# **Deploy Applause Website to Google cloud**

## **Table of Contents**

1. [Installation and Setup](#_1._Installation_and)

2. [Create Mongodb Instance](#_2._Create_Mongodb)

3. [Connect to Virtual Machine instance.](#_3._Connect_to)

4. [Export mongodb database](#_3._Export_applause).

5. [Import applause database](#_4._Import_applause)

6. [Setting mongodb.conf to access it from nodejs application](#_5.Setting_mongodb.conf_to)

7. [Create New User and set Permissions to Applause database](#_6.Create_New_User)

8[. Setting Mongodb GUI : Rockmongo](#_7._Setting_Mongodb)

9. [Edit database details](#_9._Setting_database)

10[. Create and Use Google Cloud Storage Bucket](#_10._Create_and)

11. [Deploy the Application](#_10.Deploy_the_Application)

12. [Backup of Images from cloud storage to local](#_12._Backup_of)

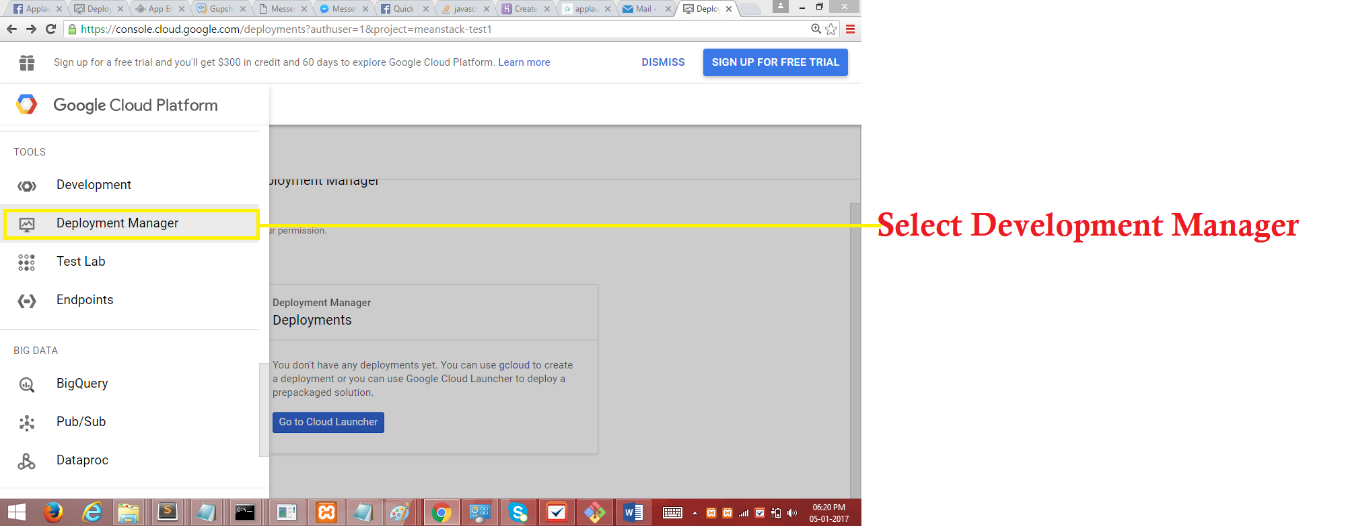
## **1. Installation and Setup**

* [Install](https://dl.google.com/dl/cloudsdk/channels/rapid/GoogleCloudSDKInstaller.exe) google cloud sdk.
* Run ‘gcloud init’ command on terminal and Select your gmail account of Project to which you want to deploy
* Select the project which need to be deployed

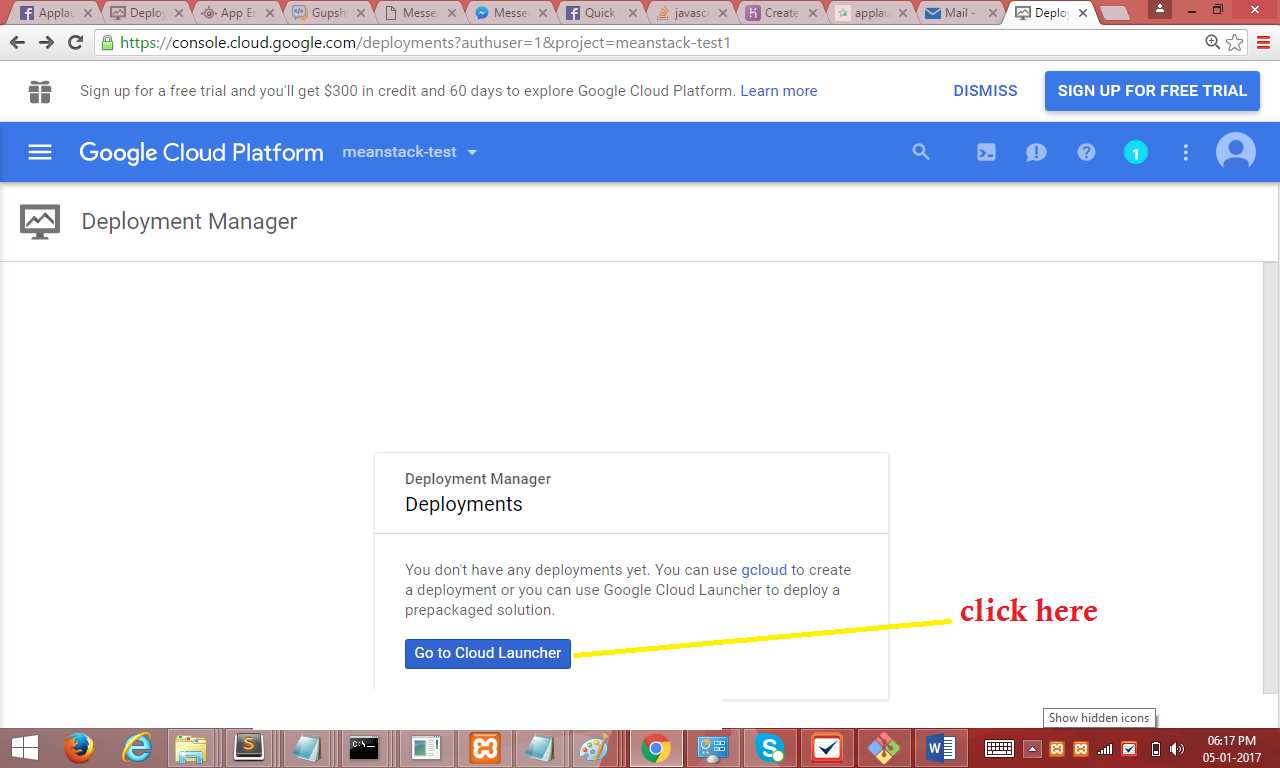
## **2. Create Mongodb Instance**

This section explains how to create Mongodb Mean instance.

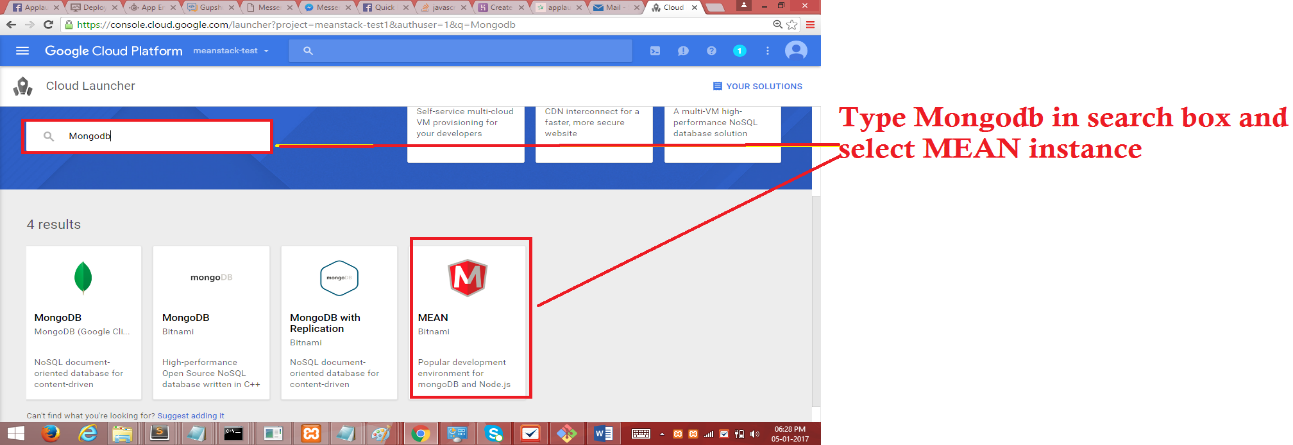
* Goto <https://console.cloud.google.com>
* Select appropriate project from the projects list
* Select Deployment Manager



