

**PDEA's**  
**COLLEGE OF ENGINEERING MANJARI(BK.),**  
**PUNE**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**LAB MANUAL**

**314458: Laboratory Practice-II**  
**(Cloud Computing)**

**Year / Semester : IV / VII**

**2022-23**

**PREPARED BY**

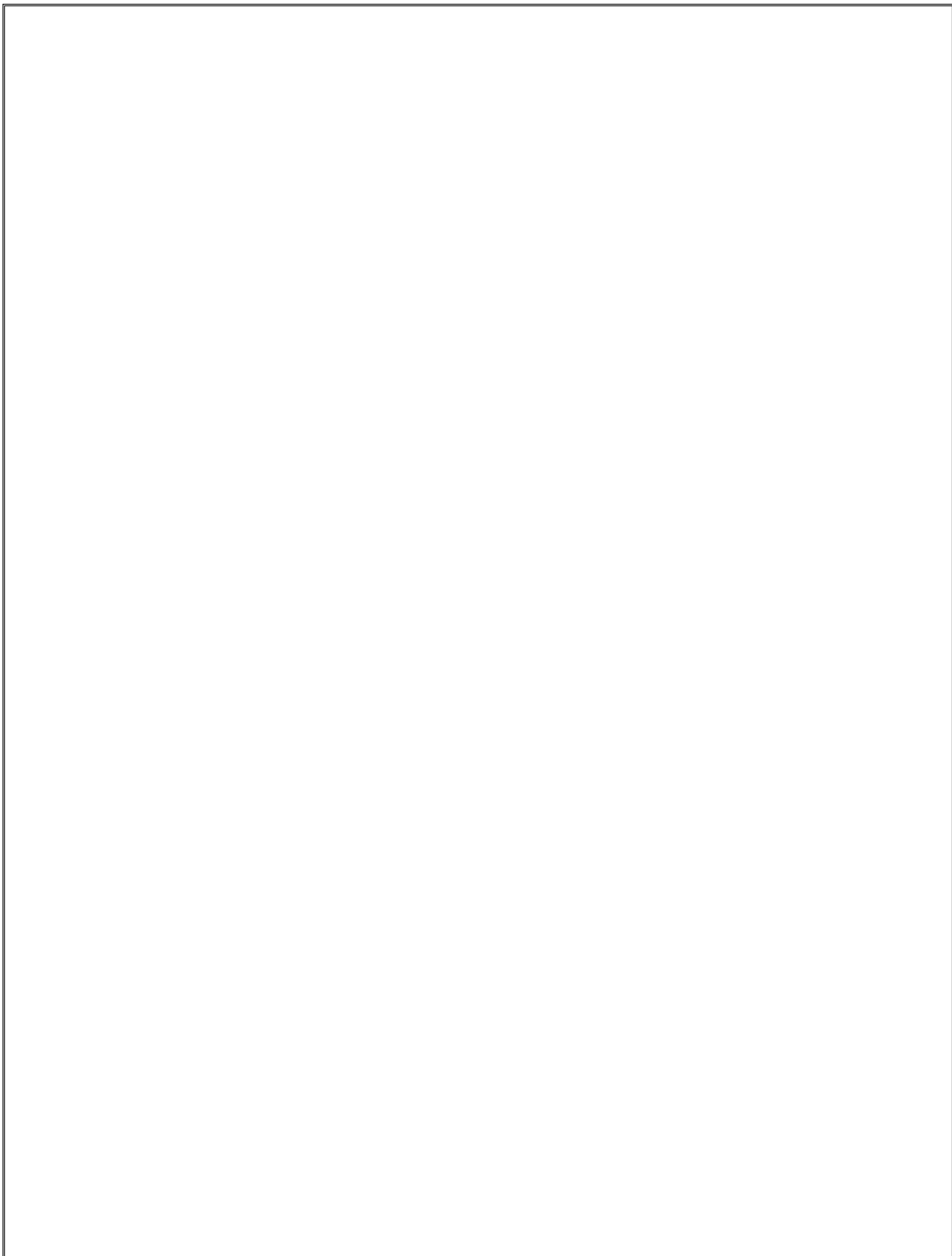
**Dr. Deepali S Hirolikar**

## **LIST OF EXPERIMENTS**

1. Install Google App Engine. Create *hello world* app and other simple web applications using python/java.
2. Use GAE launcher to launch the web applications.
3. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
4. Find a procedure to transfer the files from one virtual machine to another virtual machine.
5. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)

## TABLE OF CONTENTS

S.NO.	DATE	EXPERIMENT TITLE	MARKS /10	SIGN.
1.		Install Google App Engine. Create <i>hello world</i> app and other simple web applications using python/java.		
2		Use GAE launcher to launch the web applications.		
3.		Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.		
4.		Find a procedure to transfer the files from one virtual machine to another virtual machine.		
5 •		Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)		



EX NO.:1

DATE:

## Install Google App Engine. Create *hello world* app and other simple web applications using python/java.

**Aim:** To Install Google App Engine. Create *hello world* app and other simple web applications using python/java.

### Procedure:

#### 1. Install Google Plugin for Eclipse

Read this guide – [how to install Google Plugin for Eclipse](#). If you install the Google App Engine Java SDK together with “Google Plugin for Eclipse“, then go to step 2, Otherwise, get the [GoogleApp Engine Java SDK](#) and extract it.

#### 2. Create New Web Application Project

In Eclipse toolbar, click on the Google icon, and select “New Web Application Project...”

Figure – New Web Application Project

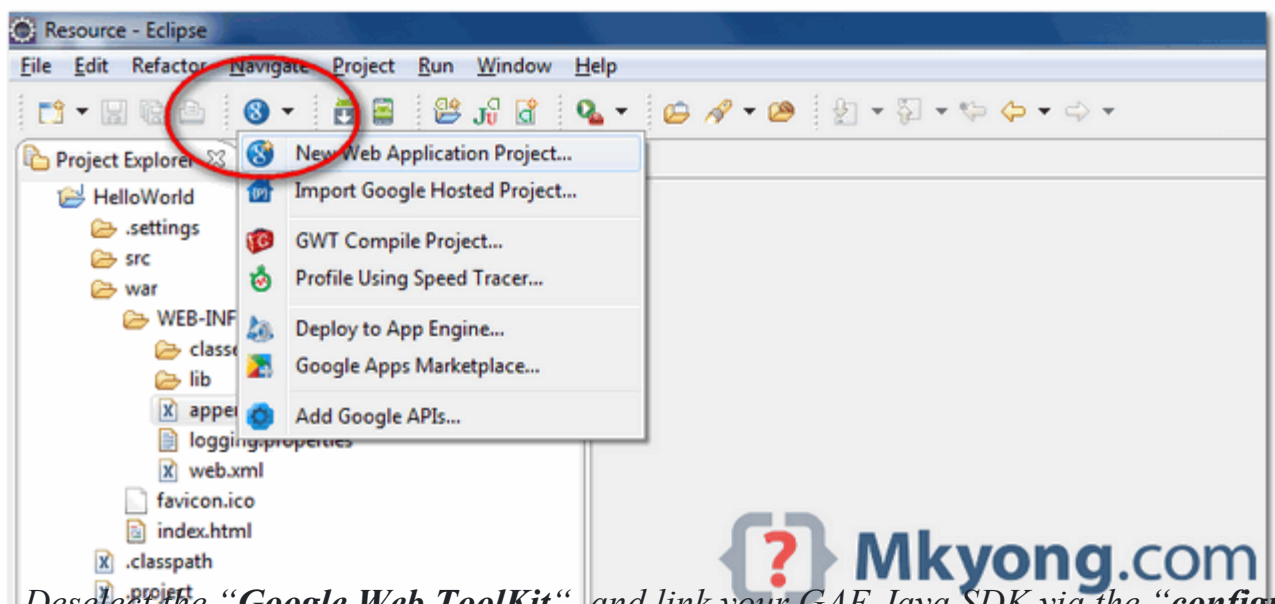
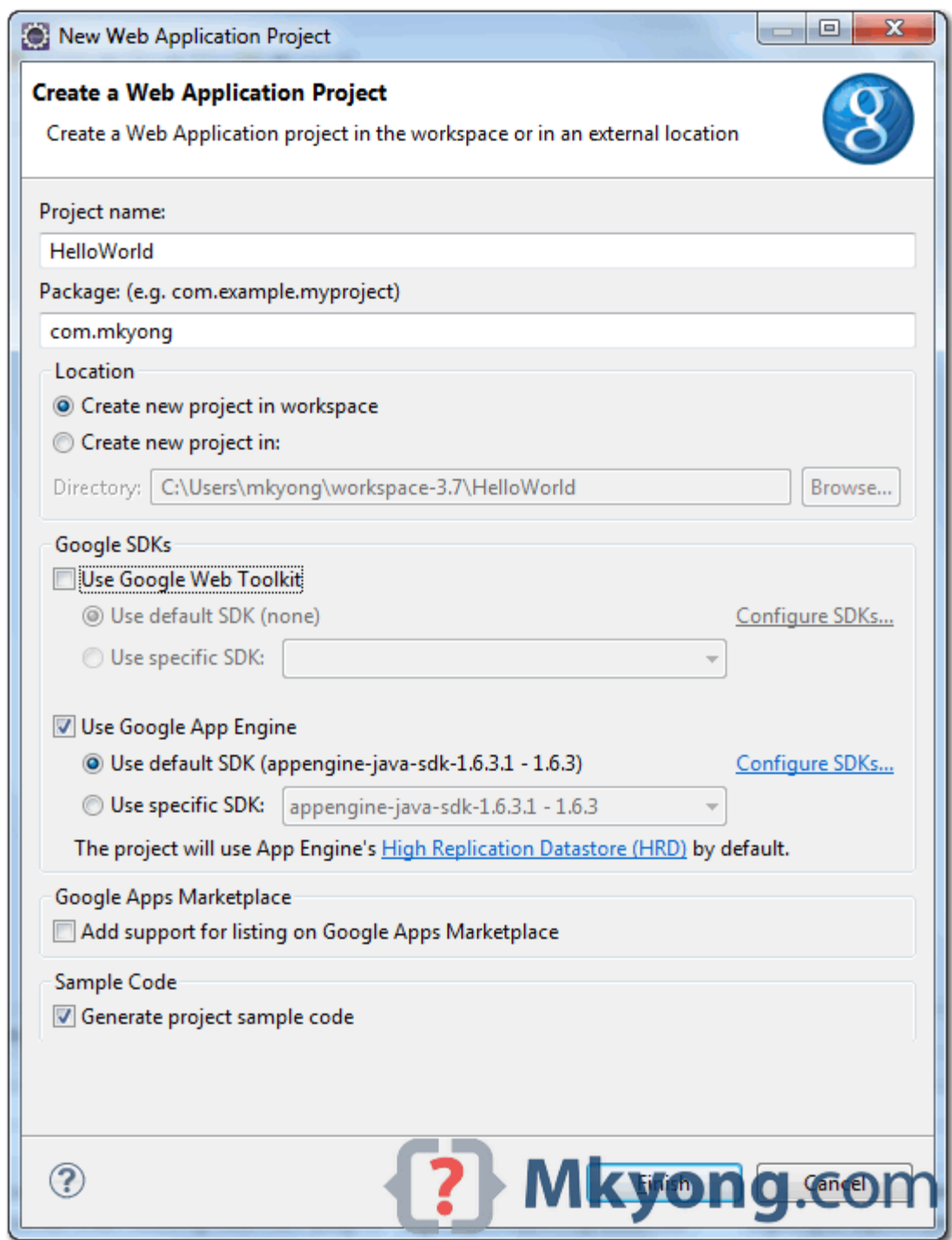


