## CMPE 282 – Cloud Services Homework 1

Student ID: 010034700

**Submitted by: Keertikeya Gupta** 

## Soln. 1.

## a.

Dev Tools: Eclipse EE Luna 4.4.1 on Ubuntu 14.04

**REST Client: HTML Page** 

REST Server: web/app component, servlet, Tomcat 7.0

REST Server (NoSQL) database component: MongoDB on Ubuntu

## b.

REST Server: used Servlet directly without relying on JAX-RS or Spring. Then deployed on Tomcat.

### c.

## **MongoDB Script:**

```
use CMPE282KeertikeyaGupta700
db.createCollection("employee")
db.createCollection("project")
db.employee.insert({_id: 1, first_name: "John", last_name: "Doe"})
db.project.insert({_id: 1, name: "ProjectX", budget:12345.67})
```

Soln. 2.

HTTP Method + URI	Description	HTTP Status – successful	HTTP Status –err, in addition 400, 500	Status
GET//rest/project/m	Retrieve employee with id m. Resp Body: XML	OK (200)	Not found (404)	Partial
GET//rest/project/n	Retrieve project with id n. Resp Body: XML	OK(200)	Not found (404)	Partial
POST//rest/employee	Create a new employee.  Req body: XML	Created (201)	Conflict (409)	Done
POST//rest/project	Create a new project. Req body: XML	Created (201)	Conflict (409)	Done
PUT//rest/project/m	Update employee with id m. Req body XML	OK (200)	Not found (404)	Done
PUT//rest/project/n	Update project with id n. Req body XML	OK (200)	Not found (404)	Done
DELETE//rest/project/m	Delete employee with id m	OK (200)	Not found (404)	Done
DELETE//rest/project/n	Delete project with id n	OK (200)	Not found (404)	Done
GET	Retrieve all employees. Resp body XML	OK (200)	Not found (404)	Partial
GET//rest/project	Retrieve all projects. Resp body XML	OK (200)	Not found (404)	Partial

**GET/.../rest/employee/m** is partially implemented. They data is being retrieved but it is not in XML format.

**GET/.../rest/project/m** is partially implemented. They data is being retrieved but it is not in XML format. **GET/.../rest/employee** is partially implemented. They data is being retrieved but it is not in XML format. **GET/.../rest/project** is partially implemented. They data is being retrieved but it is not in XML format.

## Database before any GET/PUT/POST/DELETE operations:

```
🔞 🖨 🗈 kk@ubuntu: ~
mm/transparent hugepage/enabled is 'always'.
2016-03-01T17:02:01.756-0800 I CONTROL [initandlisten] **
                                                                  We suggest set
ting it to 'never'
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten]
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten] ** WARNING: /sys/kernel/
mm/transparent_hugepage/defrag is 'always'.
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten] **
                                                                  We suggest set
ting it to 'never'
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten]
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikeyaDB
                           0.000GB
local
                           0.000GB
mydb
                           0.000GB
MongoDB Enterprise > date; show dbs
2016-03-03T07:56:31.060-0800 E QUERY
                                        [thread1] SyntaxError: missing; before
statement @(shell):1:11
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikeyaDB
                           0.000GB
local
                           0.000GB
mydb
                           0.000GB
MongoDB Enterprise > use
```

```
🔞 🖹 🗈 kk@ubuntu: ~
mm/transparent_hugepage/defrag is 'always'.
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten] **
                                                                  We suggest set
ting it to 'never'
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten]
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikeyaDB
                           0.000GB
local
                           0.000GB
                           0.000GB
mydb
MongoDB Enterprise > date; show dbs
2016-03-03T07:56:31.060-0800 E QUERY [thread1] SyntaxError: missing ; before
statement @(shell):1:11
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikevaDB
                           0.000GB
local
                           0.000GB
mydb
                           0.000GB
MongoDB Enterprise > use CMPE282KeertikeyaGupta700
switched to db CMPE282KeertikeyaGupta700
MongoDB Enterprise > show collections
employee
project
MongoDB Enterprise >
```

```
🔞 🖨 🗈 kk@ubuntu: ~
ting it to 'never'
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten]
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikeyaDB
                           0.000GB
local
                           0.000GB
                           0.000GB
MongoDB Enterprise > date; show dbs
2016-03-03T07:56:31.060-0800 E QUERY [thread1] SyntaxError: missing ; before
statement @(shell):1:11
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikevaDB
                           0.000GB
local
                           0.000GB
mydb
                           0.000GB
MongoDB Enterprise > use CMPE282KeertikeyaGupta700
switched to db CMPE282KeertikeyaGupta700
MongoDB Enterprise > show collections
employee
project
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
MongoDB Enterprise >
```

```
🚳 🖨 📵 kk@ubuntu: ~
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikeyaDB
                            0.000GB
local
                            0.000GB
mydb
                            0.000GB
MongoDB Enterprise > date; show dbs
2016-03-03T07:56:31.060-0800 E QUERY
                                        [thread1] SyntaxError: missing; before
statement @(shell):1:11
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikeyaDB
                            0.000GB
local
                            0.000GB
mydb
                            0.000GB
MongoDB Enterprise > use CMPE282KeertikeyaGupta700
switched to db CMPE282KeertikeyaGupta700
MongoDB Enterprise > show collections
employee
project
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : ProjectX", "budget" : 12345.67 }
MongoDB Enterprise >
```