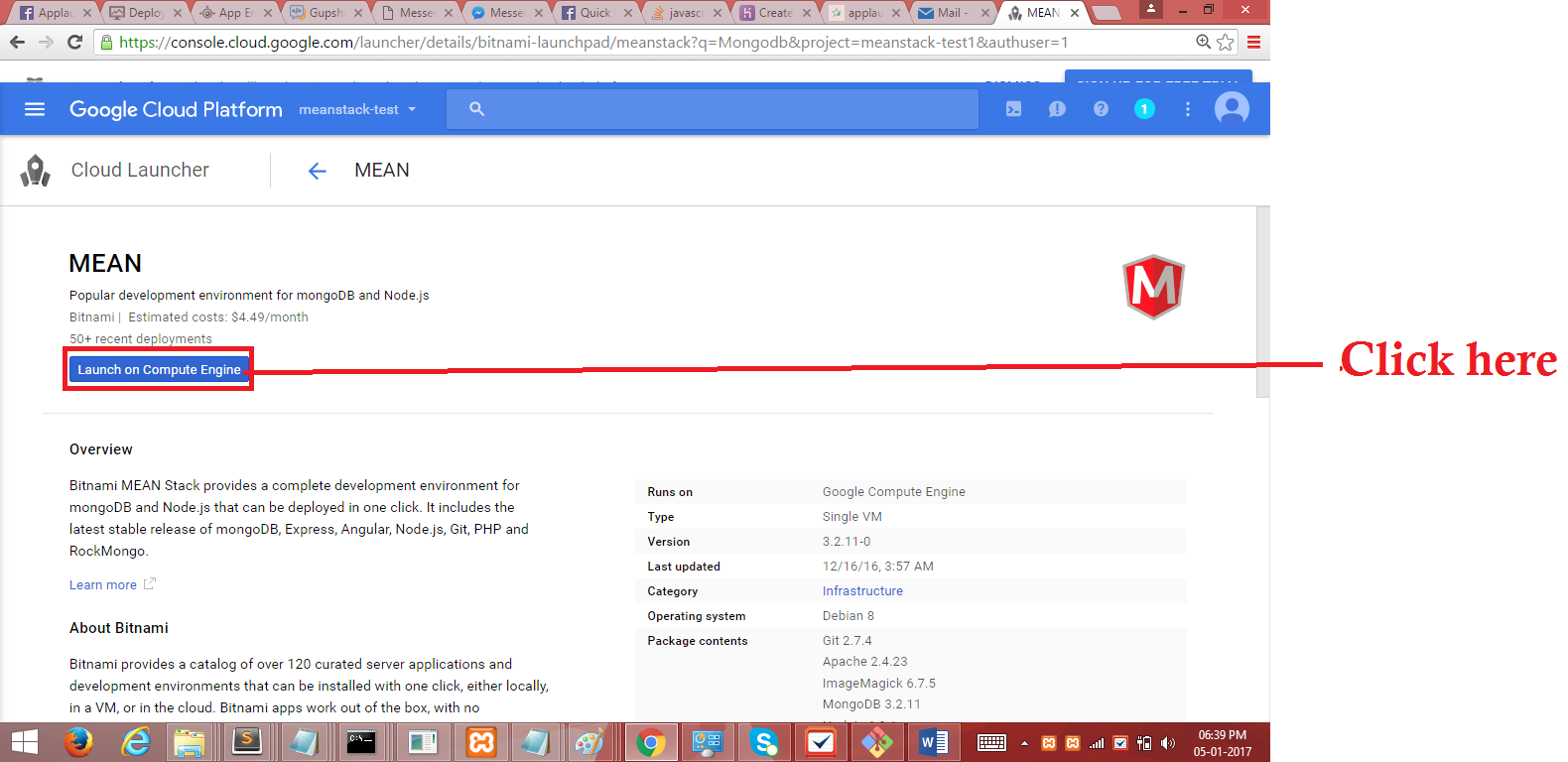
* Click on Go to Cloud Launcher (Present in hamburger option)



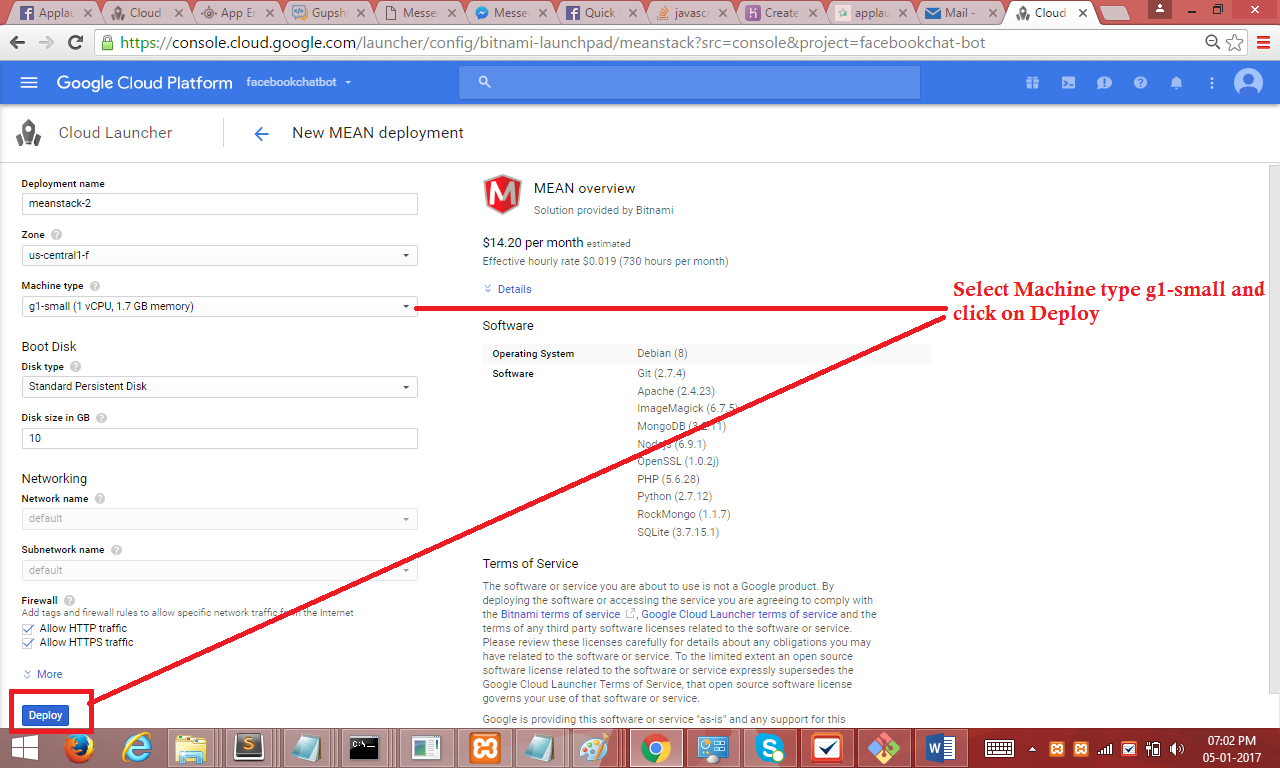
* Type Mongodb in search box.
* Select Mean Instance. (icon can be different from the screenshot)



* Click on “Launch on Compute Engine”



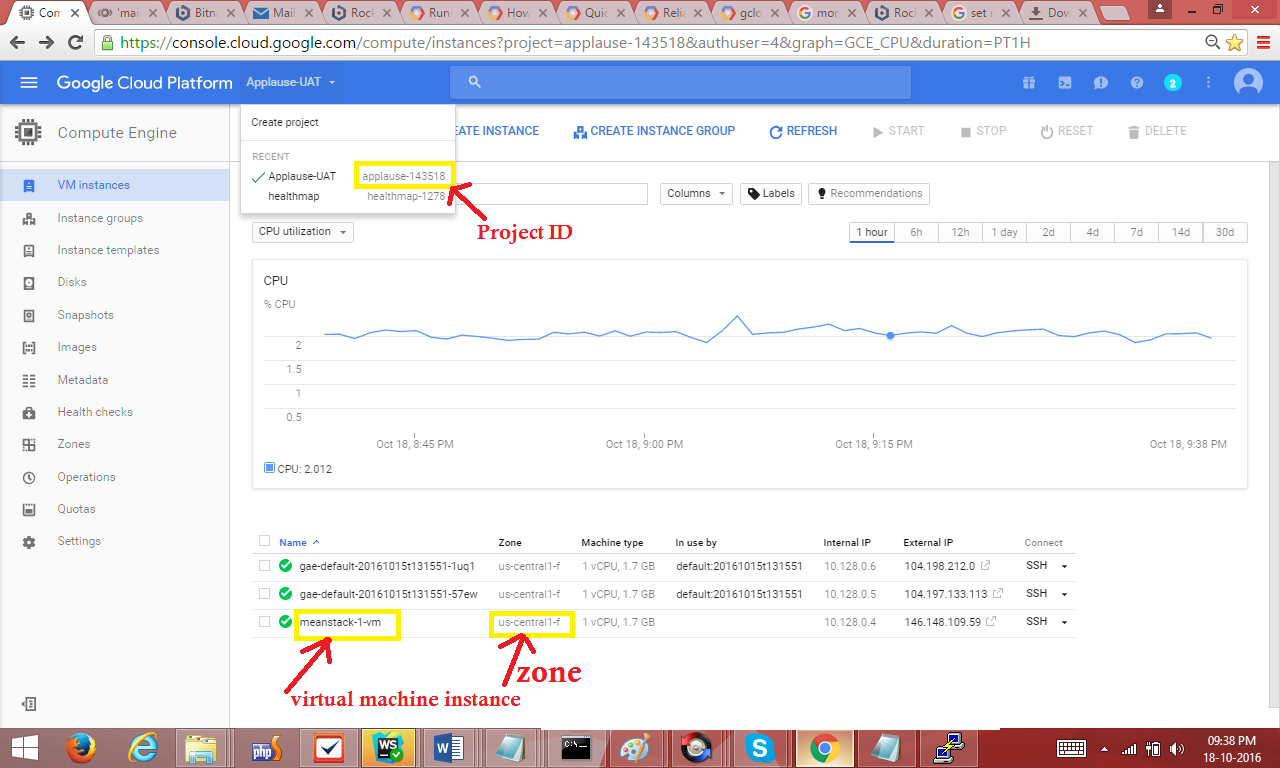
* Select Machnie Type g1-small and click on Deploy



## **3. Connect to Virtual Machine instance.mongo**

This section explains how to connect to compute engine virtual machine from local computer

* Goto compute engine (From hamburger menu icon of google cloud console)



* Run this command in google cloud sdk (command prompt/ terminal)

gcloud compute --project “project-id" ssh --zone "your vm instance zone" "virtual machine instance"

(if prompted for passphrase hit enter for empty passphrase)

After Virtual Machine started you can Export and import database.

