CLASSEC Development

# 

# Developer Priya Malamkar

# Story

**Tagging implementation for all cloud providers –** Implement tagging feature for each instance across all cloud providers.

# Tasks

* **Create a table in database for tags –** Create a table with columns Instance Id, Cloud, Tagname to store tagname for instance of each cloud provider.
* **Create a POST API to insert tagnames in database for tags –** Develop an API of POST method to insert data into above created table.
* **Update database tables to add column for tagname –** Update static data tables for each cloud provider to add column ‘Tagname’.
* **Implement tagging feature for static data –** Update insert functions for each cloud provider to add ‘Tagname’ in database tables. The value for ‘Tagname’ can be obtained from tags table (specified in step 1).

# Environment

Go version go1.7

# Modules impacted

* Database
* Server
* DAO

# Files added

* classec\src\gclassec\structs\tagstruct\tags\_struct.go
* classec\src\gclassec\dao\instancetags\tags.go

# Files changed

* classec\src\gclassec\server\main.go
* classec\src\gclassec\dao\hosinsert\insert\_hos\_instances.go
* classec\src\gclassec\dao\openstackinsert\openstack\_insert.go
* classec\src\gclassec\dao\azureinsert\azureinsert.go
* classec\src\gclassec\dao\vmwareinsert\vmwareinsert.go
* classec\src\gclassec\structs\azurestruct\azure\_struct.go
* classec\src\gclassec\structs\azurestruct\azure\_response\_struct.go
* classec\src\gclassec\structs\hosstruct\hos\_struct.go
* classec\src\gclassec\structs\openstackInstance\openstack\_structs.go
* classec\src\gclassec\structs\vmwarestructs\vmware\_structs.go
* classec\src\gclassec\controllers\azurecontroller\azure\_controller.go
* classec\src\gclassec\goclienthos\hosstaticdynamic\single\_api.go
* classec\src\gclassec\controllers\vmwarecontroller\vmware\_connect.go

# Dependencies (Libraries)

* "github.com/go-sql-driver/mysql"
* "github.com/jinzhu/gorm"

# Testcases added

* Test whether data is inserted into tags table using the POST API.
* Test whether ‘Tagname’ column in static data tables is updated for instances that are in the tags table.
* Check whether the tags table data is not nil while updating static data table.

# challenges faced

* In the insert functions for each cloud provider, the code to update ‘Tagname’ needs to be separated from the code to insert all other columns because ‘Instance Id’ from tags table need to be compared to ‘Instance Id’ of static data table and according to the comparison ‘Tagname’ column should be updated.

# Further comments

**Example URL:**

<http://110.110.112.203:9000/instancetag/421f8785-ac51-9502-b19d-173ef7209f8f>

“instanceid” is specified in the url path.

**Example POST request body:**

[{"instancename":"2012 DC","cloud":"vmware","tagname":"Testing"}]

Or

[{"cloud":"vmware","tagname":"Testing"}]

“instancename” key is optional.