## **PUT Operations:**

Send

## CMPE 282 HW 1 Stundet ID: 010034700 Stundet Name: Keertikeya Gupta URL CMPE282KeertikeyaGupta700/rest/employee Make sure URL contains proper path. E.g. CMPE282KeertikeyaGupta700/rest/... HTTP Method PUT \*\*Cemployee> \*\*cid>1c/ id> \*\*cfirst\_name>Alice</first\_name> \*\*clast\_name>Bobk/last\_name> \*\*clemployee> \*\*cl



## **CMPE 282 HW 1**

Stundet ID: 010034700

Stundet Name: Keertikeya Gupta

URL CMPE282KeertikeyaGupta700/rest/employee

Make sure URL contains proper path. E.g. CMPE282KeertikeyaGupta700/rest/...

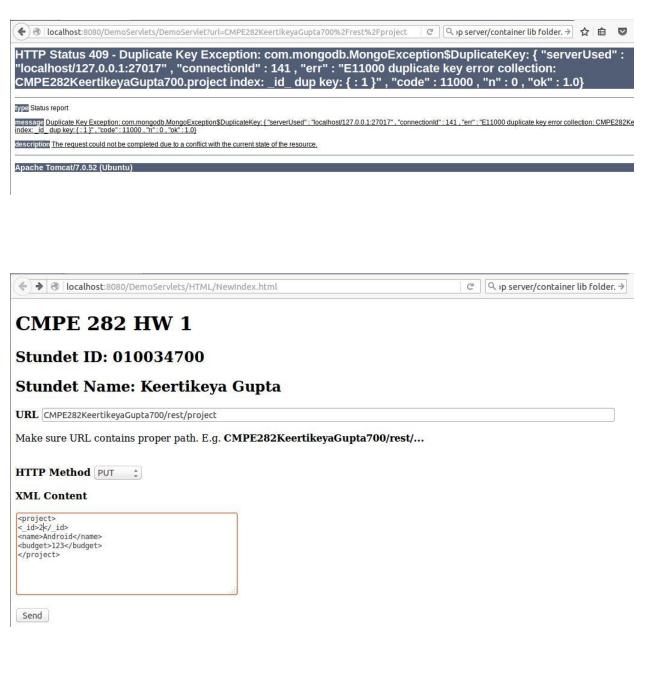
HTTP Method PUT 

XML Content

cemployee>
< id>> id>> // id> // id> </ri> </ri> 
- first\_name>Alice 
- last\_name>Bob</last\_name></le>