## **4. Export applause database. (Will not need this step if already have database dump. Please skip to step 5 if sample-db is present in docs/google\_deployment.)**

This section explains how to export applause database

Connect to virtual machine [ run command in local computer]

* gcloud compute --project "applause-143518" ssh --zone "us-central1-f" "meanstack-1-vm"

Export Mongo db database in virtual machine [ run command in VM]

mongodump --db applause --host meanstack-1-vm --username=root --password=applause123

* This command creates a dump folder with applause directory with collection files

**Copy files from Virtual machine to local computer [ run command in local computer]**

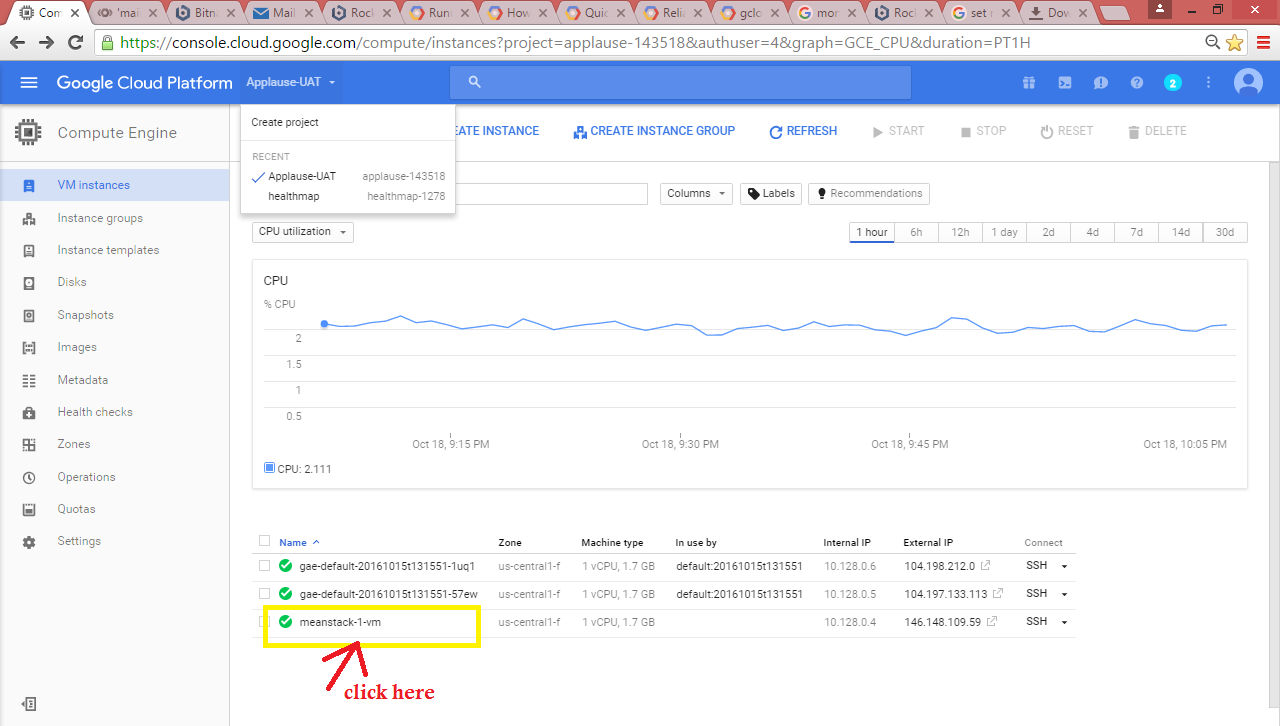
* gcloud compute copy-files "Sumayya@meanstack-1-vm:dump" "./"
* This command copy files from virtual machine to current directory (./).
* You can copy dump folder from VM to any directory by specify the path instead of “./” like “C:\applause”

path to dump is path to virtual machine dump folder

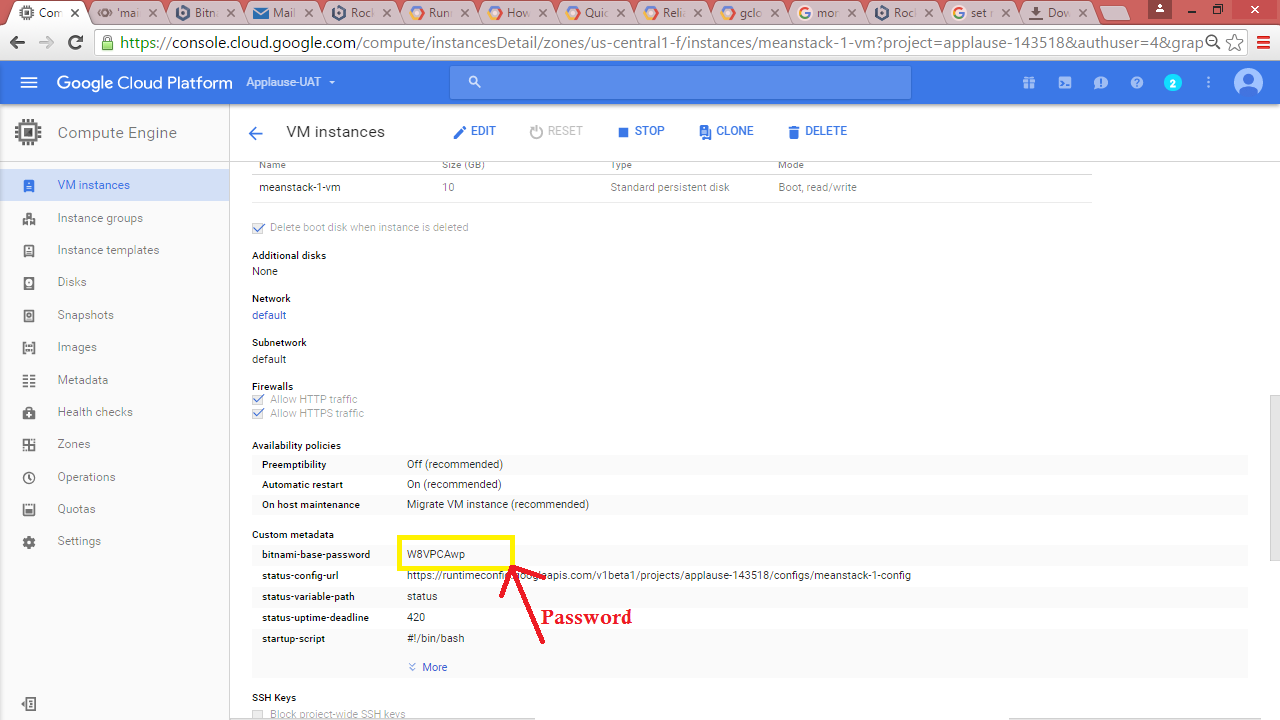
## **5. Import applause database**

This section explains how to import applause database to virtual machine.

* Run gcloud init command and Select your gmail account of Project with Meanstack instance, to which you want to import applause database. (Skip this step if already done.)
* Goto Compute Engine (In google cloud console ui)

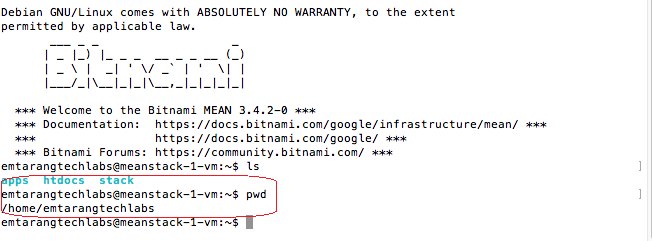
****

* Copy bitnami-base-password found under Custom Metadata (click on vm instance name and copy the password)



**Copy files from local machine to VM**

* On vm instance terminal (obtained from step 3) enter command pwd and copy the path



* In new command prompt/terminal enter following command

gcloud compute copy-files <path to mongodb backup> <compute engine instance name>:< path on vm>

path on vm : Obtained from pwd command.