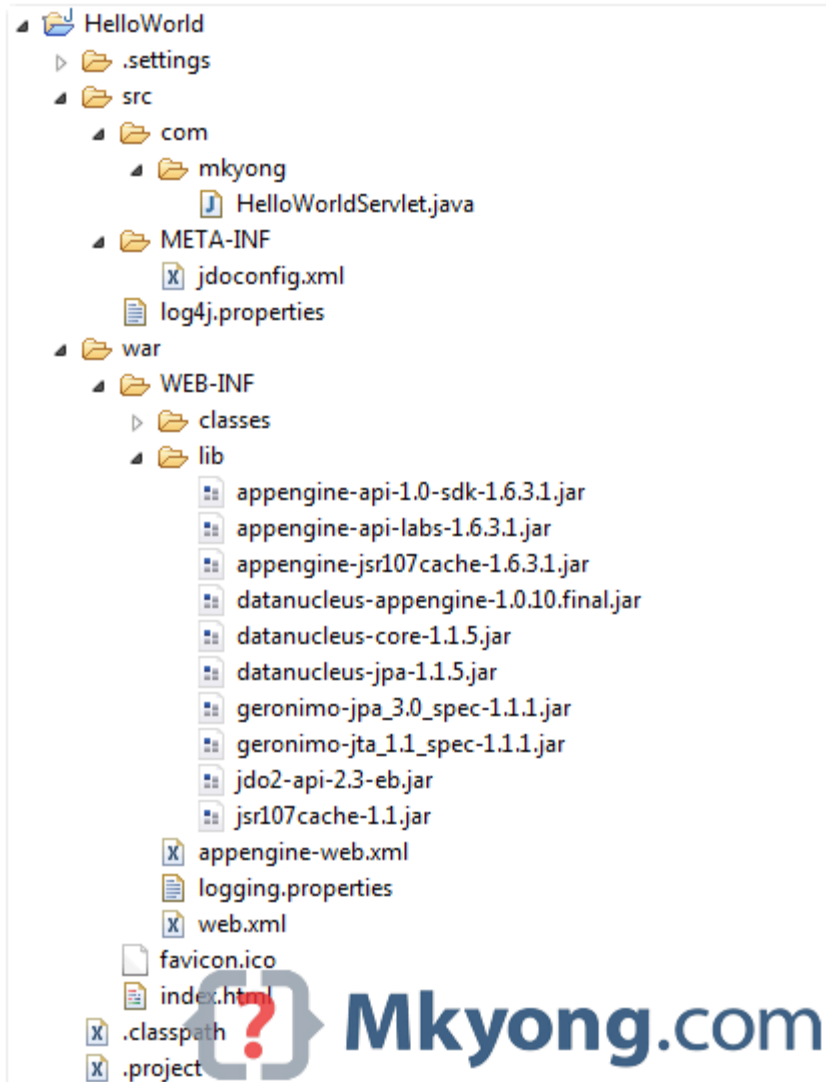
Figure – Deselect the “Google Web ToolKit“, and link your GAE Java SDK via the “configure SDK” link.



Click finished, Google Plugin for Eclipse will generate a sample project automatically.

### 3. Hello World

Review the generated project directory.



Nothing special, a standard Java web project structure.

### HelloWorld/

src/

...Java source code...

META-INF/

...other configuration...

war/

...JSPs, images, data files...

WEB-INF/

...app configuration...

lib/

...JARs for libraries...

classes/

...compiled classes...

Copy

The extra is this file “appengine-web.xml“, Google App Engine need this to run and deploy the application.

*File : appengine-web.xml*

```
<?xml version="1.0" encoding="utf-8"?>
<appengine-web-app xmlns="http://appengine.google.com/ns/1.0">
  <application></application>
  <version>1</version>

  <!-- Configure java.util.logging -->
  <system-properties>
    <property name="java.util.logging.config.file" value="WEB-INF/logging.properties"/>
  </system-properties>
</appengine-web-app>
```

Copy

#### 4. Run it local

Right click on the project and run as “**Web Application**“.

*Eclipse console :*

//...

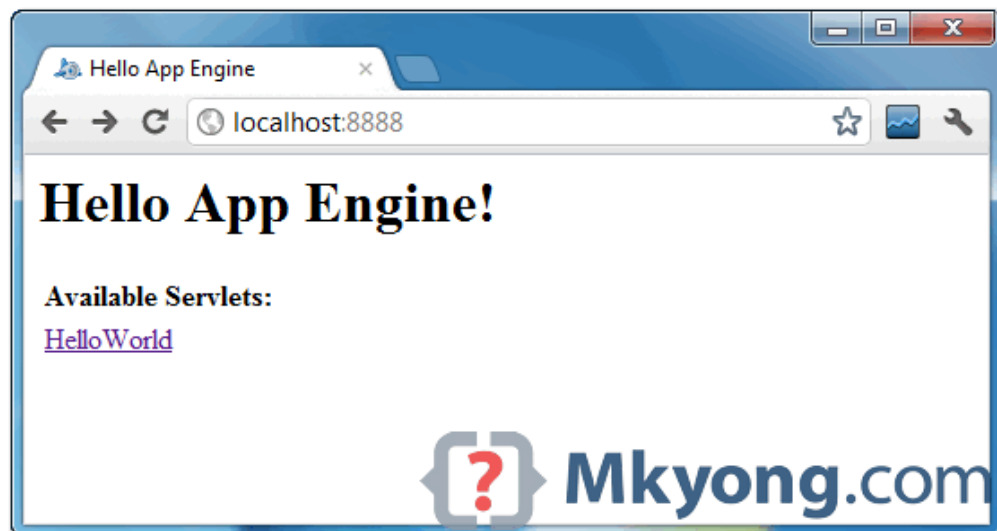
INFO: The server is running at http://localhost:8888/

30 Mac 2012 11:13:01 PM com.google.appengine.tools.development.DevAppServerImpl start

INFO: The admin console is running at http://localhost:8888/\_ah/admin

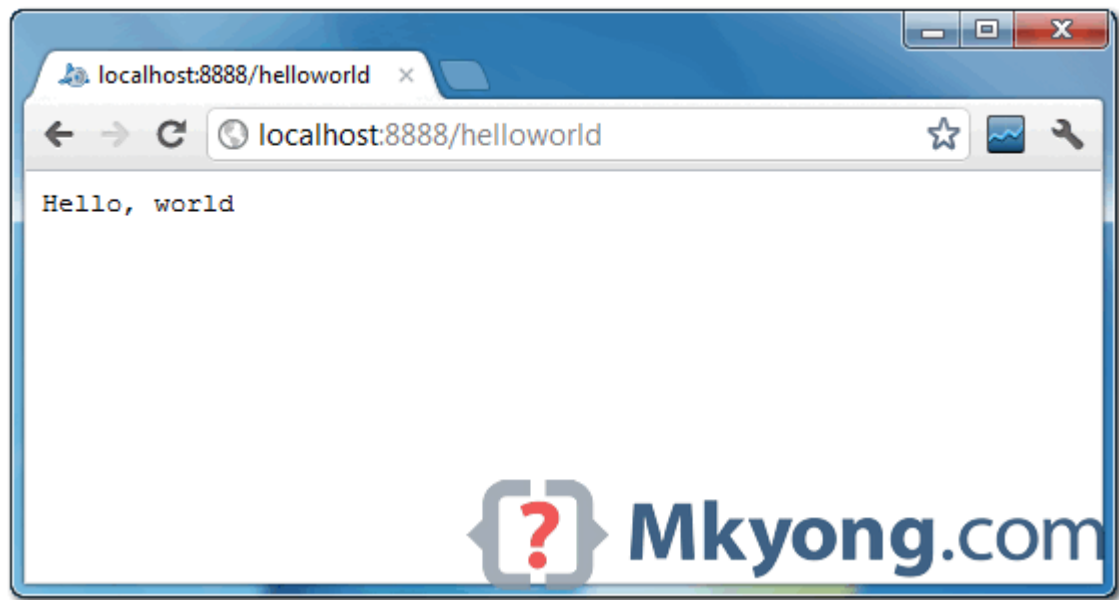
Copy

Access URL <http://localhost:8888/>, see output





and also the hello world servlet – <http://localhost:8888/helloworld>



### 5. Deploy to Google App Engine

Register an account on <https://appengine.google.com/>, and create an application ID for your web application.