Send







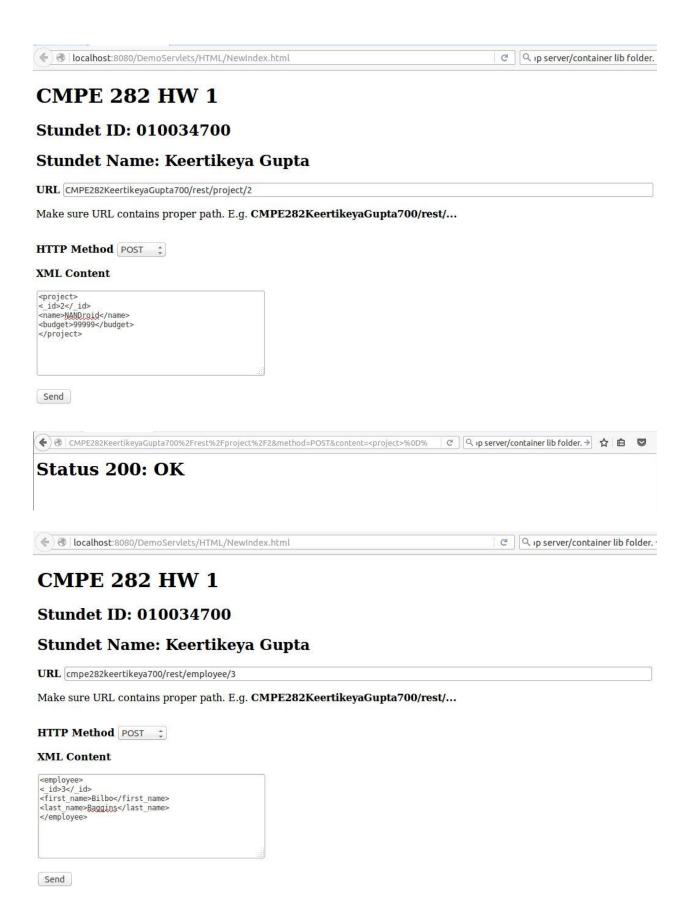
```
🔞 🖨 🗈 kk@ubuntu: ~
2016-03-01T17:02:01.756-0800 I CONTROL [initandlisten] ** WARNING: /sys/kernel/
mm/transparent hugepage/enabled is 'always'.
2016-03-01T17:02:01.756-0800 I CONTROL [initandlisten] **
                                                                     We suggest set
ting it to 'never'
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten]
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten] ** WARNING: /sys/kernel/
mm/transparent_hugepage/defrag_is_'always'.
We suggest set
ting it to 'never'
2016-03-01T17:02:01.757-0800 I CONTROL [initandlisten]
MongoDB Enterprise > show dbs
CMPE282KeertikeyaGupta700 0.000GB
keertikeyaDB
                            0.000GB
local
                            0.000GB
mydb
                            0.000GB
MongoDB Enterprise > use CMPE282KeertikeyaGupta700
switched to db CMPE282KeertikeyaGupta700
MongoDB Enterprise > show collections
employee
project
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
{ "_id" : 2, "name" : "Android", "budget" : 123 }
MongoDB Enterprise > db.employee.find()
 "_id" : 1, "first_name" : "new", "last_name" : "object" }
"_id" : 2, "first_name" : "Alice", "last_name" : "Bob" }
Manager DB Enterprise >
```

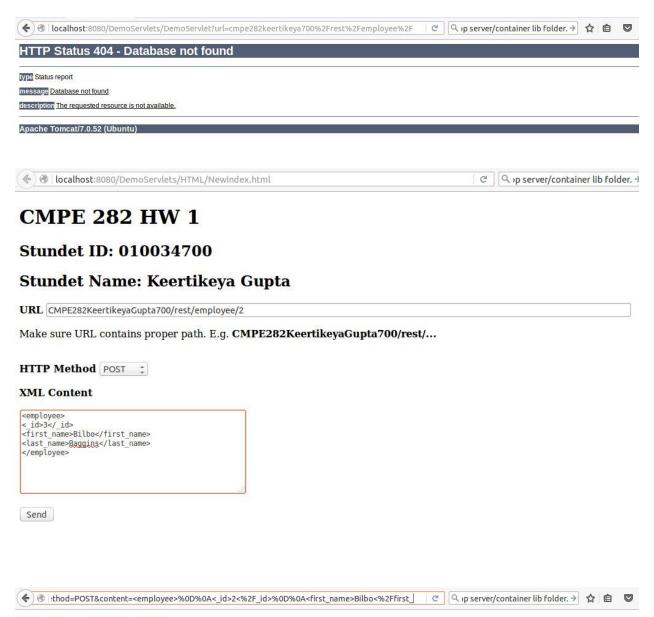
## **POST Operations:**

# CMPE 282 HW 1 Stundet ID: 010034700 Stundet Name: Keertikeya Gupta URL CMPE282KeertikeyaGupta700/rest/employee/2 Make sure URL contains proper path. E.g. CMPE282KeertikeyaGupta700/rest/... HTTP Method POST XML Content | cemployee> | clast\_name>Harry</first\_name> | clast\_name>Potter</last\_name> | clast\_name>Potter</last\_name> | clast\_name> | c