(backup of db can be found in /docs/google\_deployement/sample-db)

Eg.

gcloud compute copy-files "/Users/emtarangtechlabs/Desktop/ApplauseServer\_Setup/applause\_backend\_portal/docs/google\_deployment/sample-db" "meanstack-1-vm:/home/emtarangtechlabs"

If promted for “choose a zone” select vm instance zone

* Run the command (on vm instance ssh ed into in step 3) -

mongorestore --authenticationDatabase admin --port 27017 --host <vm-instance-name> -u root -p <bitnami password> -d applause <Path To Applause Dump>

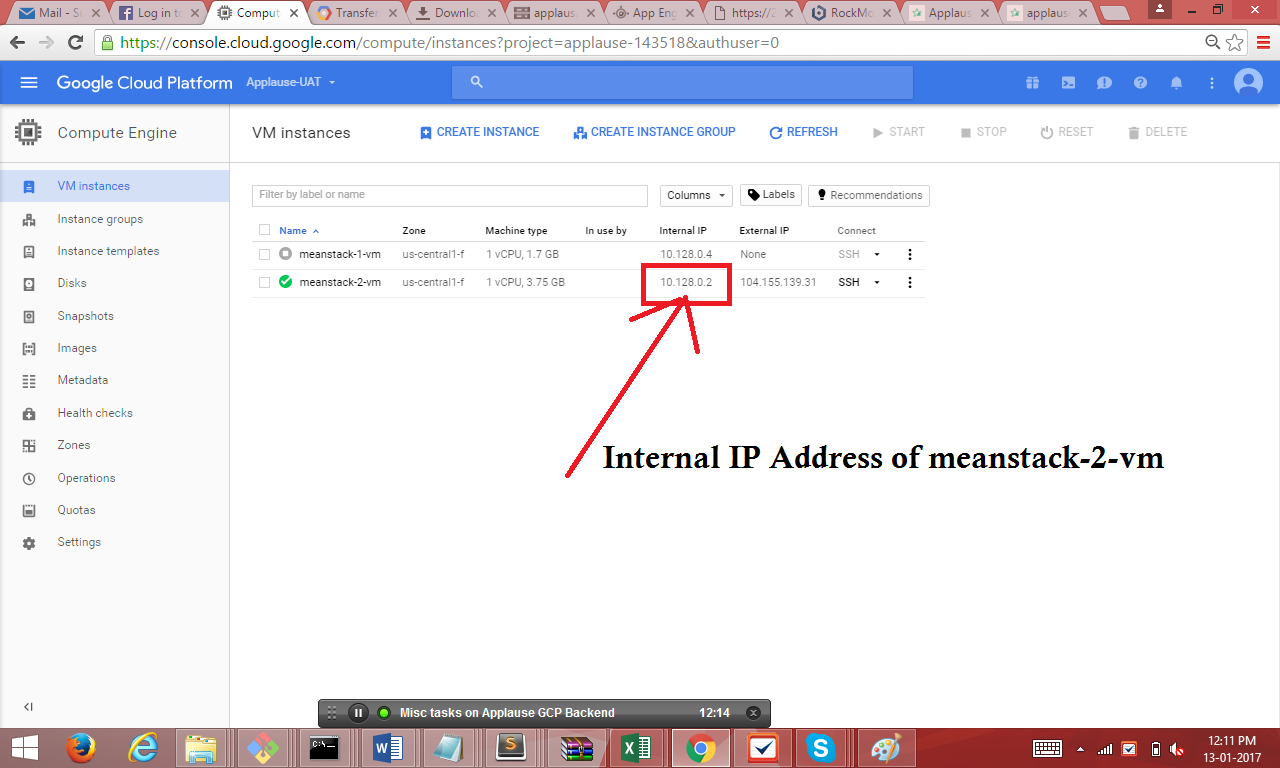
* Path to Applause Dump for example : sample-db/dump/applause (the directory name of the applause db backup copied onto the vm eg. Sample-db/empty-db)
* Example-

mongorestore --authenticationDatabase admin --port 27017 --host meanstack-1-vm --username root --password sgZJ1aZW -d applause sample-db/dump/applause

## **6.Setting mongodb.conf to access it from nodejs application**

This section explain how to set to mongodb.conf file in order to access mongodb from application.

It is necessary to add internal IP address of virtual machine in order to access Mongodb database from applause backend application.



Add internal IP address of mongodb vm to bindip (on vm instance)

cd /opt/bitnami/mongodb

* sudo chmod 726 mongodb.conf

vi mongodb.conf

Then change the following lines with replacing the proper values.

* set bind\_ip = 127.0.0.1,Internal IP Address of vm (to insert press i)

(note: add internal ip using comma. Don’t replace it)

* Save mongodb.conf file and quit from vi editor (to save press “esc:wq” and enter)

Then restart the mongo db using following command.

sudo /opt/bitnami/ctlscript.sh restart

## 

## **7. Change Admin Password Create New User and set Permissions to Applause database (on vm instance)**

This section explains how to change admin password , create new user for applause database and set roles dbAdmin and readWrite permission

**Edit the /opt/bitnami/mongodb/mongodb.conf file and replace the following line in 2 places**

**To edit enter the following command –**

cd /opt/bitnami/mongodb

vi mongodb.conf

**Then change the following lines as mentioned below (change needed in 2 places as this statement is present twice)**

setParameter = enableLocalhostAuthBypass=0

with:

#setParameter = enableLocalhostAuthBypass=0

To save press “esc :wq” and hit enter

**Change Admin Passsword**

* Enter following command to access the admin db

mongo admin --username root --password <bitnami-password>

* to change admin password enter following commands

db = db.getSiblingDB('admin')

db.changeUserPassword("root", "newpassword")

to exit( Press Ctrl +c)

**Create New User and set Applause db permissions**

**sudo /opt/bitnami/ctlscript.sh restart**

mongo admin --username root --password <new\_password>

db = db.getSiblingDB(‘applause’)

db.createUser( { user: "root", pwd: "applause123", roles: [ "readWrite", "dbAdmin" ]} )

**Revert the modifications made to /opt/bitnami/mongodb/mongodb.conf by replacing**: (In 2 places)

cd /opt/bitnami/mongodb

* sudo chmod 726 mongodb.conf

vi mongodb.conf

#setParameter = enableLocalhostAuthBypass=0

with:

setParameter = enableLocalhostAuthBypass=0

**Restart the MongoDB server.**

/opt/bitnami/ctlscript.sh restart

## **8. Setting Mongodb GUI : Rockmongo (not important for project deployment)**

This section explain how to set mongodb GUI using Rockmongo

* cd /opt/bitnami/apps/rockmongo/conf
* set write permission for httpd-app.conf file
* sudo chmod 726 httpd-app.conf
* vi httpd-app.conf
* Edit line number 31

1. <IfVersion >= 2.3>
2. Require local
3. </IfVersion>

Replace with

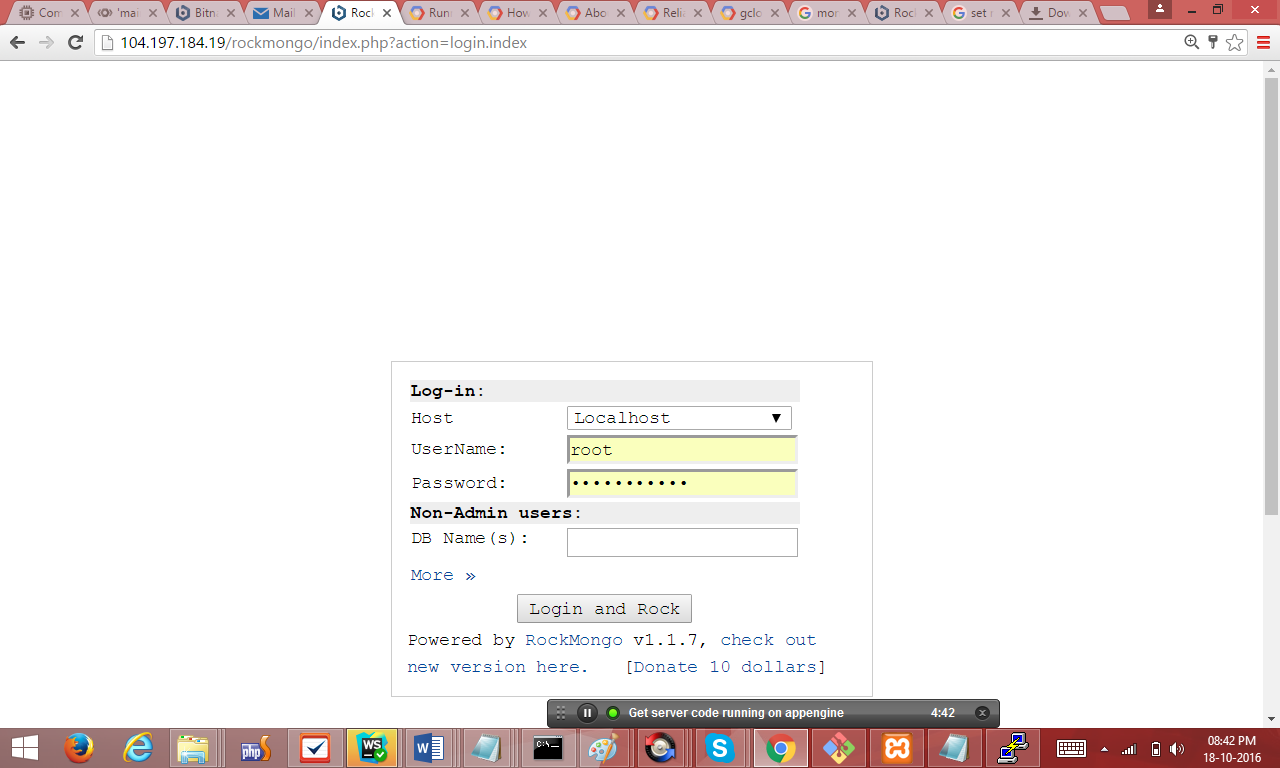
1. <IfVersion >= 2.3>
2. Require all granted
3. </IfVersion>