In this demonstration, I created an application ID, named “mkyong123”, and put it in `appengine-web.xml`.

*File : `appengine-web.xml`*

```
<?xml version="1.0" encoding="utf-8"?>
<appengine-web-app xmlns="http://appengine.google.com/ns/1.0">
  <application>mkyong123</application>
  <version>1</version>

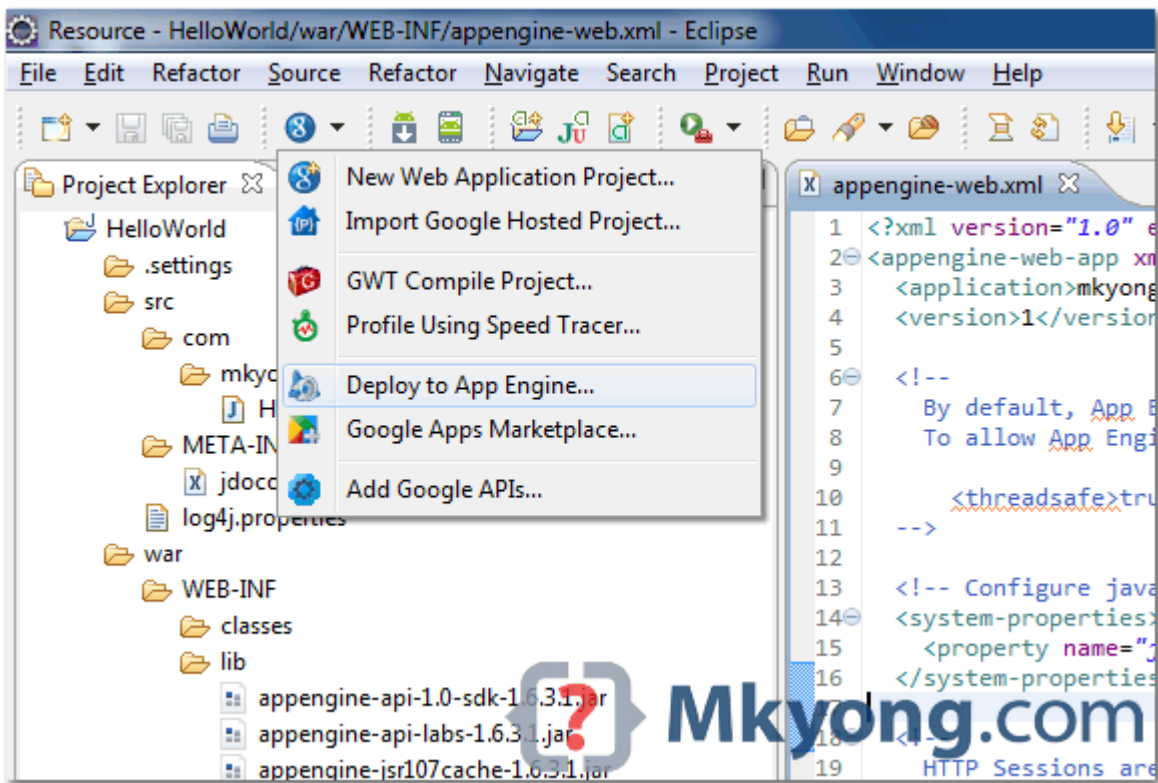
  <!-- Configure java.util.logging -->
  <system-properties>
    <property name="java.util.logging.config.file" value="WEB-INF/logging.properties"/>
  </system-properties>

</appengine-web-app>
```

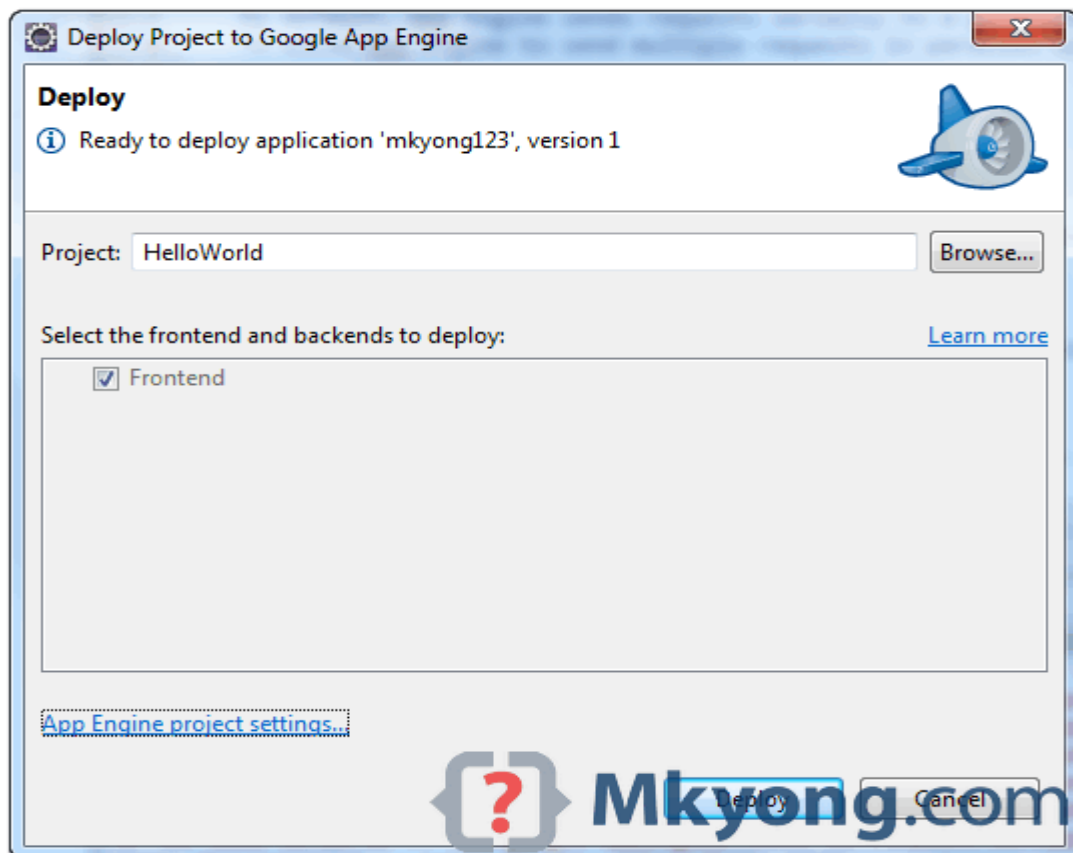
Copy

To deploy, see following steps:

*Figure 1.1 – Click on GAE deploy button on the toolbar.*



*Figure 1.2 – Sign in with your Google account and click on the Deploy button.*



*Figure 1.3 – If everything is fine, the hello world web application will be deployed to this URL – <http://mkyong123.appspot.com/>*



**Result:**

Thus the simple application was created successfully.

EX. NO.:2

DATE:

## **Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.**

**Aim:** To Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

**Steps:**

### **How to use CloudSim in Eclipse**

CloudSim is written in Java. The knowledge you need to use CloudSim is basic Java programming and some basics about cloud computing. Knowledge of programming IDEs such as Eclipse or NetBeans is also helpful. It is a library and, hence, CloudSim does not have to be installed. Normally, you can unpack the downloaded package in any directory, add it to the Java classpath and it is ready to be used. Please verify whether Java is available on your system.

To use CloudSim in Eclipse:

1. Download CloudSim installable files from <https://code.google.com/p/cloudsim/downloads/list> and unzip
  2. Open Eclipse
  3. Create a new Java Project: File -> New
  4. Import an unpacked CloudSim project into the new Java Project
- The first step is to initialise the CloudSim package by initialising the CloudSim library, as follows

`CloudSim.init(num_user, calendar, trace_flag)`

5. Data centres are the resource providers in CloudSim; hence, creation of data centres is a second step. To create Datacenter, you need the DatacenterCharacteristics object that stores the properties of a data centre such as architecture, OS, list of machines, allocation policy that covers the time or spaceshared, the time zone and its price:

`Datacenter datacenter9883 = new Datacenter(name, characteristics, new VmAllocationPolicySimple(hostList), s`

6. The third step is to create a broker:

`DatacenterBroker broker = createBroker();`

7. The fourth step is to create one virtual machine unique ID of the VM, userId ID of the VM's owner, mips, number Of Pes amount of CPUs, amount of RAM, amount of bandwidth, amount of storage, virtual machine monitor, and cloudletScheduler policy for cloudlets:

`Vm vm = new Vm(vmid, brokerId, mips, pesNumber, ram, bw, size, vmm, new CloudletSchedulerTimeShared())`

```
broker.submitVmList(vmlist)
```

```
Cloudlet cloudlet = new Cloudlet(id, length, pesNumber, fileSize, outputSize,
```

10. Submit the cloudlet list to the broker:

Sample Output from the Existing Example:

## CloudSimExample1...

## Starting CloudSim

is starting...