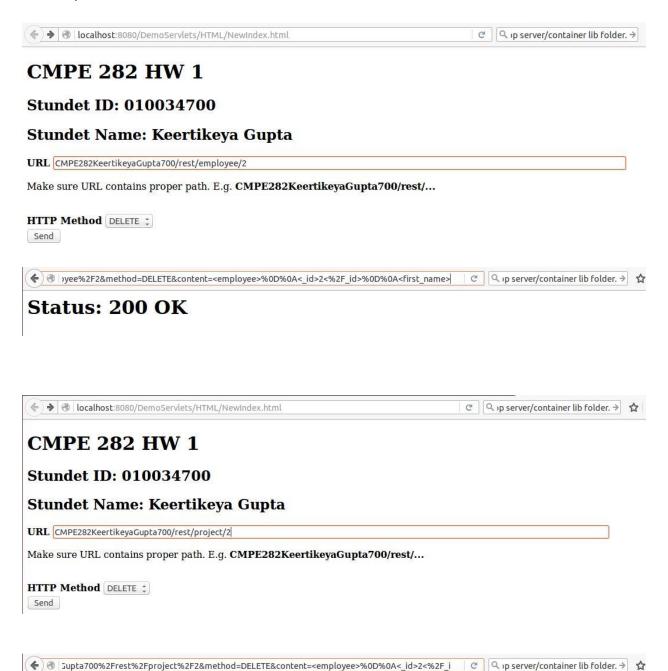


Status 200: OK

```
🔞 🖨 🗈 kk@ubuntu: ~
keertikevaDB
                                  0.000GB
local
                                  0.000GB
mvdb
                                  0.000GB
MongoDB Enterprise > use CMPE282KeertikeyaGupta700
switched to db CMPE282KeertikeyaGupta700
MongoDB Enterprise > show collections
employee
project
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
{ "_id" : 2, "name" : "Android", "budget" : 123 }
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
{ "_id" : 2, "first_name" : "Alice", "last_name" : "Bob" }
MongoDB Enterprise > db.employee.find()
Software Center rise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
{ "_id" : 2, "name" : "NANDroid", "budget" : 99999 }
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
{ "_id" : 2, "name" : "NANDroid", "budget" : 99999 }
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
{ "_id" : 2, "first_name" : "Bilbo", "last_name" : "Baggins" }
MongoDB Enterprise >
```

## **Delete Operations:**

Status: 200 OK



```
🚳 🖨 🗊 kk@ubuntu: ~
switched to db CMPE282KeertikeyaGupta700
MongoDB Enterprise > show collections
emplovee
project
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
{ "_id" : 2, "name" : "Android", "budget" : 123 }
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
{ "_id" : 2, "first_name" : "Alice", "last_name" : "Bob" }
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
{ "_id" : 2, "first_name" : "Harry", "last_name" : "Potter" }
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
{ "_id" : 2, "name" : "NANDroid", "budget" : 99999 }
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
{ "_id" : 2, "name" : "NANDroid", "budget" : 99999 }
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
  "_id" : 2, "first_name" : "Bilbo", "last_name" : "Baggins" }
MongoDB Enterprise > db.project.find()
{ "_id" : 1, "name" : "ProjectX", "budget" : 12345.67 }
MongoDB Enterprise > db.employee.find()
{ "_id" : 1, "first_name" : "new", "last_name" : "object" }
 DB Enterprise >
```

## **Known Issues:**

Session manage has not been implemented fully, so there are a few client side errors.

E.g. After a successful PUT operation, if you hit the back button in the browser and repeat the same PUT operation, it will still give a success message though actually no data is being inserted into the database.

## Additional unique design feature:

Made the client by myself, and will be working on it to make it better.