* Save the file

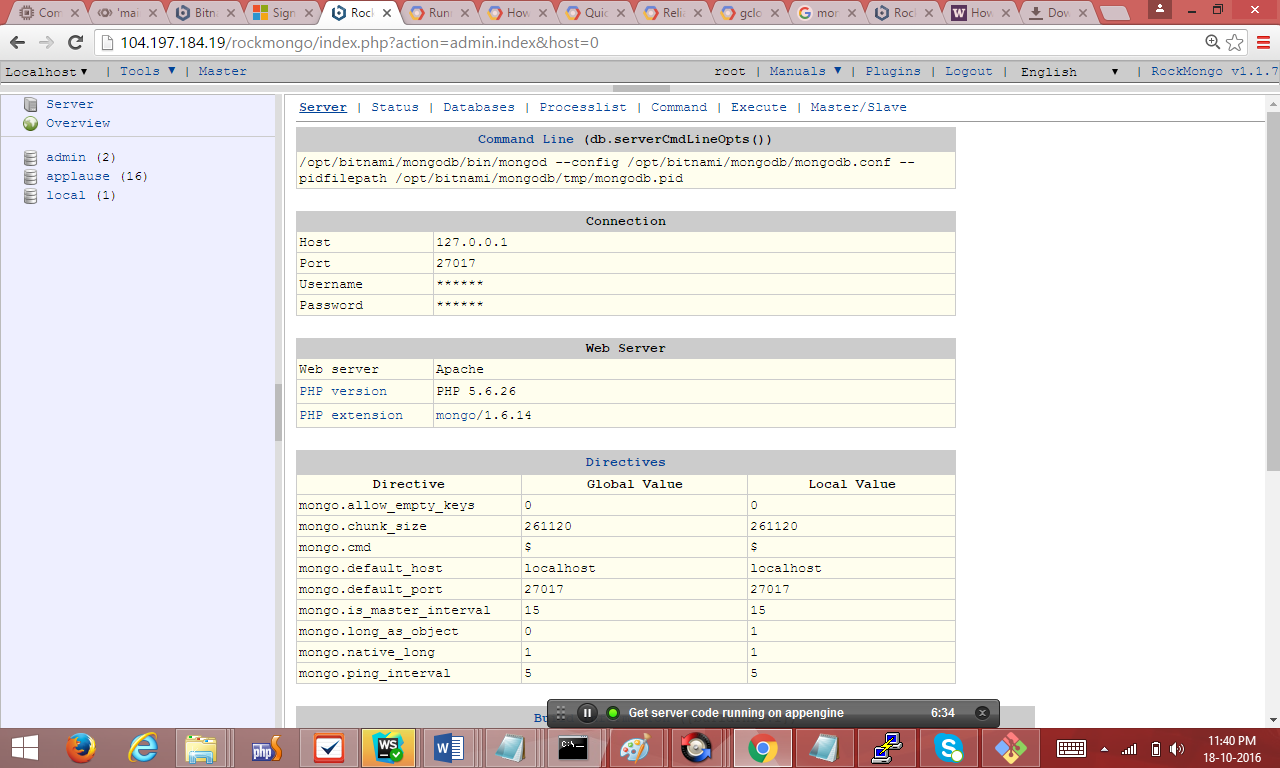
sudo /opt/bitnami/ctlscript.sh restart

Now you can access mongodb database with following url

* <http://external> IP of VM/rockmongo/index.php?action=login.index



* Enter Username and password and click on login



## **9. Setting database details (In the code)**

This section explains how to edit database details in application.

(Make these changes in the cloned code from bitbucket)

change config-dev.js file under applause\_backend\_portal/server/deploy directory

(we have created 3 folders for different environments. Make following changes )

database : {

host : 'internal IP address of VM',

user : 'database\_user',

password : 'database\_password',

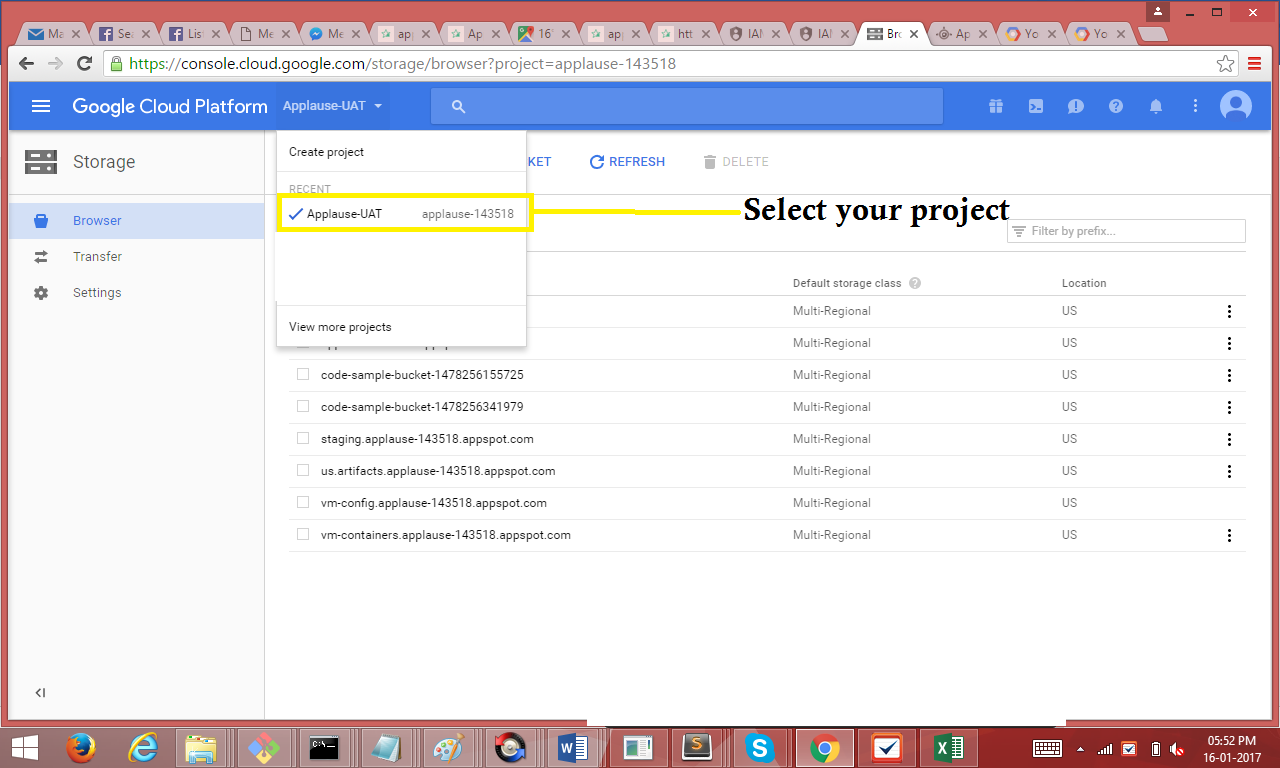
database : 'applause' }

## **10. Create and Use Storage bucket**

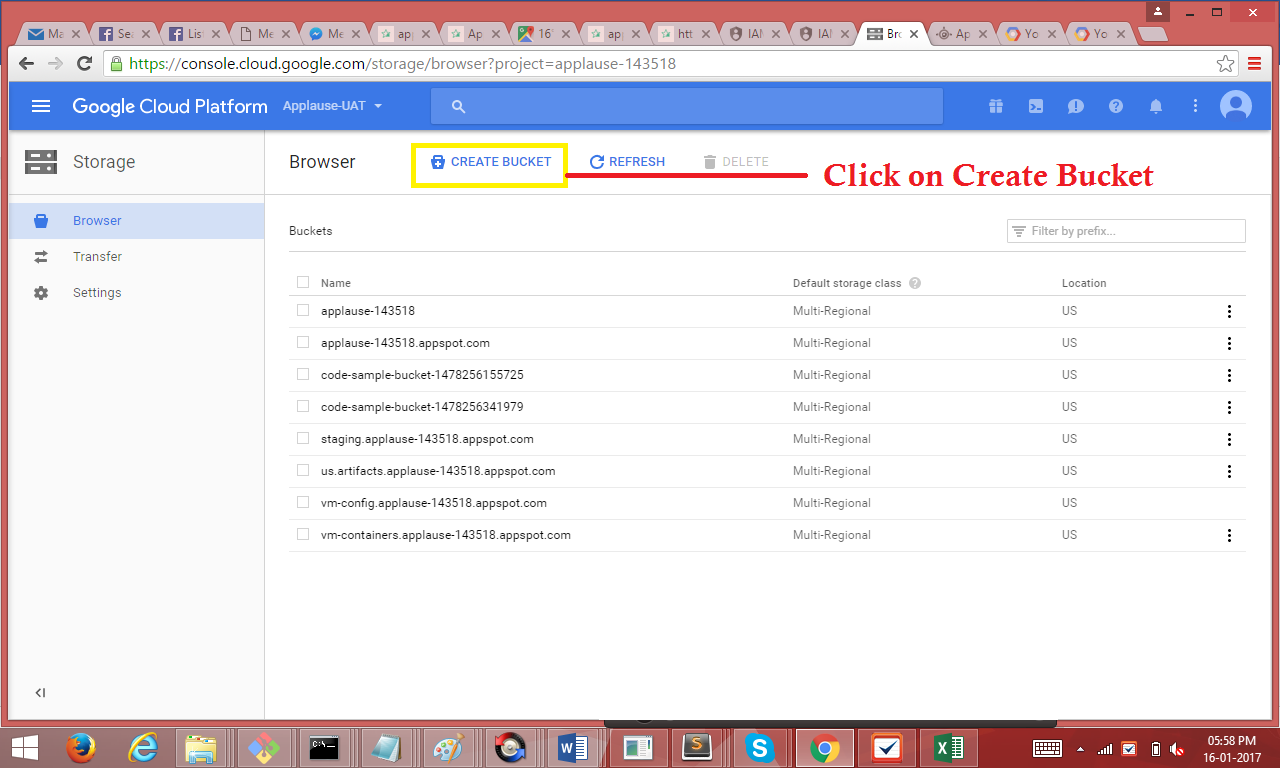
This section explains how to create, setting permissions and upload files to google cloud storage bucket for storing and accessing images from bucket in application

