# **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)

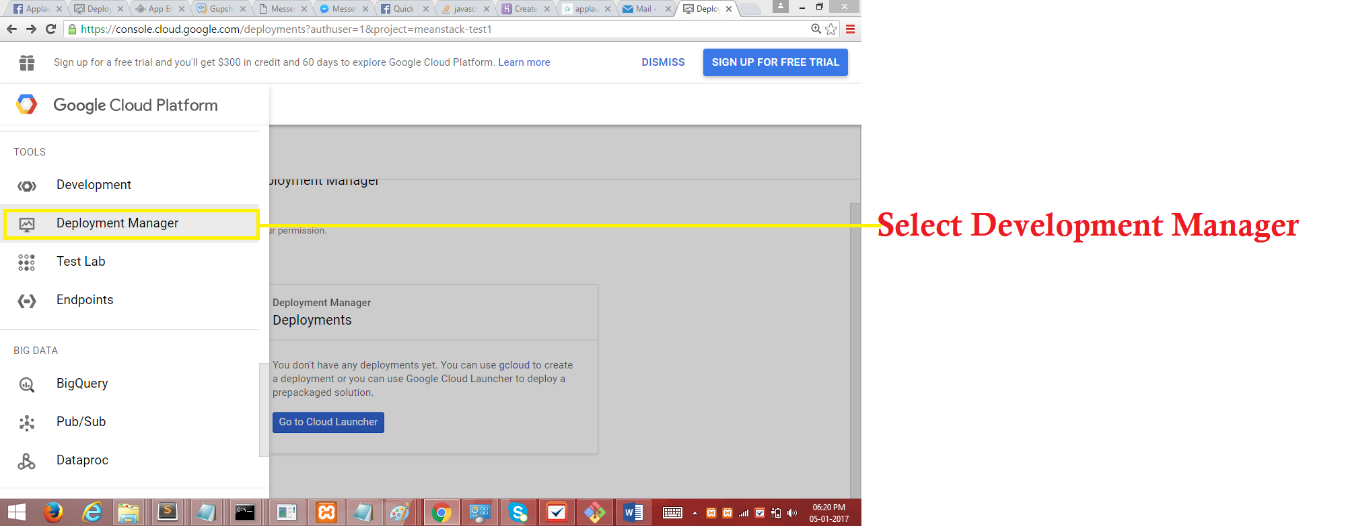
## **1. Installation and Setup**

* [Install](https://dl.google.com/dl/cloudsdk/channels/rapid/GoogleCloudSDKInstaller.exe) google cloud sdk.
* Run gcloud init and Select your gmail account of Project to which you want to deploy

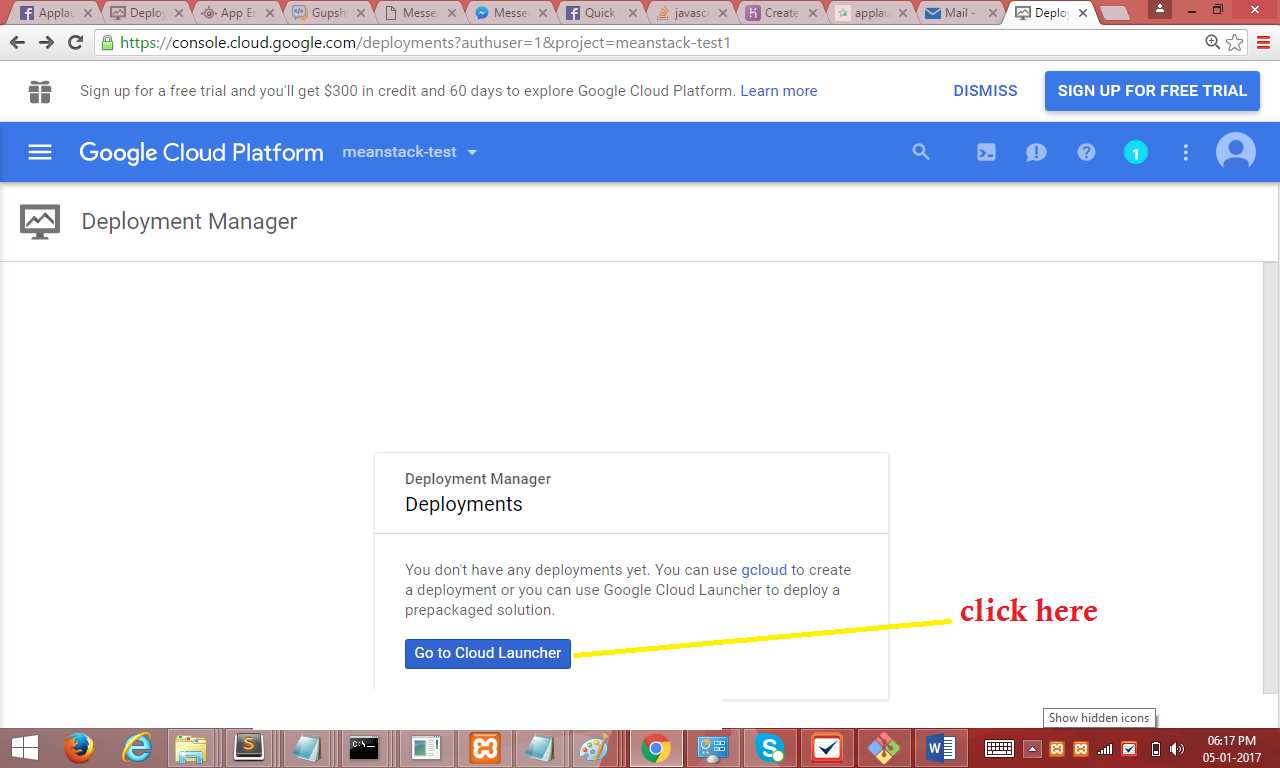
## **2. Create Mongodb Instance**

This section explains how to create Mongodb Mean instance.

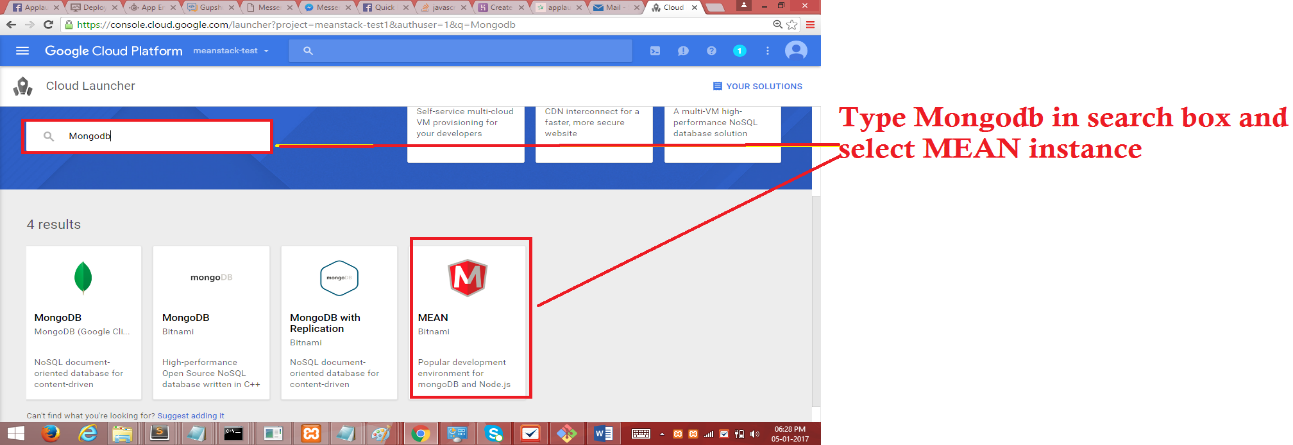
* Goto <https://console.cloud.google.com>
* Select Deployment Manager