Broker is

Entities started.

```

: Broker: VM #0 has been created in Datacenter #2, Host #0

```

400.1 : Broker: Cloudlet 0 received

Finishing 400.1: Broker: Destroying VM #0

Simulation: No more future

CloudInformationService: Notify all CloudSim entities for shutting down. Datacenter 0 is shutting down...

down Simulation

Simulation completed.

===== OUTPUT =====

Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time
Finish Time 0	SUCCESS	2		0	400
0.1		400.1			

\*\*\*\*\*Datacenter:

Datacenter_0*****	User id
Debt	
3	35.6

CloudSimExample1 finished!

## RESULT:

The simulation was successfully executed.

EX.NO.:3

DATE:

## Use GAE launcher to launch the web applications.

### Aim:

To Use GAE launcher to launch the web applications.

### Steps:

#### Making your First Application

Now you need to create a simple application. We could use the “+” option to have the launcher make us an application – but instead we will do it by hand to get a better sense of what is going on.

Make a folder for your Google App Engine applications. I am going to make the Folder on my Desktop called “**apps**” – the path to this folder is:

C:\Documents and Settings\csev\Desktop\apps

And then make a sub-- folder in within apps called “ae--01--trivial” – the path to this folder would be:

C:\Documents and Settings\csev\Desktop\apps\ae--01--trivial

Using a text editor such as JEdit ([www.jedit.org](http://www.jedit.org)), create a file called app.yaml in the ae--01--trivial folder with the following contents:

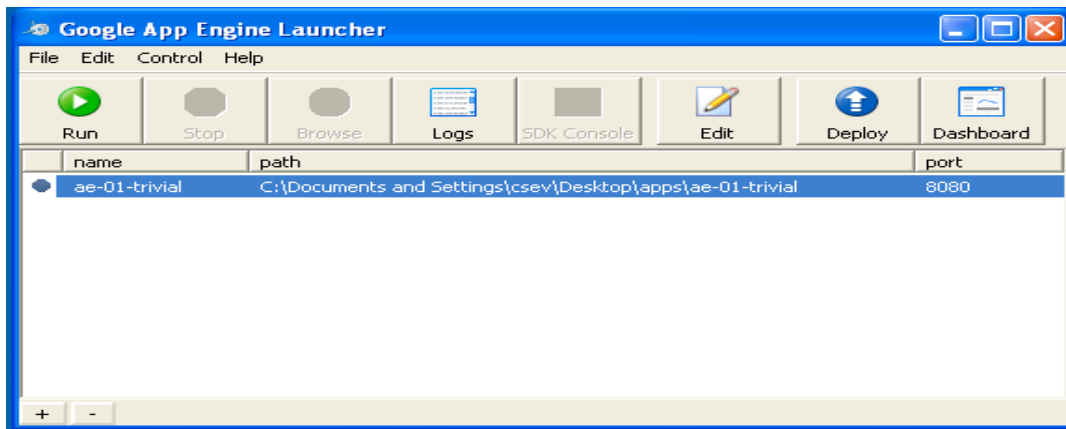
```
application: ae-01-trivial
version: 1
runtime: python
api_version: 1
handlers:- url: /*
  script: index.py
```

Note: Please do not copy and paste these lines into your text editor – you might end up with strange characters – simply type them into your editor.

Then create a file in the ae--01--trivial folder called index.py with three lines in it:

```
print 'Content-Type: text/plain'
print ' '
print 'Hello there Chuck'
```

Then start the GoogleAppEngineLauncher program that can be found under Applications. Use the File --> Add Existing Application command and navigate into the apps directory and select the ae--01--trivial folder. Once you have added the application, select it so that you can control the application using the launcher.



Once you have selected your application and press Run. After a few moments your application will start and the launcher will show a little green icon next to your application. Then press Browse to open a browser pointing at your application which is running at <http://localhost:8080/>

Paste <http://localhost:8080> into your browser and you should see your application as follows:



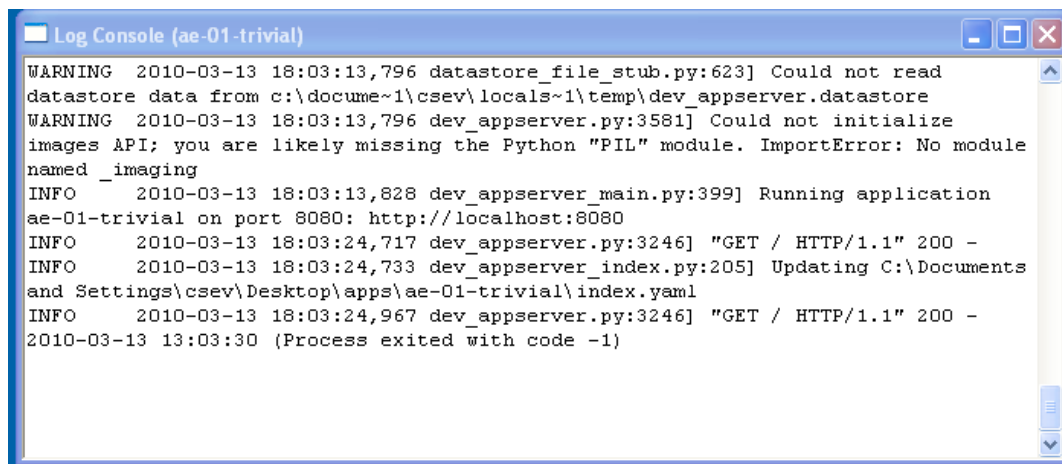
Just for fun, edit the index.py to change the name "Chuck" to your own name and press Refresh in the browser to verify your updates.

## Watching the Log

You can watch the internal log of the actions that the webserver is performing when you are interacting with your application in the browser. Select your application in the Launcher and press the Logs button to bring up a log window:

Each time you press Refresh in your browser—you can see it retrieving the output with a GET request.

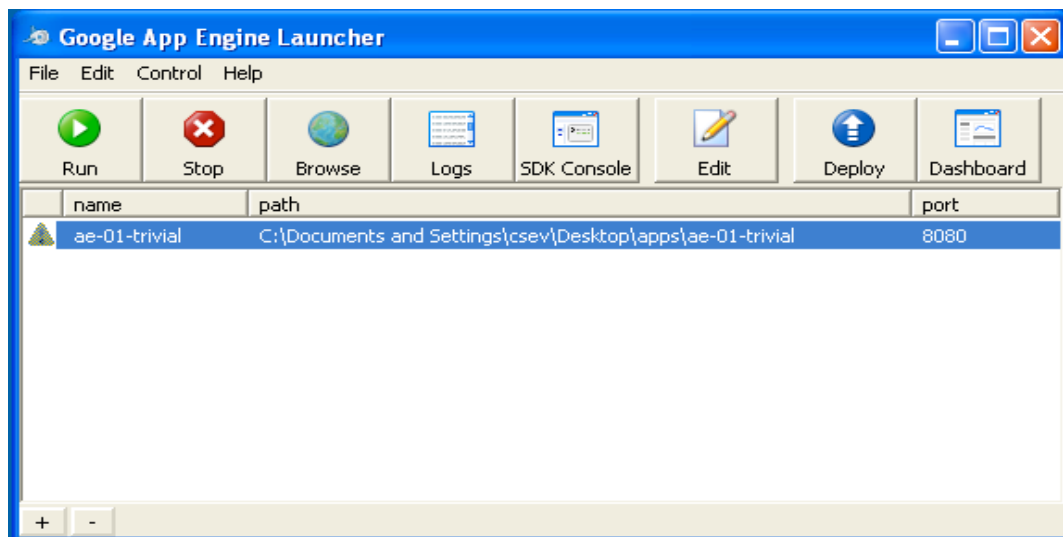




```
Log Console (ae-01-trivial)
WARNING 2010-03-13 18:03:13,796 datastore_file_stub.py:623] Could not read
datastore data from c:\docume~1\csev\locals~1\temp\dev_appserver.datastore
WARNING 2010-03-13 18:03:13,796 dev_appserver.py:3581] Could not initialize
images API; you are likely missing the Python "PIL" module. ImportError: No module
named _imaging
INFO 2010-03-13 18:03:13,828 dev_appserver_main.py:399] Running application
ae-01-trivial on port 8080: http://localhost:8080
INFO 2010-03-13 18:03:24,717 dev_appserver.py:3246] "GET / HTTP/1.1" 200 -
INFO 2010-03-13 18:03:24,733 dev_appserver_index.py:205] Updating C:\Documents
and Settings\csev\Desktop\apps\ae-01-trivial\index.yaml
INFO 2010-03-13 18:03:24,967 dev_appserver.py:3246] "GET / HTTP/1.1" 200 -
2010-03-13 13:03:30 (Process exited with code -1)
```

## Dealing With Errors

With two files to edit, there are two general categories of errors that you may encounter. If you make a mistake on the app.yaml file, the App Engine will not start and your launcher will show a yellow icon near your application:

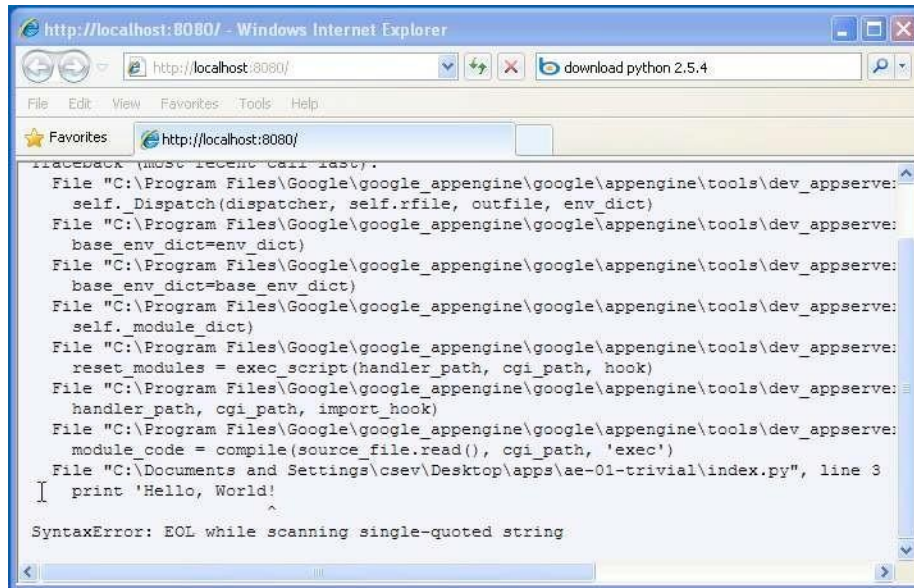


To get more detail on what is going wrong, take a look at the log for the application:



```
Log Console (ae-01-trivial)
Invalid object:
Unknown url handler type.
<URLMap
  static_dir=None
  secure=default
  script=None
  url=/. *
  static_files=None
  upload=None
  mime_type=None
  login=optional
  require_matching_file=None
  auth_fail_action=redirect
  expiration=None
>
in "C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial\app.yaml", line 8,
column 1
```

In this instance – the mistake is mis-- indenting the last line in the app.yaml (line 8). If you make a syntax error in the index.py file, a Python traceback error will appear in your browser.



```
File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
self._Dispatch(dispatcher, self.rfile, outfile, env_dict)
File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
base_env_dict=env_dict)
File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
base_env_dict=base_env_dict)
File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
self._module_dict)
File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
reset_modules = exec_script(handler_path, cgi_path, hook)
File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
handler_path, cgi_path, import_hook)
File "C:\Program Files\Google\google_appengine\google\appengine\tools\dev_appserve:
module_code = compile(source_file.read(), cgi_path, 'exec')
File "C:\Documents and Settings\csev\Desktop\apps\ae-01-trivial\index.py", line 3
I  print 'Hello, World!
      ^
SyntaxError: EOL while scanning single-quoted string
```

The error you need to see is likely to be the last few lines of the output – in this case I made a Python syntax error on line one of our one-- line application.

Reference: [http://en.wikipedia.org/wiki/Stack\\_trace](http://en.wikipedia.org/wiki/Stack_trace)

When you make a mistake in the app.yaml file – you must fix the mistake and attempt to start the application again.

If you make a mistake in a file like index.py, you can simply fix the file and press refresh in your browser – there is no need to restart the server.

## Shutting Down the Server

To shut down the server, use the Launcher, select your application and press the Stop button.

## Result:

Thus the GAE web applications was created.

EX.NO:4  
DATE:

**Find a procedure to transfer the files from one virtual machine to another virtual machine.**

**Aim:**

To Find a procedure to transfer the files from one virtual machine to another virtual machine.

**Steps:**

1. You can copy few (or more) lines with *copy & paste* mechanism.  
For this you need to share clipboard between host OS and guest OS, installing Guest Addition on both the virtual machines (probably setting *bidirectional* and restarting them). You *copy* from *guest OS* in the clipboard that is shared with the *host OS*.  
Then you *paste* from the *host OS* to the second *guest OS*.
2. You can enable drag and drop too with the same method (Click on the machine, settings, general, advanced, drag and drop: set to *bidirectional* )
3. You can have common *Shared Folders* on both virtual machines and use one of the directory shared as buffer to copy.  
Installing Guest Additions you have the possibility to set Shared Folders too. As you put a file in a shared folder from *host OS* or from *guest OS*, is immediately visible to the other. (Keep in mind that can arise some problems for date/time of the files when there are different clock settings on the different virtual machines).  
*If you use the same folder shared on more machines you can exchange files directly copying them in this folder.*
4. You can use usual method to copy files between 2 different computer with client-server application. (e.g. scp with sshd active for linux, winscp... you can get some info about SSH servers e.g. [here](#))  
You need an active server (sshd) on the receiving machine and a client on the sending machine. Of course you need to have the authorization setted (via password or, better, via an automatic authentication method).  
Note: many Linux/Ubuntu distribution install sshd by default: you can see if it is running with `pgrep sshd` from a shell. You can install with `sudo apt-get install openssh-server`.
5. You can mount part of the file system of a virtual machine via NFS or SSHFS on the other, or you can share file and directory with Samba.  
You may find interesting the article [Sharing files between guest and host without VirtualBox shared folders with detailed step by step instructions](#).

You should remember that you are dialling with a little network of machines with different operative systems, and in particular:

- Each virtual machine has its own operative system running on and acts as a physical machine.
- Each virtual machine is an instance of a program *owned* by an *user* in the hosting operative system and should undergo the restrictions of the *user* in the *hosting OS*.

E.g Let we say that Hastur and Meow are users of the hosting machine, but they did not allow each other to see their directories (no read/write/execute authorization). When each of them run a virtual machine, for the hosting OS those virtual machine are two normal programs owned by Hastur and Meow and cannot see the private directory of the other user. This is a restriction due to the *hosting OS*. It's easy to overcome it: it's enough to give authorization to read/write/execute to a directory or to choose a different directory in which both users can read/write/execute.

- Windows likes mouse and Linux fingers. :-)

I mean I suggest you to enable *Drag & drop* to be cosy with the Windows machines and the *Shared folders* or to be cosy with Linux.

When you will need to be fast with Linux you will feel the need of ssh-keygen and to Generate once SSH Keys to copy files on/from a remote machine without writing password anymore. In this way it functions bash auto-completion remotely too!

## PROCEDURE:

### Steps:

1. Open Browser, type localhost:9869
2. Login using username: oneadmin, password: opennebula
3. Then follow the steps to migrate VMs
  - a. Click on infrastructure
  - b. Select clusters and enter the cluster name
  - c. Then select host tab, and select all host
  - d. Then select Vnets tab, and select all vnet
  - e. Then select datastores tab, and select all datastores
  - f. And then choose host under infrastructure tab
  - g. Click on + symbol to add new host, name the host then click on create.
4. on instances, select VMs to migrate then follow the steps
  - a. Click on 8<sup>th</sup> icon ,the drop down list display
  - b. Select migrate on that ,the popup window display
  - c. On that select the target host to migrate then click on migrate.

## Before migration

### Host:SACET

The screenshot shows the OpenNebula web interface for Host 1 naveenkumar. The left sidebar contains navigation links: Dashboard, Instances, VMs, Services, Virtual Routers, Templates, Storage, Network, Infrastructure, Clusters, Hosts, Zones, System, and Settings. The main content area displays the VMs tab for Host 1 naveenkumar. The VMs table shows 6 entries, all with a status of FAILURE. The table columns are ID, Owner, Group, Name, Status, Host, and IPs. The bottom of the interface shows a Windows taskbar with the time 2:34 PM on 8/23/2016.

ID	Owner	Group	Name	Status	Host	IPs
5	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.205
4	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.204
3	oneadmin	oneadmin	vm1	FAILURE	naveenkumar	172.16.100.203
2	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.202
1	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.201
0	oneadmin	oneadmin	ttlinux-0	FAILURE	naveenkumar	172.16.100.200

## Host:one-sandbox

The screenshot shows the OpenNebula web interface for Host 0 one-sandbox. The left sidebar contains navigation links: Dashboard, Instances, VMs, Services, Virtual Routers, Templates, Storage, Network, Infrastructure, Clusters, Hosts, Zones, System, and Settings. The main content area displays the VMs tab for Host 0 one-sandbox. The VMs table shows 2 entries, both with a status of RUNNING. The table columns are ID, Owner, Group, Name, Status, Host, and IPs. The bottom of the interface shows a Windows taskbar with the time 2:34 PM on 8/23/2016.