**10.1 Create Google Storage bucket**

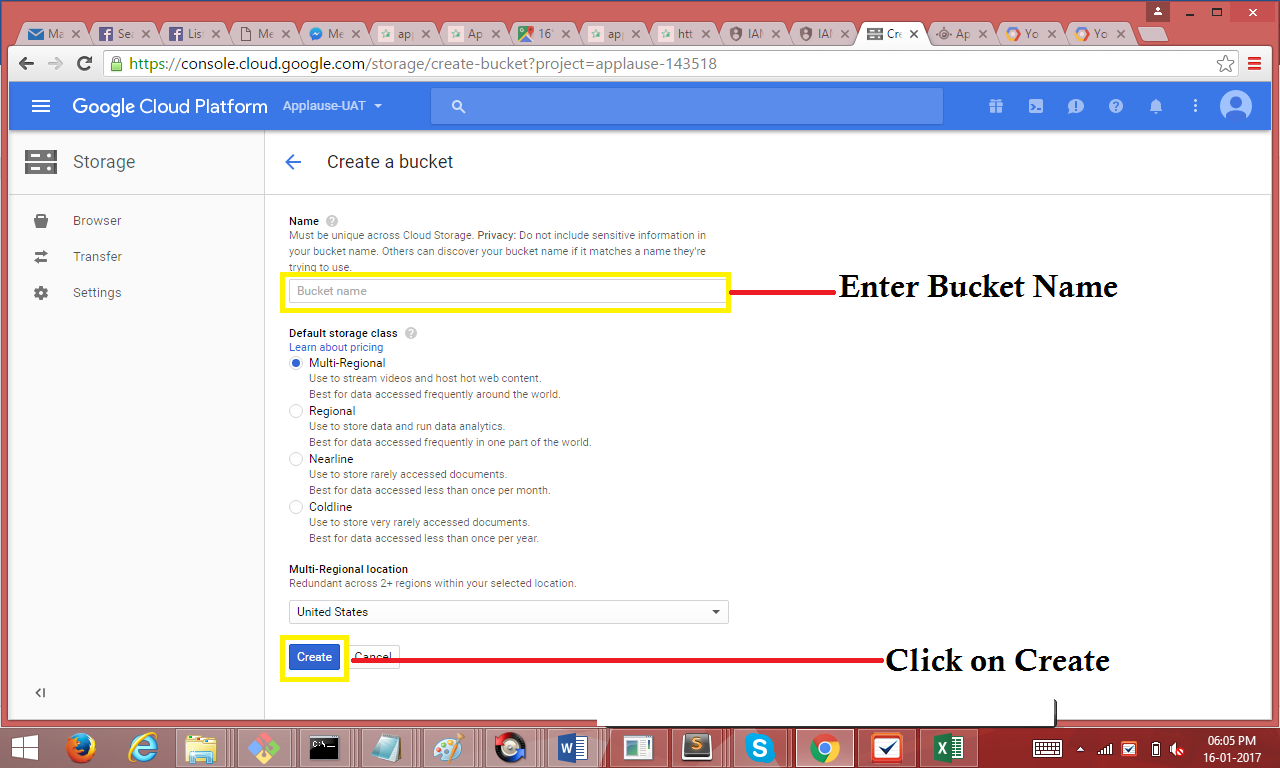
* Goto url <https://console.cloud.google.com/storage>
* Select you Project