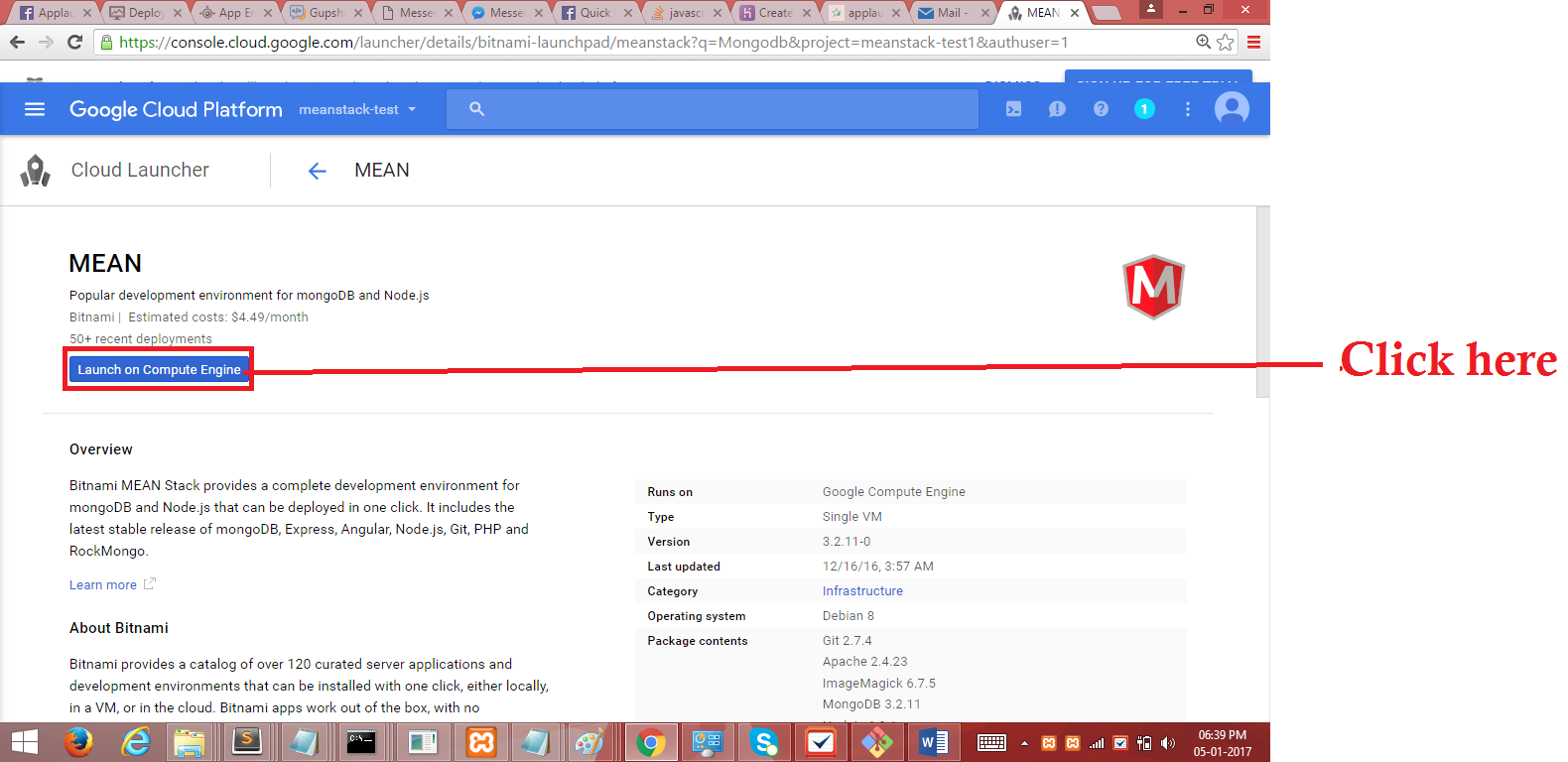
* Click on Go to Cloud Launcher



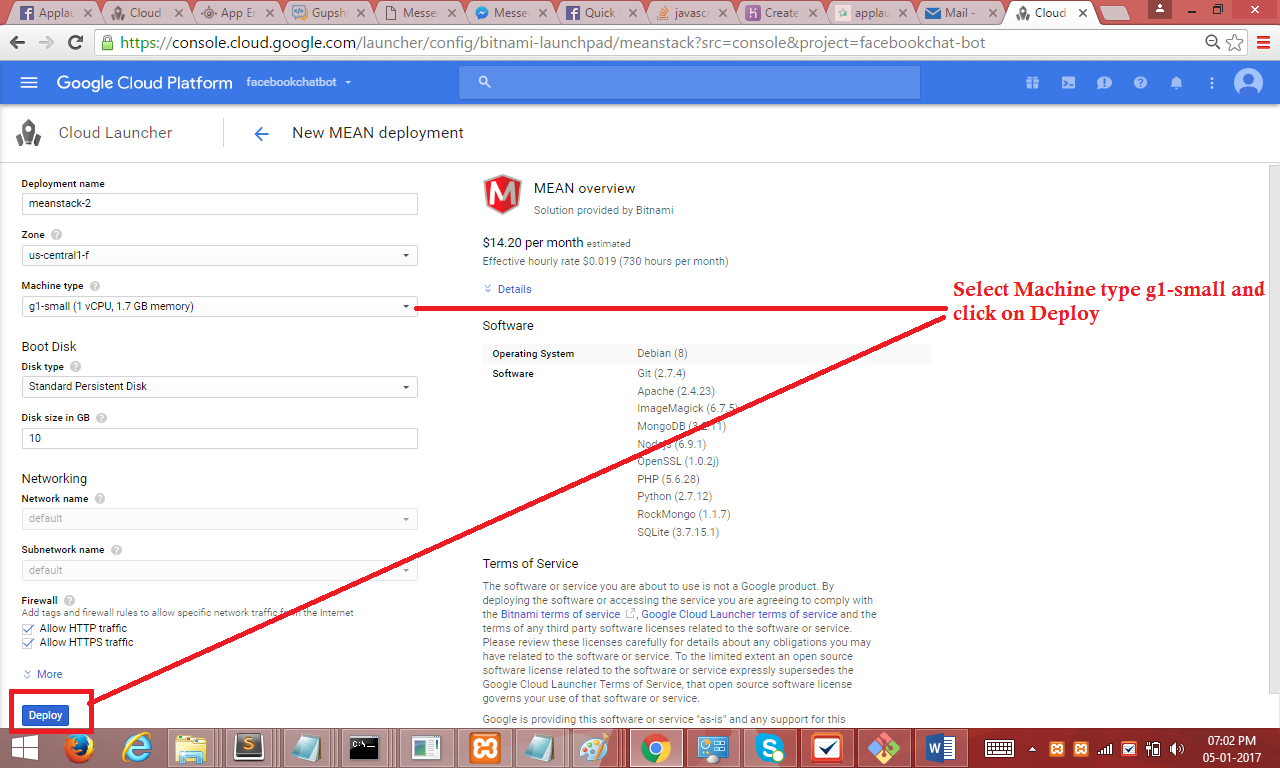
* Select Mean Instance.



