
Server: Apache-Coyote/1.1

{"message":"Search successful",
"result":[{"value":"foaf:familyName"},{"value":"foaf:givenname"},{"value":"foaf:givenName"},{"value":"foaf:lastName"},{"value":"foaf:firstName"},{"value":"foaf:name"},{"value":"foaf:accountName"},{"value":"foaf:surname"},{"value":"foaf:family_name"}]}

4. Update vocabulary

URL:	api/vocabulary/update
HTTP Method:	POST
Description:	update vocabulary path based on vocabulary name and namespace

Inputs:	{
	"name": "<vocabulary name>",
	"namespace" : "<vocabulary namespace>",
	"path" : "<vocabulary path>"
	}

Response	{
	"message": "<error message>"
	}

5. Get all vocabulary name

URL:	api/vocabulary/getAll
HTTP Method:	GET
Description:	get the name of all vocabulary in triple store.
Inputs:	

Response	{
	"message": "<error message>"
	"result": {
	"name": "<vocabulary name>"
	"namespace": "<vocabulary namespace>"
	}
	}

example:
Request:
Remote Address: [::1]:8080
Request URL: http://localhost:8080/ManageVocabulary/api/vocabulary/getAll
Request Method: GET
Status Code: 200 OK

Respond:
Response header:
Content-Length: 196
Content-Type: application/json
Date: Mon, 15 Jun 2015 07:21:33 GMT
Server: Apache-Coyote/1.1

{"Message":"Get Vocabulary successfully",
"result":[{"name":"dcterms","namespace":"http:\/\/purl.org\/dc\/terms\/"},
        {"name":"foaf","namespace":"http:\/\/xmlns.com\/foaf\/0.1\/"}]}