* Click on Create Bucket



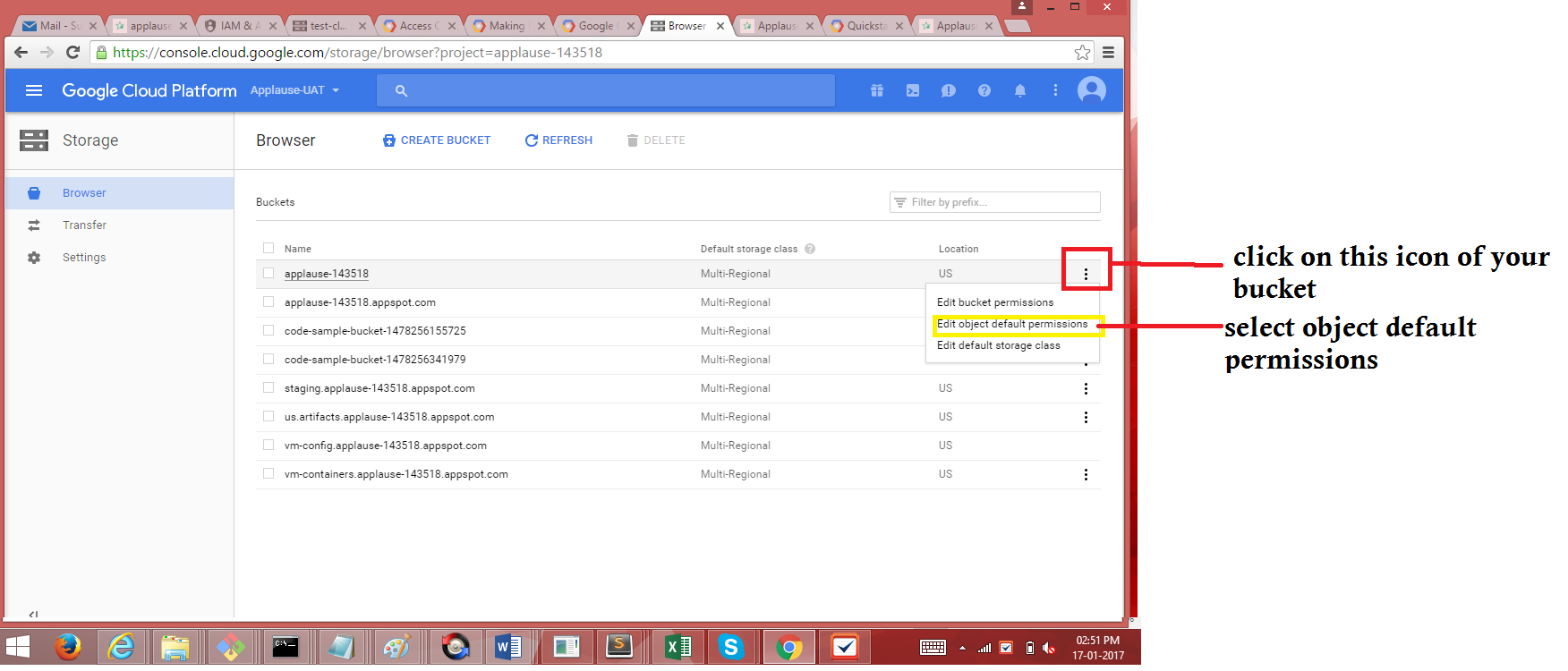
* Enter Bucket name and Click on Create



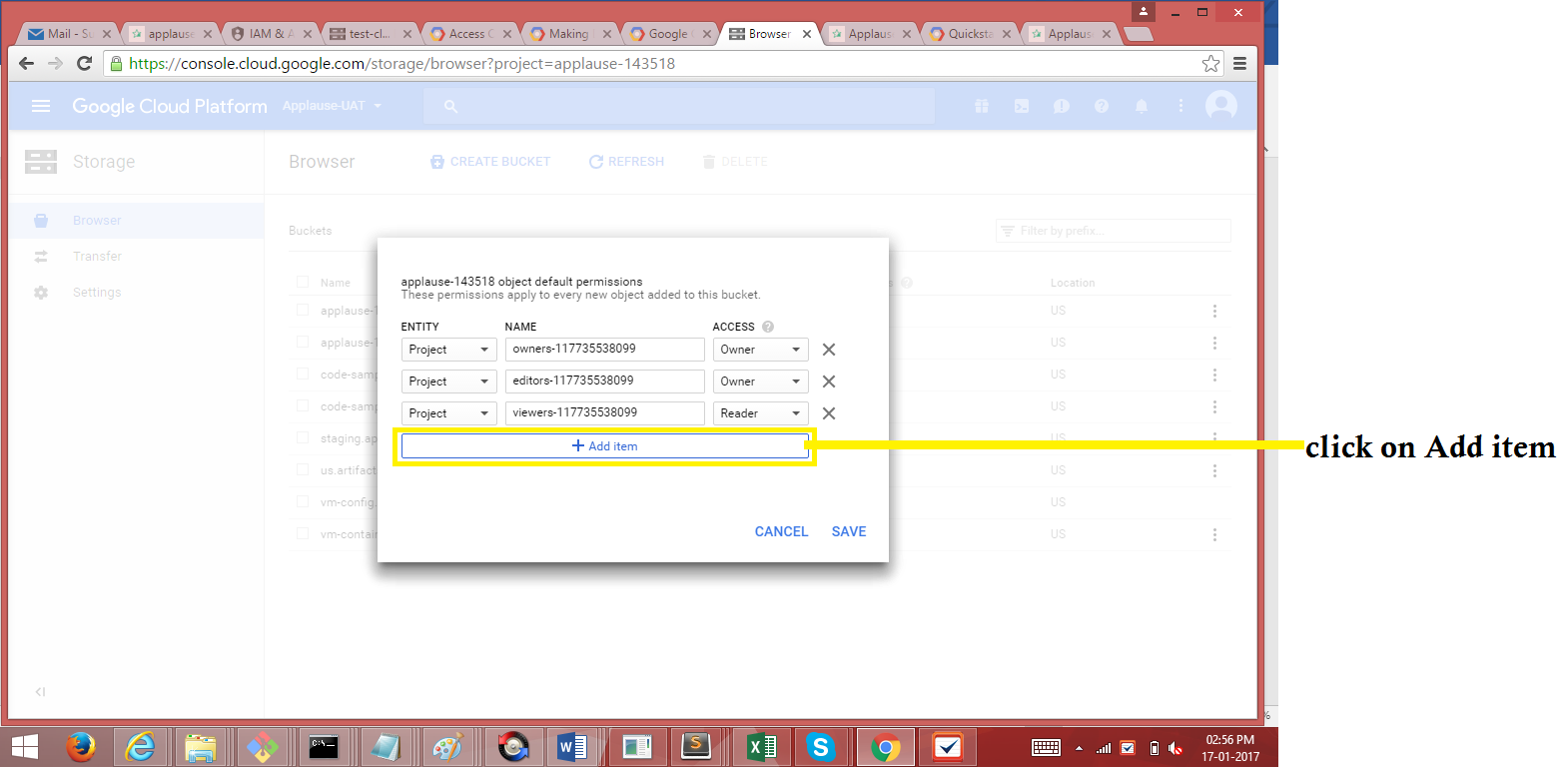
* Bucket created successfully

**10.2 Set Permission for storage bucket (first click on browse)**

* Click on browse option in storage.
* Select object default permission of your bucket.

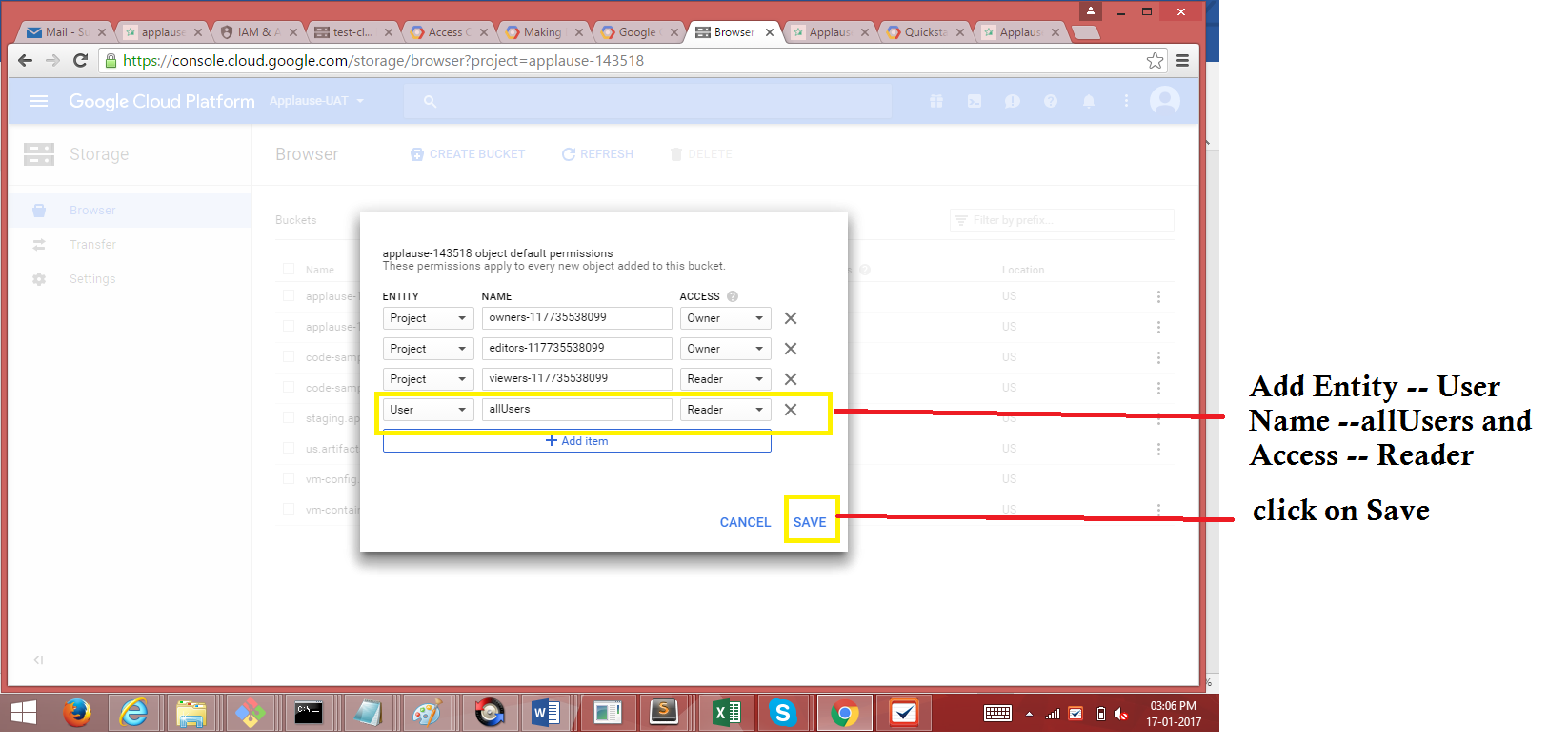


* Click on Add item



* Select **Entity** -- User, **Name** – allUsers and **Access** – Reader

And click on Save



* Click on Save.

**10.3 Store Images to Bucket**

* Open google cloud sdk and navigate to the path where you have brand\_img , loc\_img and emp\_img and template directories with images. (present in applause\_backend\_portal/docs/google\_deployment/images-empty-db) select the appropriate folder based on the database restored in previous steps.
* Run gcloud init and Select your gmail account of Project to which you want to deploy (not required if already done)
* Run gsutil -m cp -R . gs://CLOUD\_BUCKET\_NAME

Dot here

**Edit the following file**

* In applause\_backend\_portal/server/deploy/[dev,prod,uat]/ folder change js\conf\v1\config-dev.js
* Chnage following object –

const : {

CMS\_HEADER\_NAME : "gvb",

E\_SIG\_HEADER\_NAME : "gvb",

CLOUD\_BUCKET : "CLOUD\_BUCKET\_NAME",

projectId : "project-id",

CLOUD\_IMAGE\_URL : "http://storage.googleapis.com/CLOUD\_BUCKET\_NAME/",

SKIPPABLE\_PATH\_PARTS : 4

}

## **11.Deploy the Application**

This section explains list of changes need to be done in order deploy or publish application using publish.bat file

By using this batch file we can publish the application to dev/prod/uat

The batch file path = applause\_backend\_portal/server/publish.bat

Note: Make changes in the environments section which needs to be deployed

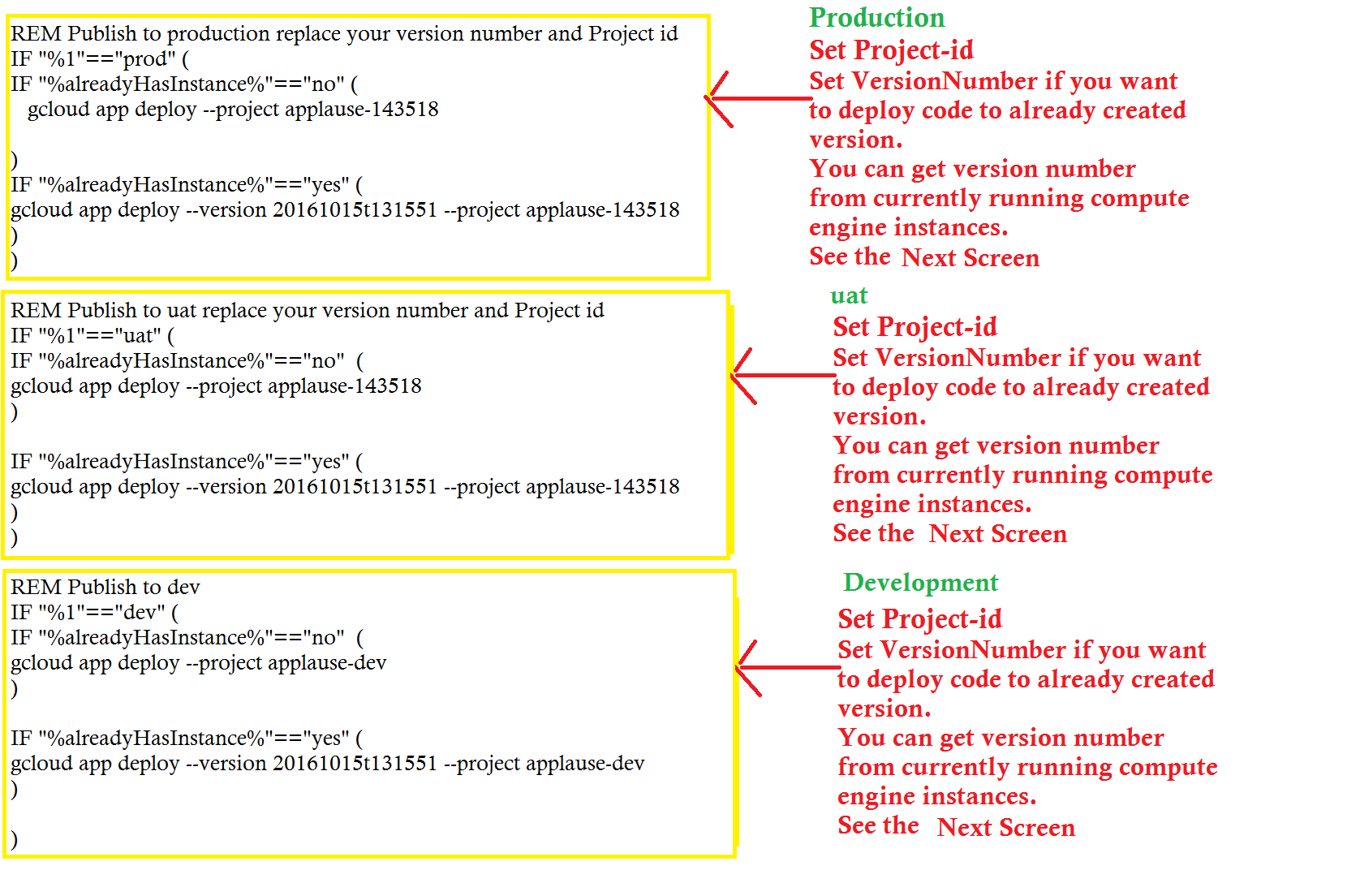
* Fresh Deploy (New app engine instance)