* Click on Launch on Compute Engine



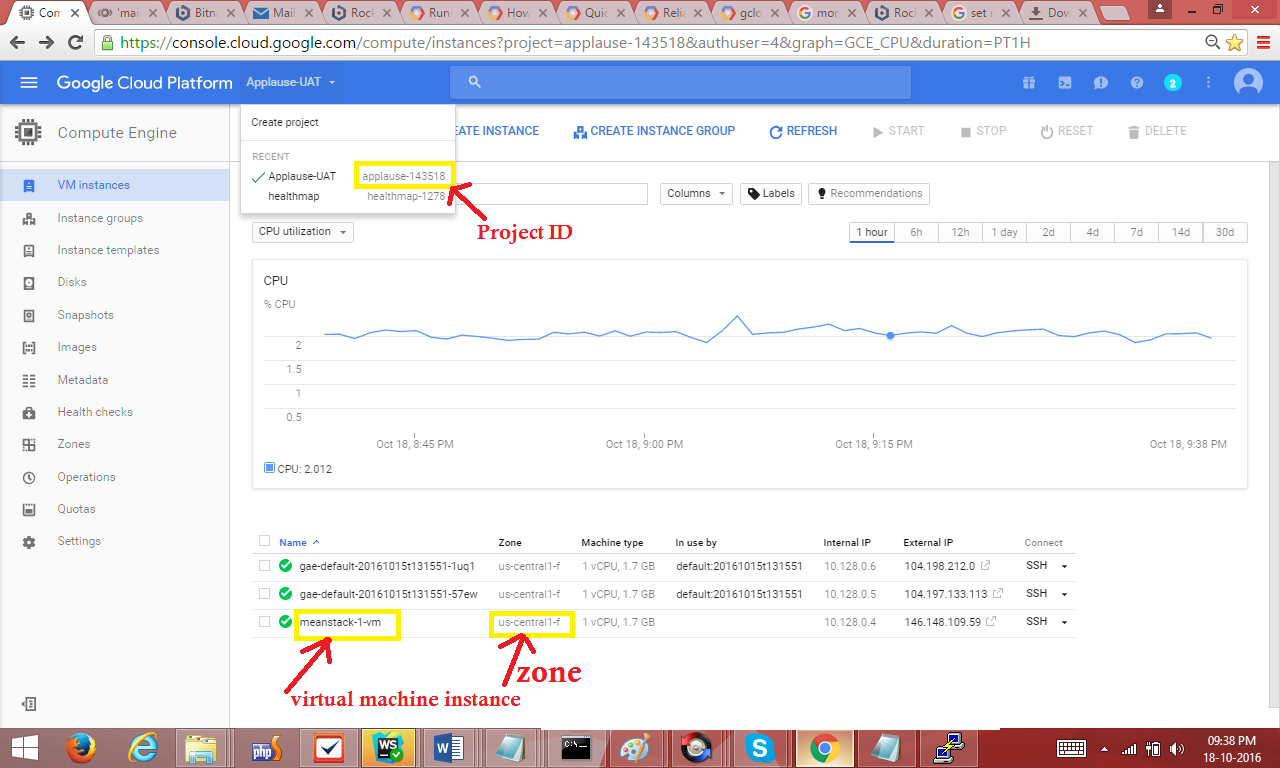
* Select Machnie Type g1-snall and click on Deploy



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

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

* Goto compute engine



* Run this command in google cloud sdk

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

After Virtual Machine started you can Export and import database.

## **4. Export applause database.**

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]**

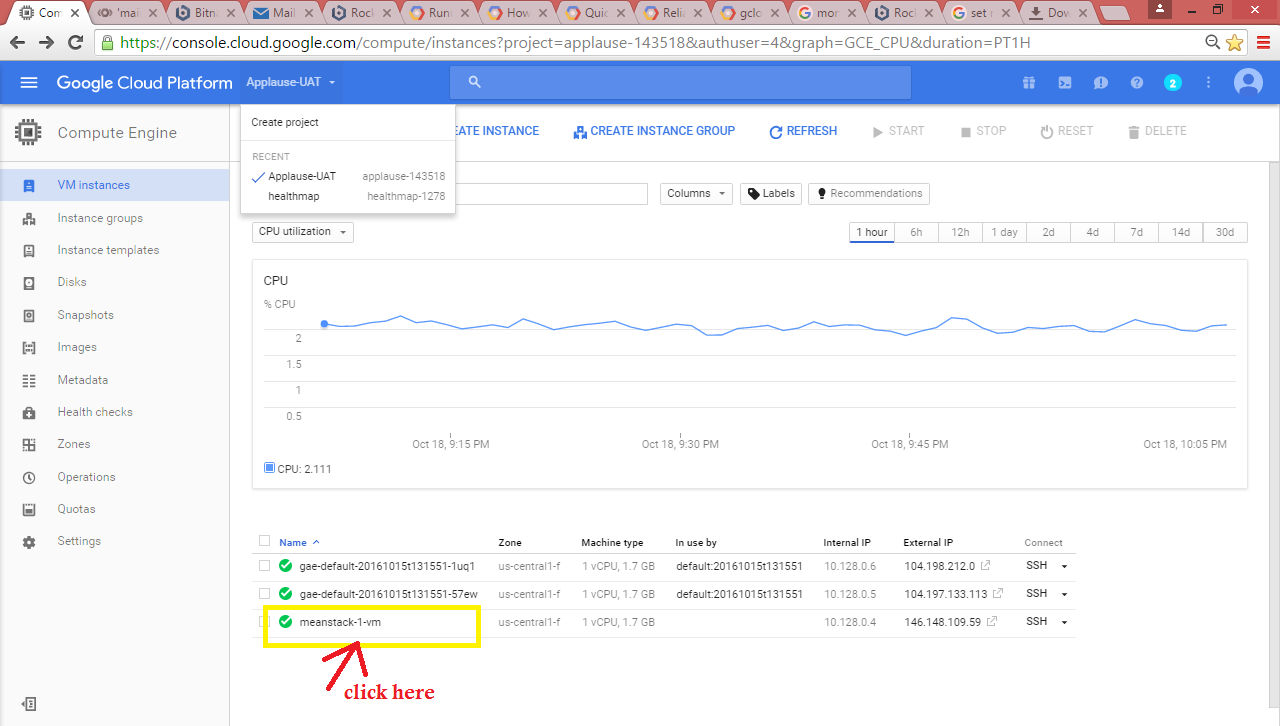
* gcloud compute copy-files "Sumayya@meanstack-1-vm:dump" "./"
* This command copy files from virtual machine to current directory (./).
* You can copy dump folder from VM to any directory by specify the path instead of “./” like “C:\applause”

path to dump is path to virtual machine dump folder

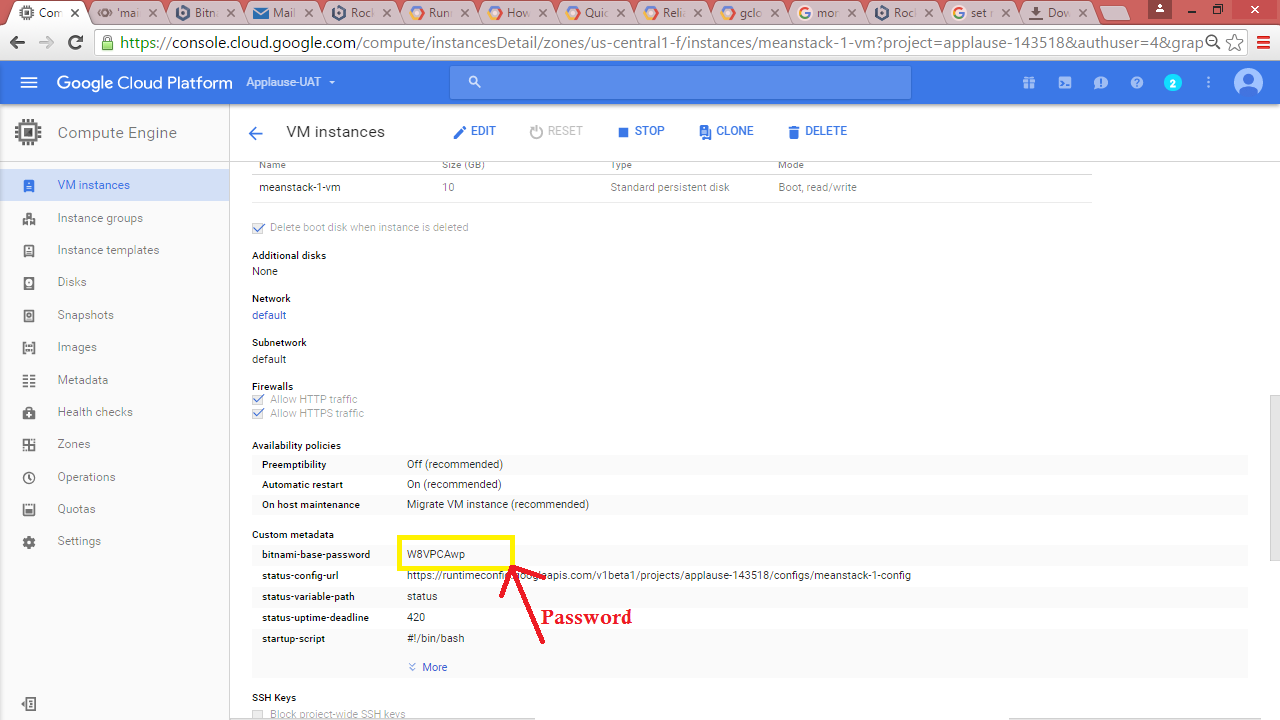
## **5. Import applause database**

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.
* Goto Compute Engine

****

* Copy binami password



* Create a dump directory in Virtual machine

**Copy files from local machine to VM**

* Create a directory called dump in VM
* gcloud compute copy-files "C:\applause\dump\applause" "Sumayya@meanstack-1-vm:dump"
* Run the command