ID	Owner	Group	Name	Status	Host	IPs
7	oneadmin	oneadmin	vm8	RUNNING	one-sandbox	172.16.100.207
6	oneadmin	oneadmin	vm8	RUNNING	one-sandbox	172.16.100.206

Open Nebula VMs

### Migrate Virtual Machine

VM 6 vm8 is currently running on Host one-sandbox  
VM 7 vm8 is currently running on Host one-sandbox

Select a Host

Please select a Host from the list

ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status
2	rasa	default	0	0 / 0	0K / -	RETRY
1	naveenkumar	rama	6	62 / 0	441B / -	ERROR
0	one-sandbox	rama	2	20 / 100 (20%)	411B / 7411B (1%)	ON

Showing 1 to 3 of 3 entries

Previous 1 Next

Advanced Options

Migrate

Open Nebula VMs

Dashboard

Instances

VMs

Services

Virtual Routers

Templates

Storage

Network

Infrastructure

Clusters

Hosts

Zones

System

Settings

Support Not connected

Upgrade Available

8 TOTAL 2 ACTIVE 0 OFF 0 PENDING 6 FAILED

ID	Owner	Group	Name	Status	Host	IPs
7	oneadmin	oneadmin	vm8	SAVE	naveenkumar	172.16.100.207
6	oneadmin	oneadmin	vm8	SAVE	naveenkumar	172.16.100.206
5	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.205
4	oneadmin	oneadmin	vm2	FAILURE	naveenkumar	172.16.100.204
3	oneadmin	oneadmin	vm1	FAILURE	naveenkumar	172.16.100.203
2	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.202
1	oneadmin	oneadmin	naveen	FAILURE	naveenkumar	172.16.100.201
0	oneadmin	oneadmin	tylinux-0	FAILURE	naveenkumar	172.16.100.200

Showing 1 to 8 of 8 entries

Previous 1 Next

## After Migration:

Open Nebula Sunstone interface showing Hosts. The table displays the following data:

ID	Name	Cluster	RVMs	Allocated CPU	Allocated MEM	Status
2	raa	default	0	0 / 0	0 KB / -	ERROR
1	naveenkumar	rama	8	82 / 0	48 MB / -	ERROR
0	one-sandbox	rama	0	0 / 100 (0%)	0 KB / 741 MB (0%)	ON

Summary: 3 TOTAL, 1 ON, 0 OFF, 2 ERROR

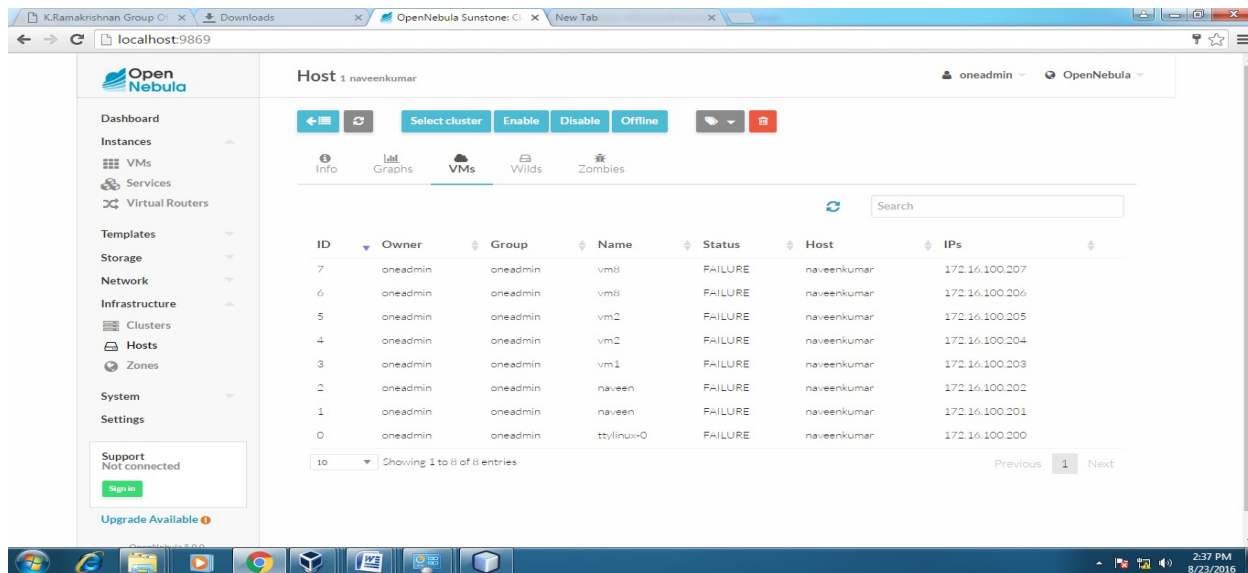
## Host:one-sandbox

Open Nebula Sunstone interface showing Host:one-sandbox. The table displays the following data:

ID	Owner	Group	Name	Status	Host	IPs
----	-------	-------	------	--------	------	-----

Summary: There is no data available

# Host:SACET



## APPLICATIONS:

Easily migrate your virtual machine from one pc to another.

## Result:

Thus the file transfer between VM was successfully completed.....



EX.NO:5

DATE:

## Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)

### Aim:

To Find a procedure to launch virtual machine using trystack.

### Steps:

**OpenStack** is an open-source software cloud computing platform.

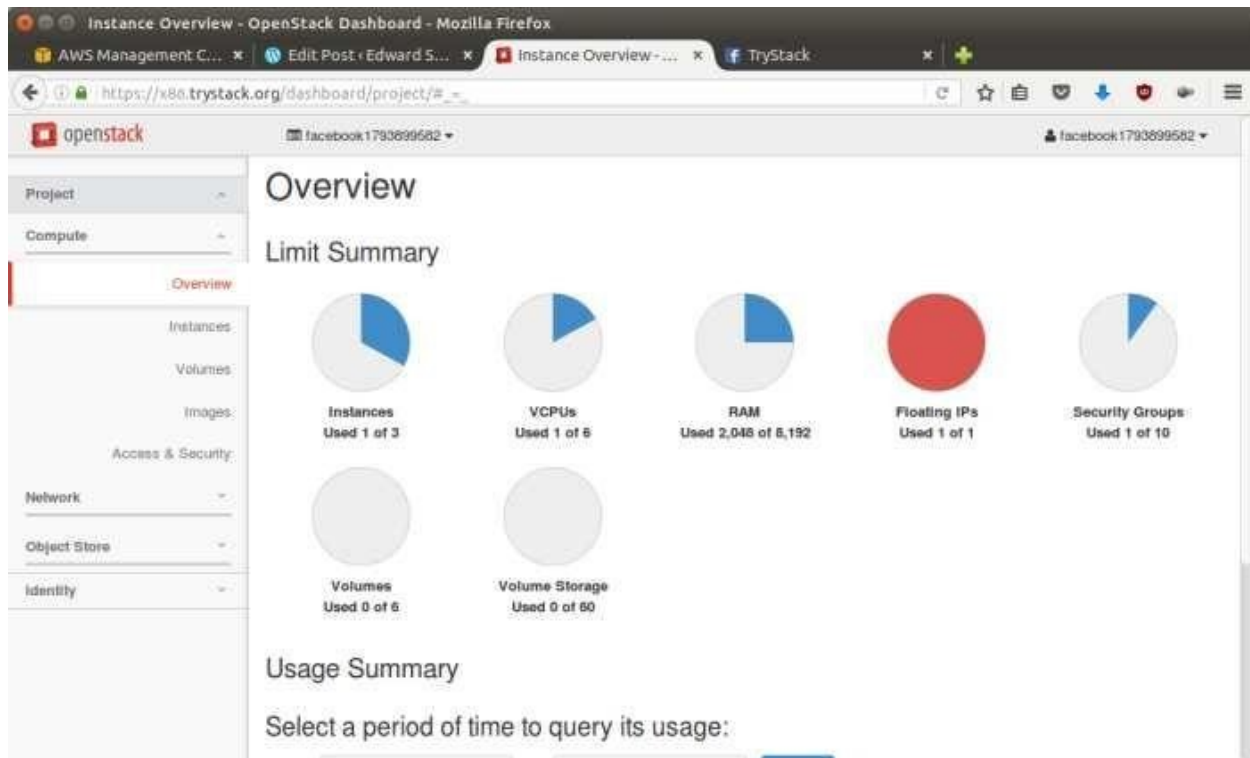
OpenStack is primarily used for deploying an infrastructure as a service (IaaS) solution like Amazon Web Service (AWS). In other words, you can *make your own AWS* by using OpenStack. If you want to try out OpenStack, **TryStack** is the easiest and free way to do it.

In order to try OpenStack in TryStack, you must register yourself by joining TryStack Facebook Group. The acceptance of group needs a couple days because it's approved manually. After you have been accepted in the TryStack Group, you can log in TryStack.



TryStack.org Homepage

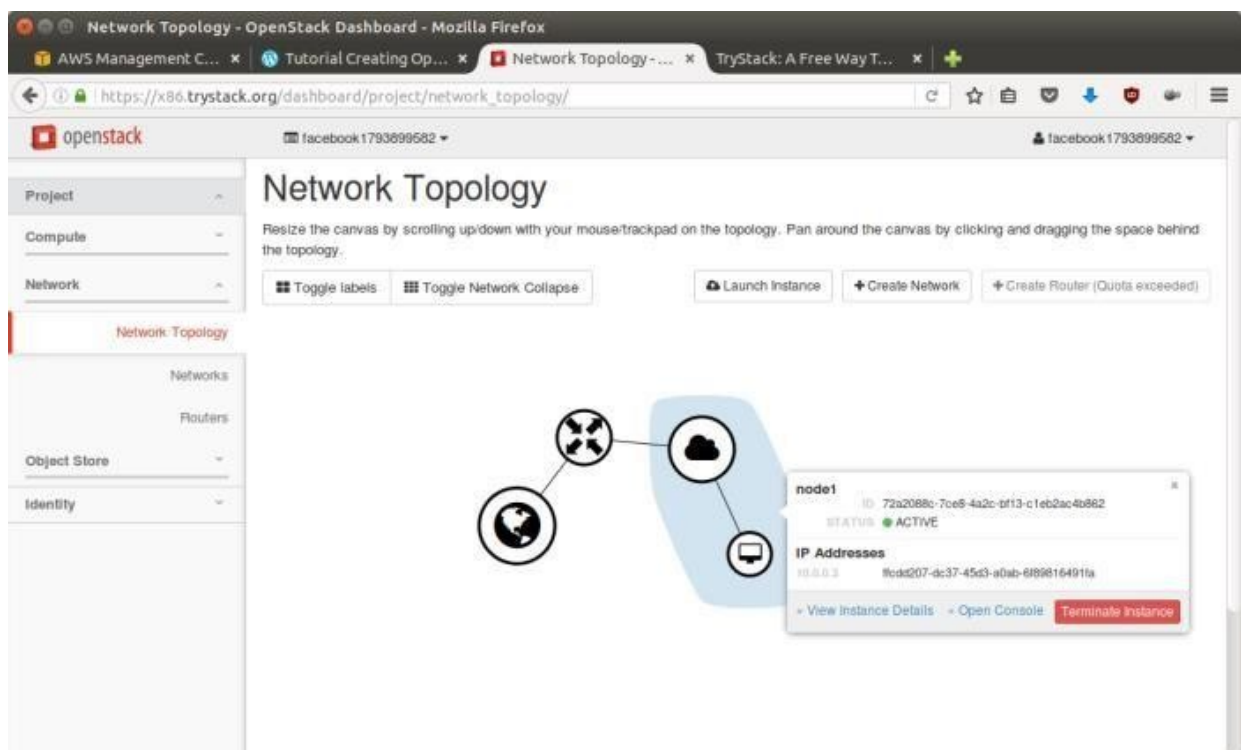
I assume that you already join to the Facebook Group and login to the dashboard. After you log in to the TryStack, you will see the Compute Dashboard like:



## OpenStack Compute Dashboard

### Overview: What we will do?

In this post, I will show you how to run an OpenStack instance. The instance will be accessible through the internet (have a public IP address). The final topology will like:



## Network topology

As you see from the image above, the instance will be connected to a local network and the local network will be connected to internet.

### Step 1: Create Network

Network? Yes, the network in here is our own local network. So, your instances will be not mixed up with the others. You can imagine this as your own LAN (Local Area Network) in the cloud.

1. Go to **Network > Networks** and then click **Create Network**.
2. In **Network** tab, fill **Network Name** for example `internal` and then click **Next**.
3. In **Subnet** tab,
  1. Fill **Network Address** with appropriate CIDR, for example `192.168.1.0/24`. Use **private network CIDR block** as the best practice.
  2. Select **IP Version** with appropriate IP version, in this case `IPv4`.
  3. Click **Next**.
4. In **Subnet Details** tab, fill **DNS Name Servers** with `8.8.8.8` (Google DNS) and then click **Create**.

### Step 2: Create Instance

Now, we will create an instance. The instance is a virtual machine in the cloud, like AWS EC2. You need the instance to connect to the network that we just created in the previous step.

1. Go to **Compute > Instances** and then click **Launch Instance**.
2. In **Details** tab,
  1. Fill **Instance Name**, for example `Ubuntu 1`.
  2. Select **Flavor**, for example `m1.medium`.
  3. Fill **Instance Count** with `1`.
  4. Select **Instance Boot Source** with **Boot from Image**.
  5. Select **Image Name** with **Ubuntu 14.04 amd64 (243.7 MB)** if you want install Ubuntu 14.04 in your virtual machine.
3. In **Access & Security** tab,
  1. Click **[+]** button of **Key Pair** to import key pair. This key pair is a public and private key that we will use to connect to the instance from our machine.
  2. In **Import Key Pair** dialog,
    1. Fill **Key Pair Name** with your machine name (for example `Edward-Key`).
    2. Fill **Public Key** with your **SSH public key** (usually is in `~/.ssh/id_rsa.pub`). See description in Import Key Pair dialog box for more information. If you are using Windows, you can use **Puttygen** to generate key pair.
    3. Click **Import key pair**.
  3. In **Security Groups**, mark/check **default**.



4. In **Networking** tab,
  1. In **Selected Networks**, select network that have been created in Step 1, for example **internal**.
5. Click **Launch**.
6. If you want to create multiple instances, you can repeat step 1-5. I created one more instance with instance name **Ubuntu 2**.

### Step 3: Create Router

I guess you already know what router is. In the step 1, we created our network, but it is isolated. It doesn't connect to the internet. To make our network has an internet connection, we need a router that running as the gateway to the internet.

1. Go to **Network > Routers** and then click **Create Router**.
2. Fill **Router Name** for example **router1** and then click **Create router**.
3. Click on your **router name link**, for example **router1**, **Router Details** page.
4. Click **Set Gateway** button in upper right:
  1. Select **External networks** with **external**.
  2. Then **OK**.
5. Click **Add Interface** button.
  1. Select **Subnet** with the network that you have been created in Step 1.
  2. Click **Add interface**.
6. Go to **Network > Network Topology**. You will see the network topology. In the example, there are two network, i.e. external and internal, those are bridged by a router. There are instances those are joined to internal network.

### Step 4: Configure Floating IP Address

*Floating IP address* is public IP address. It makes your instance is accessible from the internet. When you launch your instance, the instance will have a private network IP, but no public IP. In OpenStack, the public Ips is collected in a pool and managed by admin (in our case is TryStack). You need to request a public (floating) IP address to be assigned to your instance.

1. Go to **Compute > Instance**.
2. In one of your instances, click **More > Associate Floating IP**.
3. In **IP Address**, click Plus [+].
4. Select **Pool** to **external** and then click **Allocate IP**.
5. Click **Associate**.
6. Now you will get a public IP, e.g. 8.21.28.120, for your instance.

## Step 5: Configure Access & Security

OpenStack has a feature like a firewall. It can whitelist/blacklist your in/out connection. It is called *Security Group*.

1. Go to **Compute > Access & Security** and then open **Security Groups** tab.
2. In **default** row, click **Manage Rules**.
3. Click **Add Rule**, choose **ALL ICMP** rule to enable ping into your instance, and then click **Add**.
4. Click **Add Rule**, choose **HTTP** rule to open HTTP port (port 80), and then click **Add**.
5. Click **Add Rule**, choose **SSH** rule to open SSH port (port 22), and then click **Add**.
6. You can open other ports by creating new rules.

## Step 6: SSH to Your Instance

Now, you can SSH your instances to the floating IP address that you got in the step 4. If you are using Ubuntu image, the SSH user will be ubuntu.

## Result:

Thus the openstack demo worked successfully.

Title:

Design and develop custom Application (Mini Project) using Salesforce Cloud.

Theory:

### Introduction

Salesforce.com Inc. is an American cloud-based software company headquartered in San Francisco, California. Though the bulk of its revenue comes from a customer relationship management (CRM) product, Salesforce also sells a complementary suite of enterprise applications focused on customer service, marketing automation, analytics and application development.

Salesforce is the primary enterprise offering within the Salesforce platform. It provides companies with an interface for case management and task management, and a system for automatically routing and escalating important events. The Salesforce customer portal provides customers the ability to track their own cases, includes a social networking plug-in that enables the user to join the conversation about their company on social networking websites, provides analytical tools and other services including email alert, Google search, and access to customers' entitlement and contracts.

### Lightning Platform

Lightning Platform (also known as Force.com) is a platform as a service (PaaS) that allows developers to create add-on applications that integrate into the main Salesforce.com application. These third-party applications are hosted on Salesforce.com's infrastructure. Force.com applications are built using declarative tools, backed by Lightning and Apex (a proprietary Java-like programming language for Force.com) and Lightning and Visual force (a framework that includes an XML syntax typically used to generate HTML). The Force.com platform typically receives three complete releases a year. As the platform is provided as a service to its developers, every single development instance also receives all these updates.

### Community Cloud

Community Cloud provides Salesforce customers the ability to create online web properties for external collaboration, customer service, channel sales, and other custom portals in their instance of Salesforce. Tightly integrated to Sales Cloud, Service Cloud, and App Cloud, Community Cloud can be quickly customized to provide a wide variety of web properties

### Salesforce Sales Cloud

Salesforce Sales Cloud is a customer relationship management (CRM) platform designed to support sales, marketing and customer support in both business-to-business (B2B) and business-to-customer (B2C) contexts. Sales Cloud is a fully customizable product that brings all the customer information together in an integrated platform that incorporates marketing, lead generation, sales, customer service and business analytics and provides access to thousands of applications through the AppExchange. The platform is provided as Software as a Service (SaaS) for browser-based access; a mobile app is also available.

A realtime social feed for collaboration allows users to share information or ask questions of the user

community. Salesforce.com offers five versions of Sales Cloud on a per-user, per month basis,

from lowest to highest: Group, Professional, Enterprise, Unlimited and Performance. The company offers three levels of support contracts: Standard Success Plan, Premier Success Plan and Premier+ Success Plan.

### Create Custom Apps for Salesforce Classic

Create custom apps to give your Salesforce Classic users' access to everything they need all in one place.

If you're new to custom apps, we recommend using Lightning Platform quick start to create an app. With this tool, you can generate a basic working app in just one step.

If you've already created the objects, tabs, and fields you need for your app, follow these steps. With this option, you create an app label and logo, add items to the app, and assign the app to profiles.

1. From Setup, enter Apps in the Quick Find box, then select Apps.
2. Click New.
3. If the Salesforce console is available, select whether you want to define a custom app or a Salesforce console.
4. Give the app a name and description.  
An app name can have a maximum of 40 characters, including spaces.
5. Optionally, brand your app by giving it a custom logo.
6. Select which items to include in the app.
7. Optionally, set the default landing tab for your new app using the Default Landing Tab drop-down menu below the list of selected tabs. This determines the first tab a user sees when logging into this app.
8. Choose which profiles the app will be visible to.
9. Check the Default box to set the app as that profile's default app, meaning that new users with the profile see this app the first time they log in. Profiles with limits are excluded from this list.
10. Click Save

What is the difference between custom application and console application in sales force?