When the application is deployed first time then change the following line alreadyHasInstance to “no”. Default is yes

set alreadyHasInstance=yes to set alreadyHasInstance=no

There are three sections of this file. Publish to uat, prod, dev

Change the values of --project argument to your project id



* Re Deploy (to already created app engine. With version number)

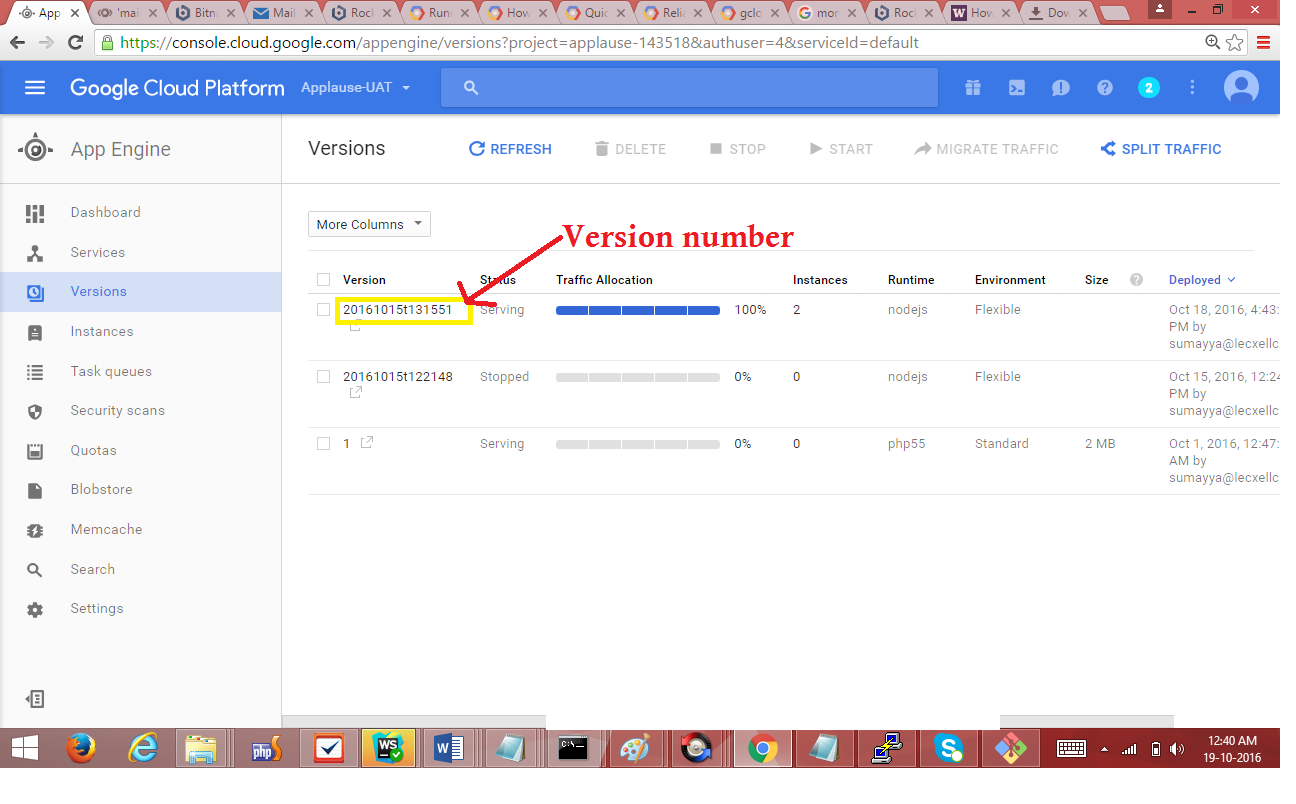
When the application is redeployed then change the following line alreadyHasInstance to “yes”.

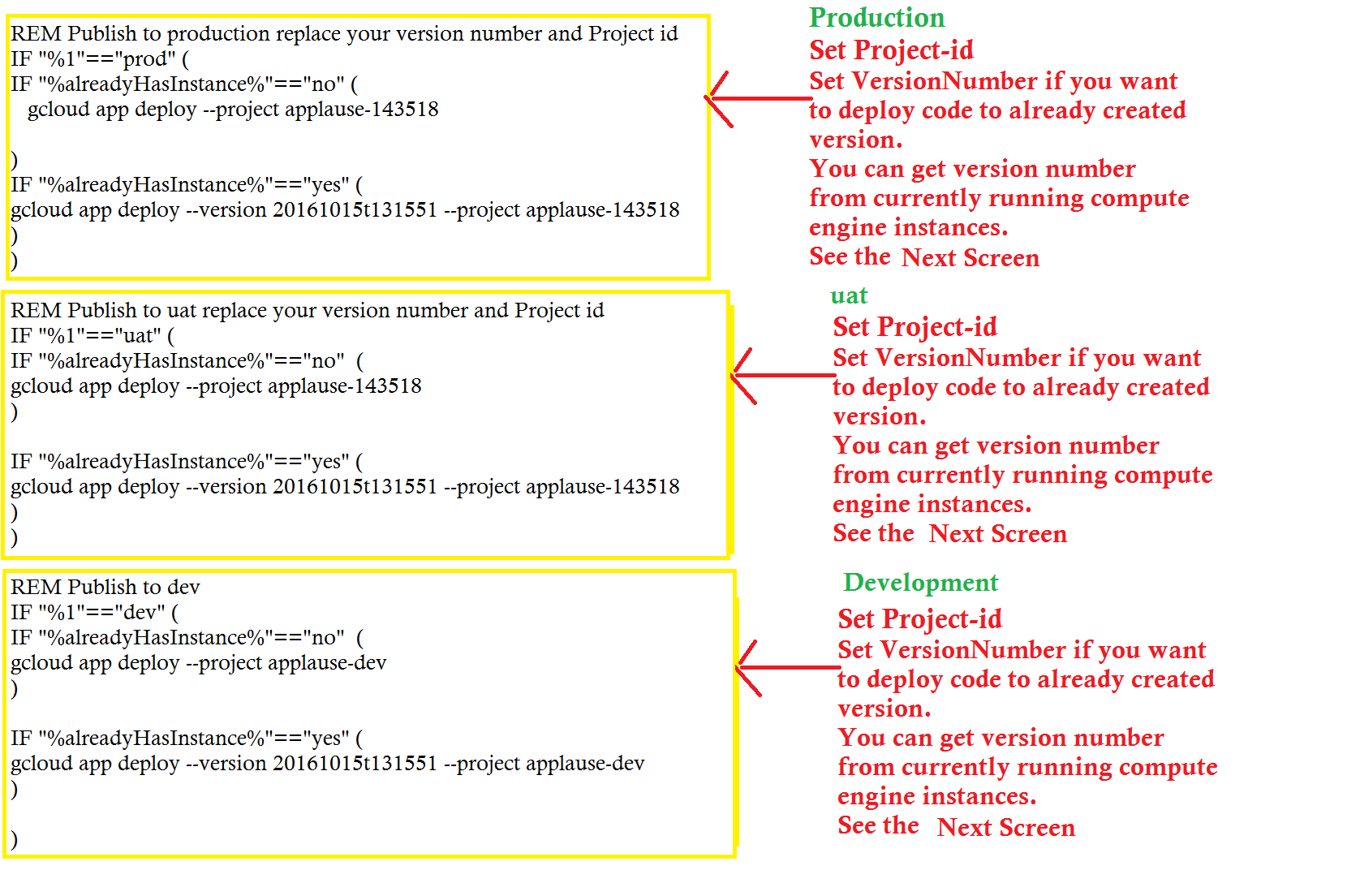
set alreadyHasInstance=no to set alreadyHasInstance=yes

There are three sections of this file. Publish to uat, prod, dev

Change the values of --project and --version arguments to your project id and version number. (See the following screenshot to get version number)

Goto AppEngine -> Versions (from google cloud console hamburger icon)





* Run “publish.bat uat/dev/prod” in the folder applause\_backend\_portal/server (from command prompt/ terminal)
* This will deploy the server with sample db which contains superadmin user with username = [admin@meimodo.com](mailto:admin@meimodo.com) & password = password

## **12. Backup of Images from cloud storage to local**

* Run gcloud init command and select you project with cloud storage bucket
* Run gsutil -m cp -R gs://CLOUD\_BUCKET Path
* Path is destination directory path