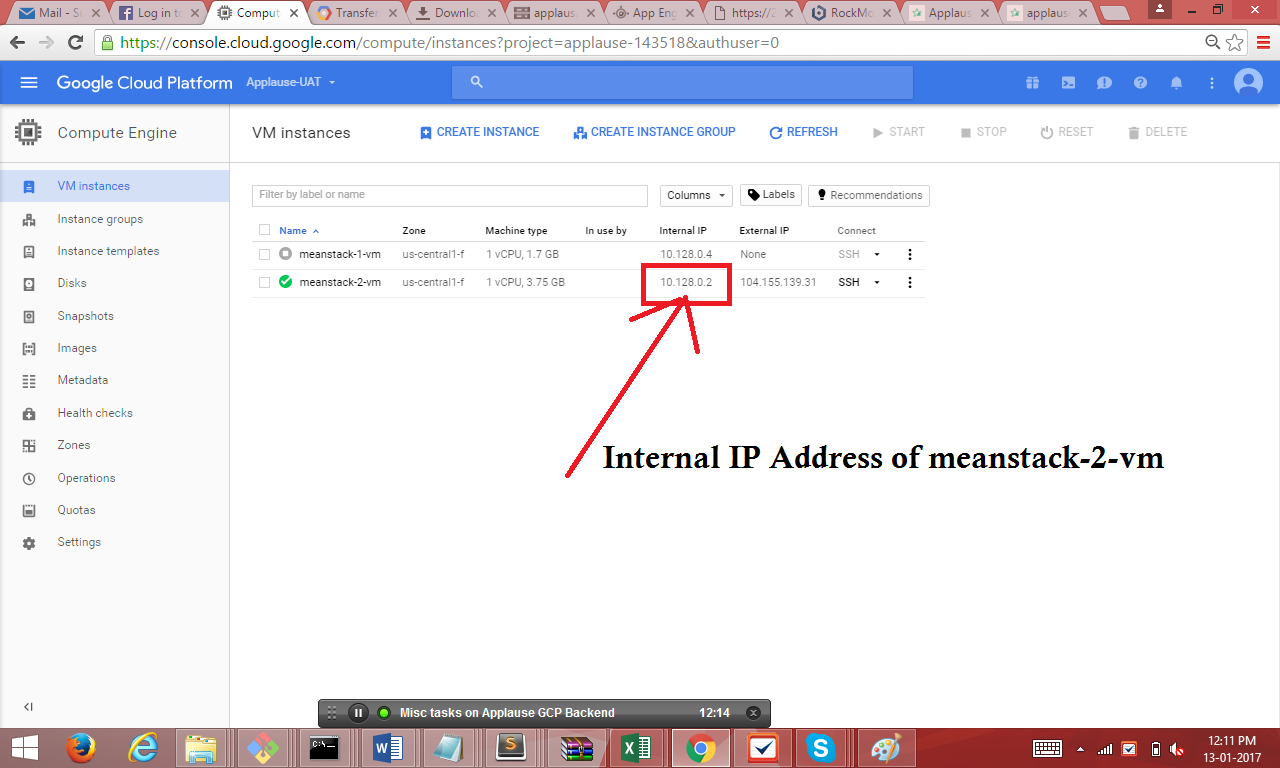
mongorestore --port 27017 --host meanstack-1-vm -u root -p password -d applause Path To Applause

* Path to Applause for example 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

cd /opt/bitnami/mongodb

* sudo chmod 726 mongodb.conf

vi mongodb.conf

* set bind\_ip = 127.0.0.1,Internal IP Address of vm
* Save mongodb,conf file and quit from vi editor
* sudo /opt/bitnami/ctlscript.sh restart

## 

## **7. Change Admin Password Create New User and set Permissions to Applause database**

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**

setParameter = enableLocalhostAuthBypass=0

with:

#setParameter = enableLocalhostAuthBypass=0

**Change Admin Passsword**

* mongo admin --username root --password bitnami-password
* change admin password

db = db.getSiblingDB('admin')

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

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**:

#setParameter = enableLocalhostAuthBypass=0

with:

setParameter = enableLocalhostAuthBypass=0

**Restart the MongoDB server.**

/opt/bitnami/ctlscript.sh restart

## **8. Setting Mongodb GUI : Rockmongo**

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

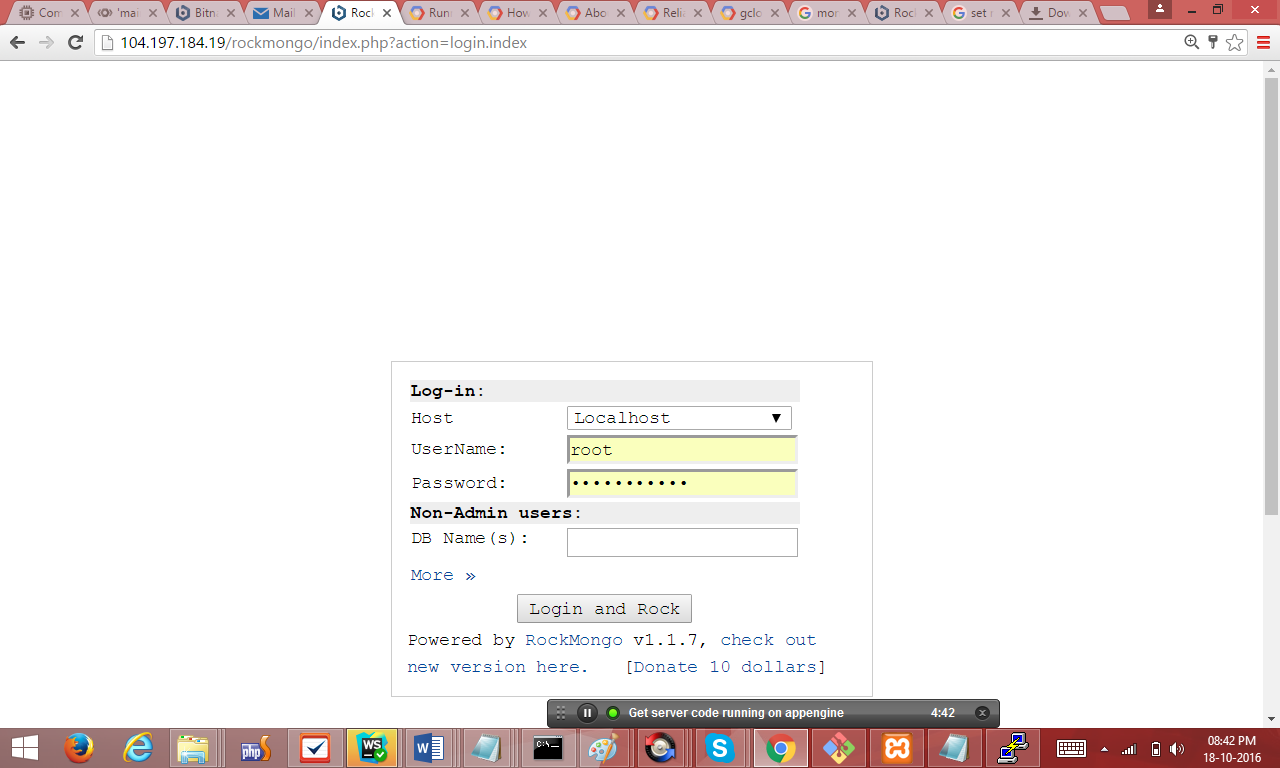
1. <IfVersion >= 2.3>
2. Require all granted
3. </IfVersion>

* Save the file

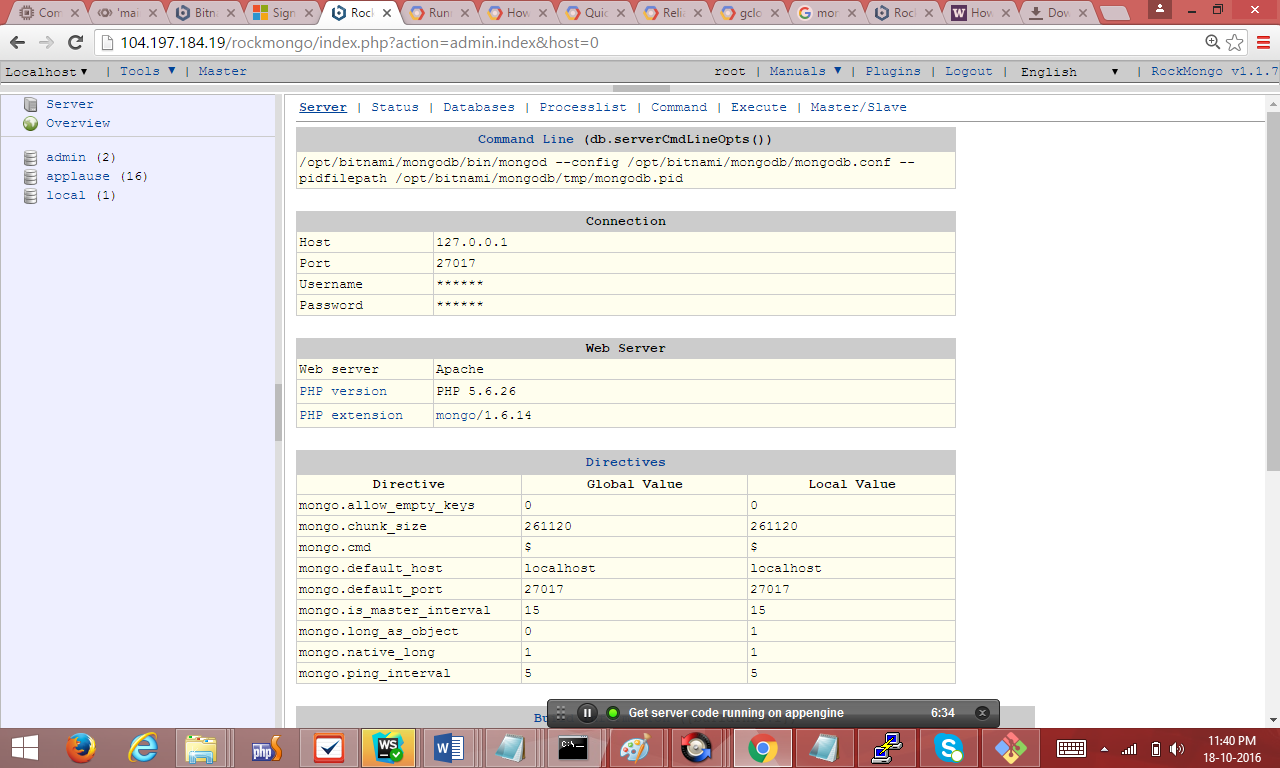
sudo /opt/bitnami/ctlscript.sh restart

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**

This section explains how to edit database details in application.

change config-dev.js file under deploy directory

database : {

host : 'internal IP address of VM',

user : 'database\_user',

password : 'database\_password',

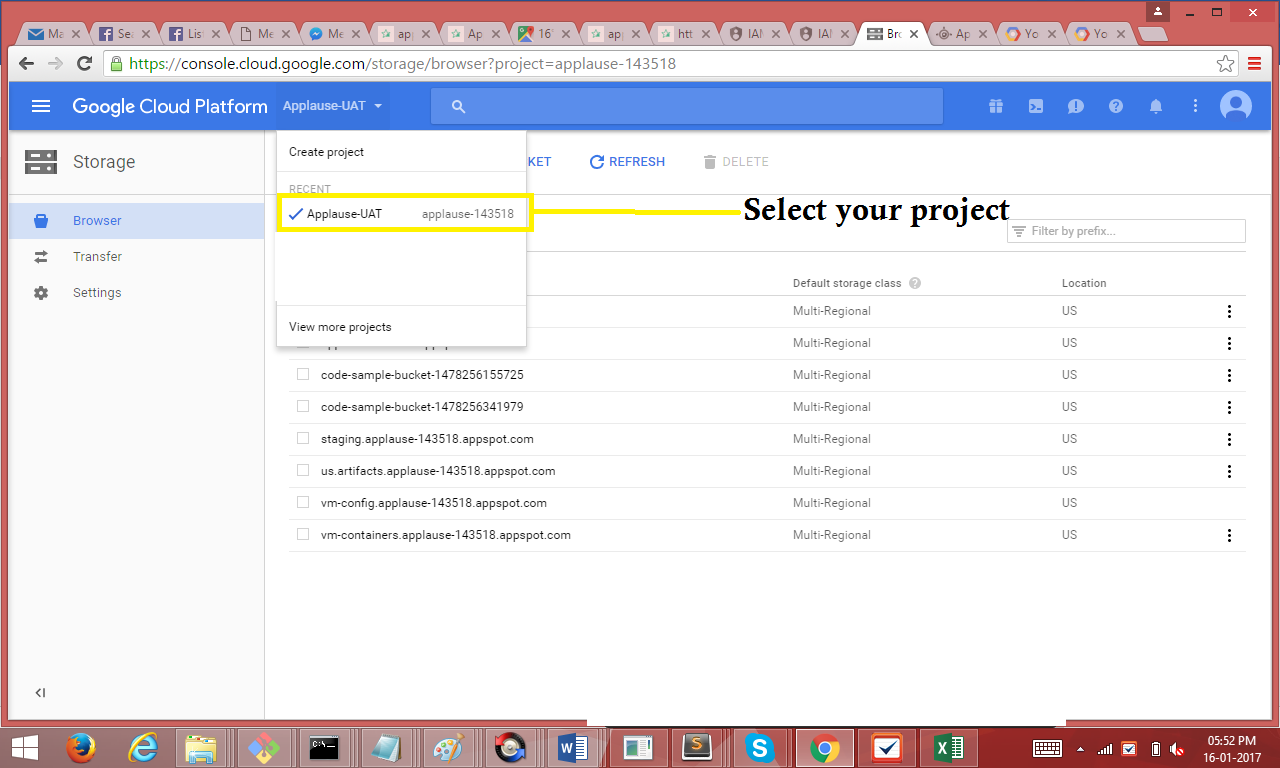
database : 'applause' }

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

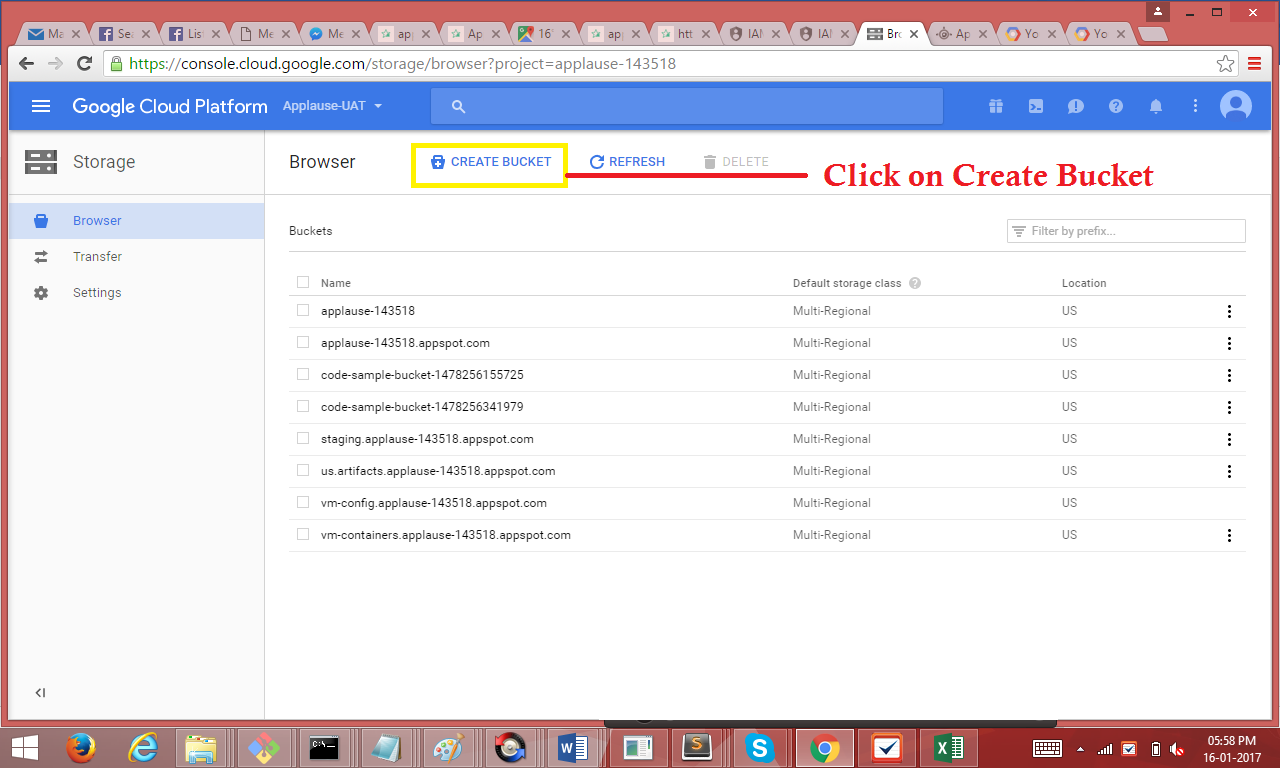
This section explains how to create, setting permissions and upload files to google cloud storage bucket for storing and accessing images from bucket in application

**10.1 Create Google Storage bucket**

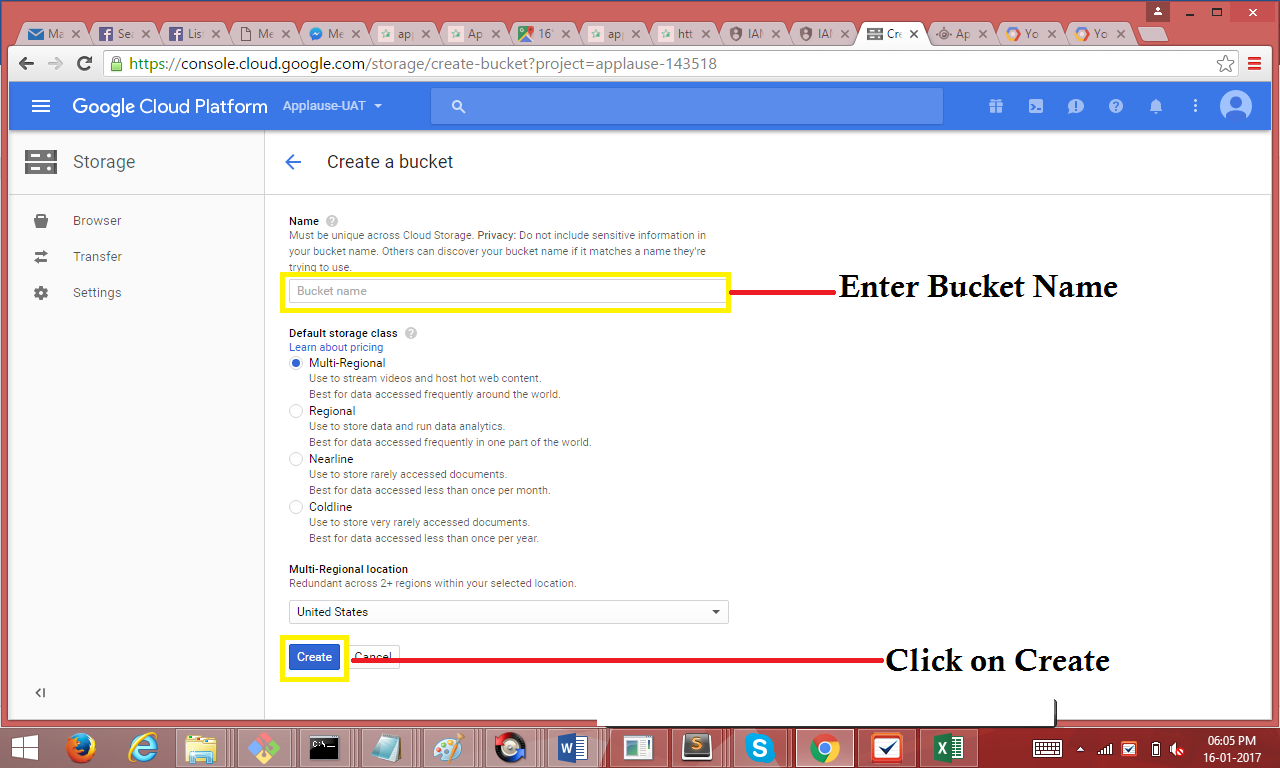
* Goto url <https://console.cloud.google.com/storage>
* Select you Project



* Click on Create Bucket



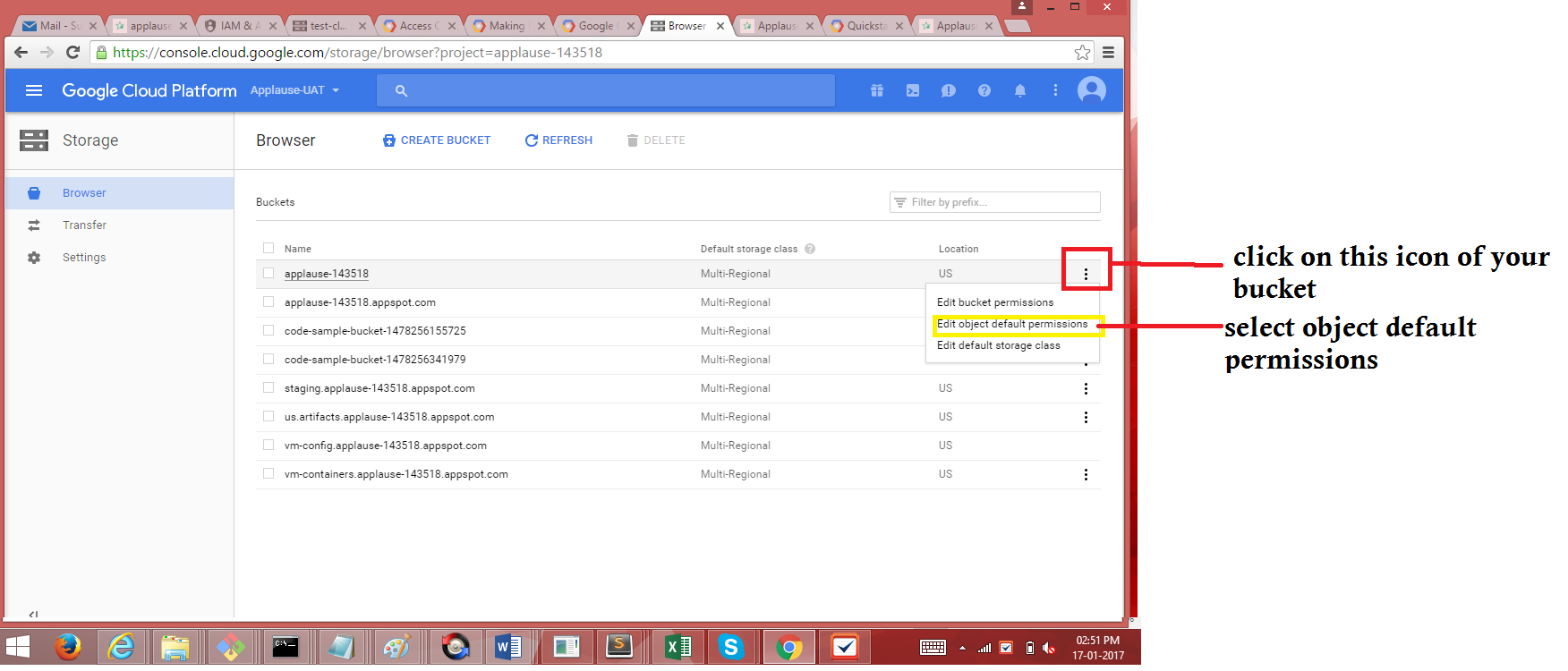
* Enter Bucket name and Click on Create



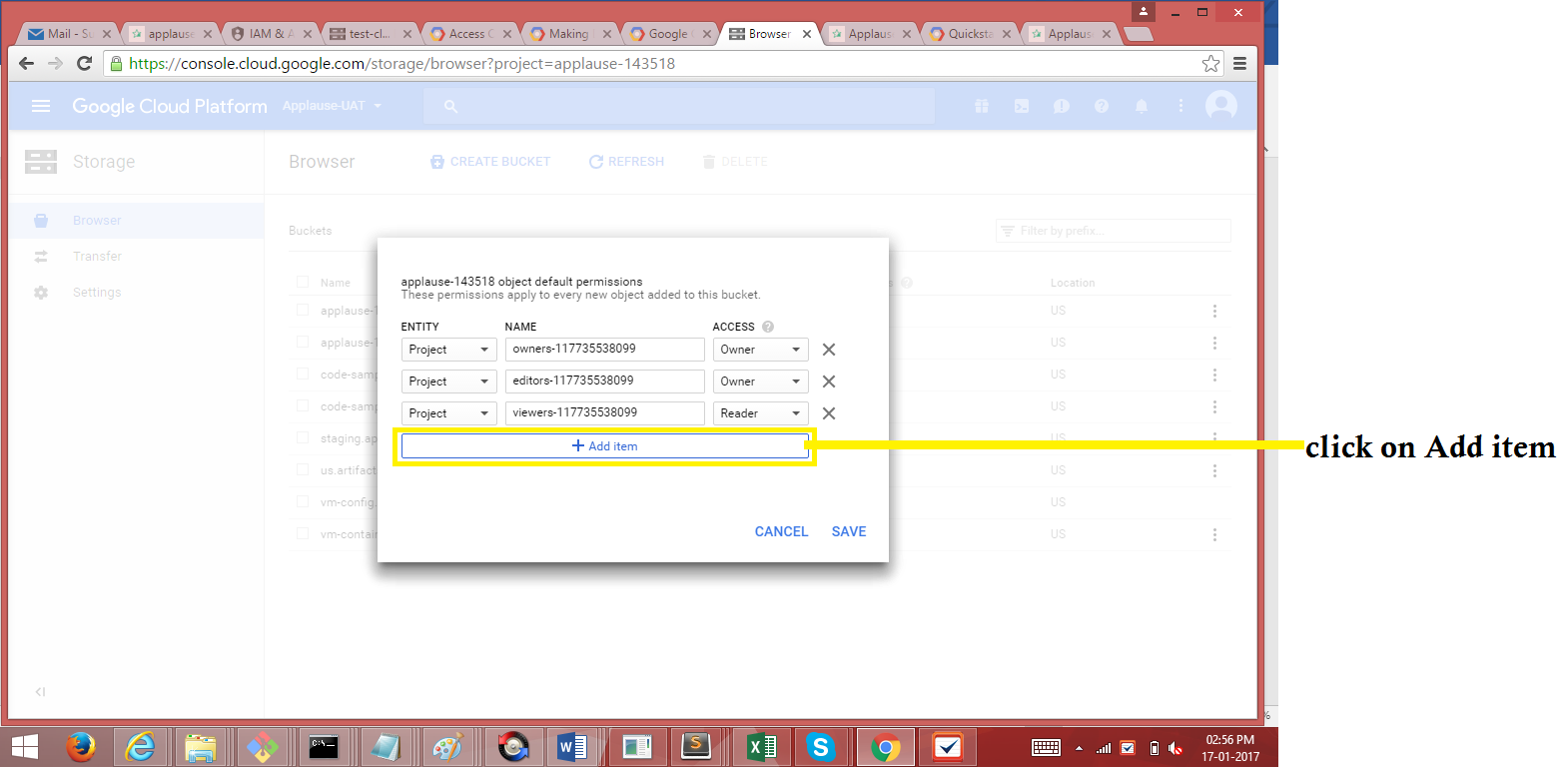
* Bucket created successfully

**10.2 Set Permission for storage bucket**

* Select object default permission of your bucket.

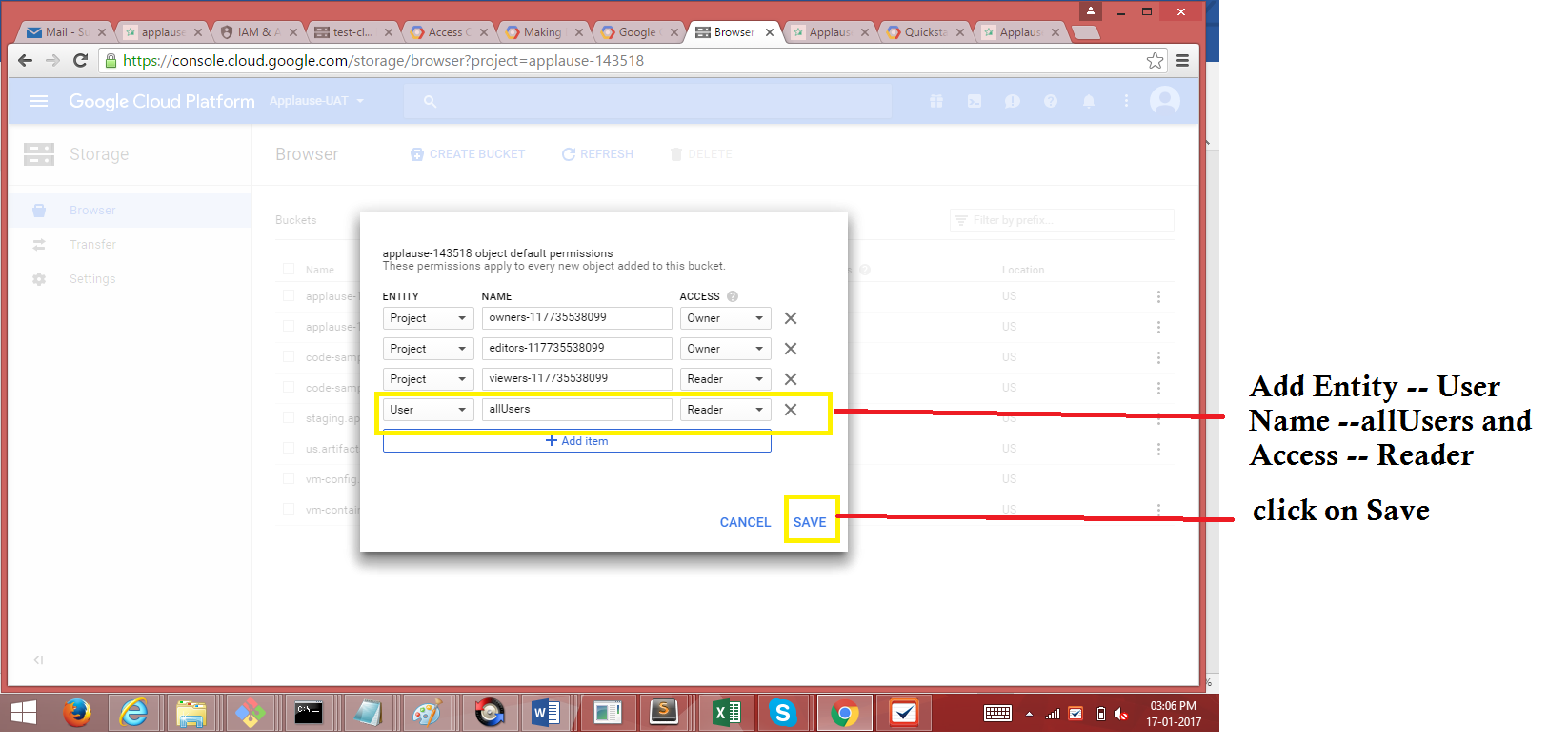


* Click on Add item



* 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.
* Run gcloud init and Select your gmail account of Project to which you want to deploy
* Run gsutil -m cp -R . gs://CLOUD\_BUCKET

Dot here

**Edit the following 2 files**

* In deploy folder change js\conf\v1\config-dev.js line numbers 37 and 38 change the constants CLOUD\_BUCKET , projectId.
* In deploy folder change public\js\main.js line number 78 change ASSET\_URI

.constant('ASSET\_URI', "http://storage.googleapis.com/ CLOUD\_BUCKET/")

* CLOUD\_BUCKET is bucket name that you created which contains brand\_img,

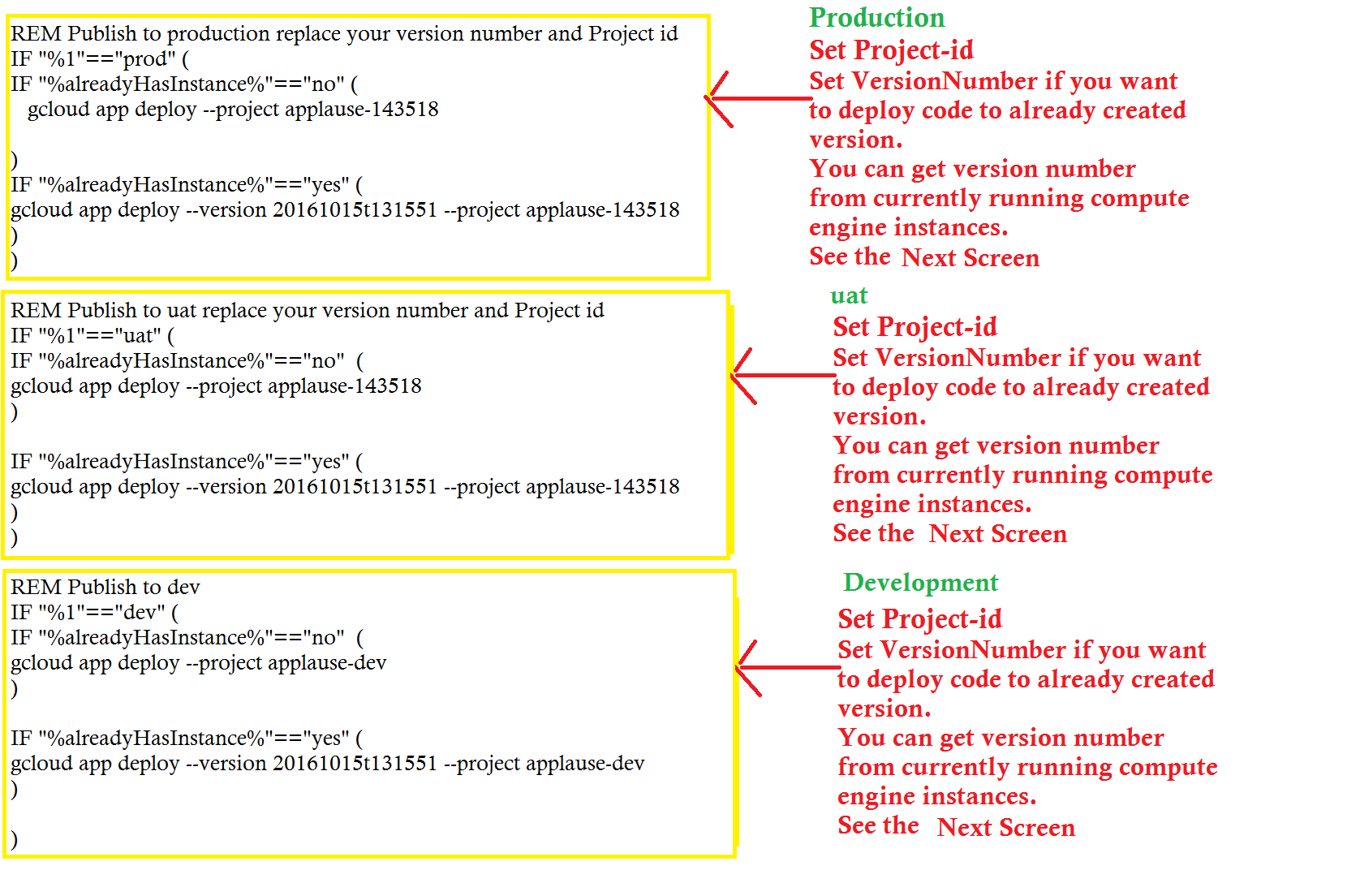
## **11.Deploy the Application**

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

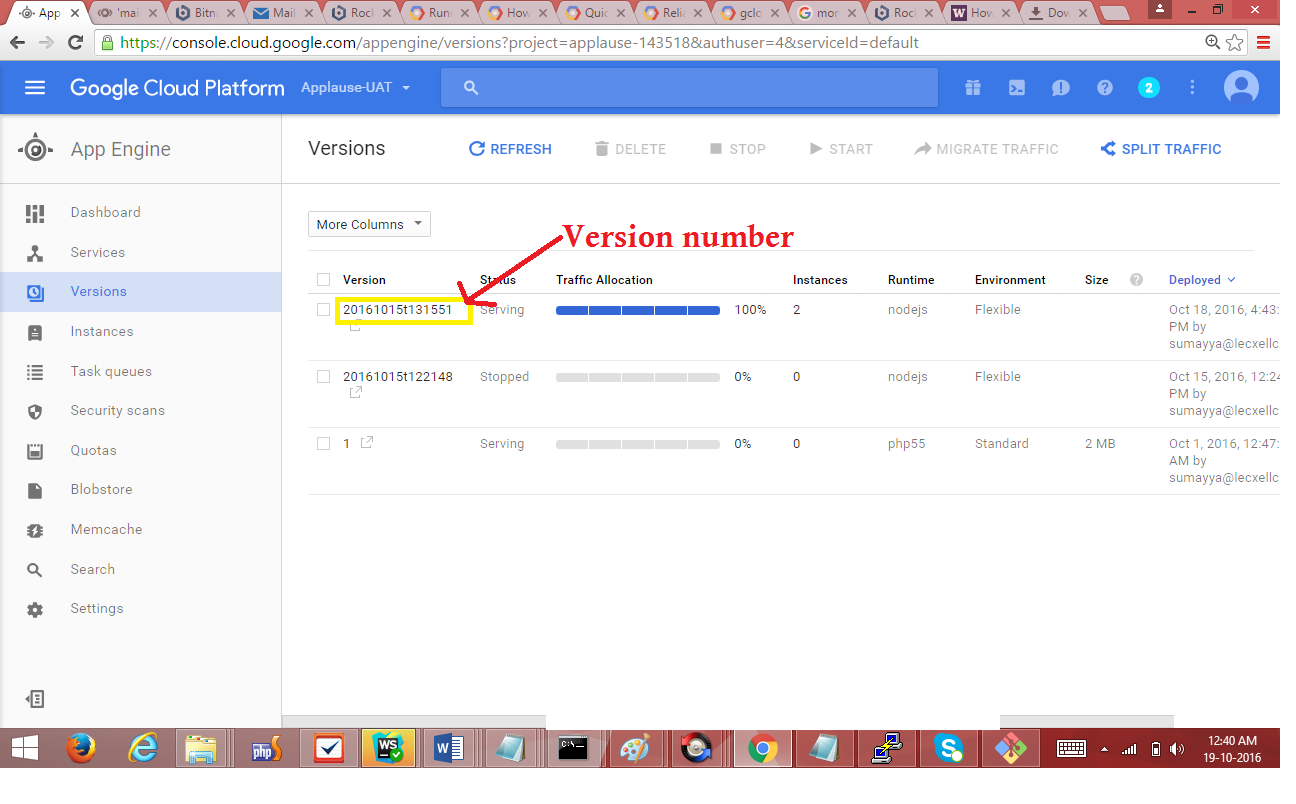
By using this batch file we can publish the application to dev/prod/uat

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

* set alreadyHasInstance=no



Goto AppEngine -> Versions



* Cd C:\applause\default\server\public
* Run npm install
* Run publish.bat uat/dev/prod

## **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