A custom application is a collection of tabs, objects etc that function together to solve a particular problem.

A console application uses a specific Salesforce UI - the console. Console applications are intended to enhance productivity by allowing everything to be done from a single, tabbed, screen.

Conclusion:



Title:

Design an Assignment to retrieve, verify, and store user credentials using Firebase Authentication, the Google App Engine standard environment, and Google Cloud Data store.

Theory:

Firebase Authentication

Most apps need to know the identity of a user. Knowing a user's identity allows an app to securely save user data in the cloud and provide the same personalized experience across all of the user's devices.

Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to your app. It supports authentication using passwords, phone numbers, popular federated identity providers like Google, Facebook and Twitter, and more. Firebase Authentication integrates tightly with other Firebase services, and it leverages industry standards like OAuth 2.0 and OpenID Connect, so it can be easily integrated with your custom backend.

How does it work?

To sign a user into your app, you first get authentication credentials from the user. These credentials can be the user's email address and password, or an OAuth token from a federated identity provider. Then, you pass these credentials to the Firebase Authentication SDK. Our backend services will then verify those credentials and return a response to the client.

After a successful sign in, you can access the user's basic profile information, and you can control the user's access to data stored in other Firebase products. You can also use the provided authentication token to verify the identity of users in your own backend services.

Authenticating Users on App Engine Using Firebase:

Now we show how to retrieve, verify, and store user credentials using Firebase Authentication, the Google App Engine standard environment, and Google Cloud Data store.

The document walks you through a simple note-taking application called Firenotes that stores users' notes in their own personal notebooks. Notebooks are stored per user, and identified by each user's unique Firebase Authentication ID. The application has the following components:

1. The frontend configures the sign-in user interface and retrieves the Firebase Authentication ID. It also handles authentication state changes and lets users see their notes.
2. FirebaseUI is an open-source, drop-in solution that handles user login, linking multiple providers to one account, recovering passwords, and more. It implements authentication best practices for a smooth and secure sign-in experience.
3. The backend verifies the user's authentication state and returns user profile information as well as the user's notes.

The application stores user credentials in Cloud Data store by using the NDB client library, but you can store the credentials in a database of your choice.

The following diagram shows how the frontend and backend communicate with each other and how user credentials travel from Firebase to the database.

Objectives:

Configure the Firebase Authentication user interface.

Obtain a Firebase ID token and verify it using server-side authentication.

Store user credentials and associated data in Cloud Datastore.

Query a database using the NDB client library.

Deploy an app to App Engine.

Laboratory Practice-IV A.Y.-2018-19

Prof.A.R.Jain PVGCOE, Nashik Page 4

Costs:

This tutorial uses billable components of Cloud Platform, including:

Google Cloud Data store

Conclusion:

## VIVA QUESTIONS AND ANSWERS

### **1. Define Cloud Computing with example.**

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

### **2. What is the working principle of Cloud Computing?**

The cloud is a collection of computers and servers that are publicly accessible via the Internet. This hardware is typically owned and operated by a third party on a consolidated basis in one or more data center locations. The machines can run any combination of operating systems.

### **3. What are the advantages and disadvantages of Cloud Computing?**

#### Advantages

- Lower-Cost Computers for Users
- Improved Performance
- Lower IT Infrastructure Costs
- Fewer Maintenance Issues
- Lower Software Costs
- Instant Software Updates
- Increased Computing Power
- Unlimited Storage Capacity
- Increased Data Safety
- Improved Compatibility Between Operating Systems
- Improved Document Format Compatibility
- Easier Group Collaboration
- Universal Access to Documents
- Latest Version Availability
- Removes the Tether to Specific Devices

#### Disadvantages

- Requires a Constant Internet Connection
- Doesn't Work Well with Low-Speed Connections
- Can Be Slow
- Features Might Be Limited
- Stored Data Might Not Be Secure
- If the Cloud Loses Your Data, You're Screwed

### **4. What is distributed system?**

A *distributed system* is a software system in which components located on networked computers communicate and coordinate their actions by passing messages. The components interact with each other in order to achieve a common goal.

Three significant characteristics of distributed systems are:

- ✓ Concurrency of components
- ✓ Lack of a global clock
- ✓ Independent failure of components
- ✓ What is cluster?
- ✓ A computing cluster consists of interconnected stand-alone computers which work cooperatively as a single integrated computing resource. In the past, clustered computer systems have demonstrated

## 5. What is grid computing?

Grid Computing enables virtual organizations to share geographically distributed resources as they pursue common goals, assuming the absence of central location, central control, omniscience, and an existing trust relationship.

(or)

- ✓ Grid technology demands new distributed computing models, software/middleware support, network protocols, and hardware infrastructures.
- ✓ National grid projects are followed by industrial grid platform development by IBM, Microsoft, Sun, HP, Dell, Cisco, EMC, Platform Computing, and others. New grid service providers (GSPs) and new grid applications have emerged rapidly, similar to the growth of the Internet and web services in the past two decades.
- ✓ Grid systems are classified into essentially two categories: computational or data grids and P2P grids.

## 6. What are the business areas needs in Grid computing?

- ✓ Life Sciences
- ✓ Financial services
- ✓ Higher Education
- ✓ Engineering Services
- ✓ Government
- ✓ Collaborative games

## 7. List out the Grid Applications:

- ✓ Application partitioning that involves breaking the problem into discrete pieces
- ✓ Discovery and scheduling of tasks and workflow
- ✓ Data communications distributing the problem data where and when it is required
- ✓ Provisioning and distributing application codes to specific system nodes
- ✓ Autonomic features such as self-configuration, self-optimization, self-recovery and self-management

## 8. List some grid computing toolkits and frameworks?

- ✓ Globus Toolkit Globus Resource Allocation Manager (GRAM)
- ✓ Grid Security Infrastructure (GSI)
- ✓ Information Services
- ✓ Legion, Condor and Condor-G
- ✓ NIMROD, UNICORE, NMI.

## 9. What are Desktop Grids?

These are grids that leverage the compute resources of desktop computers.

Because of the true (but unfortunate) ubiquity of Microsoft® Windows® operating system in corporations, desktop grids are assumed to apply to the Windows environment.

The Mac OS™ environment is supported by a limited number of vendors.

## 10. What are Server Grids?

- ✓ Some corporations, while adopting Grid Computing, keep it limited to server resources that are within the purview of the IT department.
- ✓ Special servers, in some cases, are bought solely for the purpose of creating an internal “utility grid” with resources made available to various departments.
- ✓ No desktops are included in server grids. These usually run some flavor of the Unix/Linux operating system.

## 11. Define Opennebula.

OpenNebula is an open source management tool that helps virtualized data centers oversee private clouds, public clouds and hybrid clouds. OpenNebula is vendor neutral, as well as platform- and API-agnostic. It can use KVM, Xen or VMware hypervisors.

## 12. Define Eclipse.

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment.

### **13. Define Netbeans.**

NetBeans is an open-source integrated development environment (IDE) for developing with Java, PHP, C++, and other programming languages. NetBeans is also referred to as a platform of modular components used for developing Java desktop applications.

### **14. Define Apache Tomcat.**

Apache Tomcat (or Jakarta Tomcat or simply Tomcat) is an open source servlet container developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run."

### **15. What is private cloud?**

The *private cloud* is built within the domain of an intranet owned by a single organization. Therefore, they are client owned and managed. Their access is limited to the owning clients and their partners. Their deployment was not meant to sell capacity over the Internet through publicly accessible interfaces. Private clouds give local users a flexible and agile private infrastructure to run service workloads within their administrative domains.

### **16. What is public cloud?**

A *public cloud* is built over the Internet, which can be accessed by any user who has paid for the service. Public clouds are owned by service providers. They are accessed by subscription. Many companies have built public clouds, namely Google App Engine, Amazon AWS, Microsoft Azure, IBM Blue Cloud, and Salesforce Force.com. These are commercial providers that offer a publicly accessible remote interface for creating and managing VM instances within their proprietary infrastructure.

### **17. What is hybrid cloud?**

A *hybrid cloud* is built with both public and private clouds, Private clouds can also support a *hybrid cloud* model by supplementing local infrastructure with computing capacity from an external public cloud. For example, the *research compute cloud* (RC2) is a private cloud built by IBM.

### **18. What is a Community Cloud ?**

A community cloud in computing is a collaborative effort in which infrastructure is shared between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.), whether managed internally or by a third-party and hosted internally or externally. This is controlled and used by a group of organizations that have shared interest. The costs are spread over fewer users than a public cloud (but more than a private cloud)

### **19. Define IaaS?**

The IaaS layer offers storage and infrastructure resources that is needed to deliver the Cloud services. It only comprises of the infrastructure or physical resource. Top IaaS Cloud Computing Companies: Amazon (EC2), Rackspace, GoGrid, Microsoft, Terremark and Google.

### **20. Define PaaS?**

PaaS provides the combination of both, infrastructure and application. Hence, organisations using PaaS don't have to worry for infrastructure nor for services. Top PaaS Cloud Computing Companies: Salesforce.com, Google, Concur Technologies, Ariba, Unisys and Cisco..

### **21. Define SaaS?**

In the SaaS layer, the Cloud service provider hosts the software upon their servers. It can be defined as a in model in which applications and softwares are hosted upon the server and made available to customers over a network. Top SaaS Cloud Computing Companies: Amazon Web Services, AppScale, CA Technologies, Engine Yard, Salesforce and Windows Azure.

## 22. What is meant by virtualization?

Virtualization is a computer architecture technology by which multiple virtual machines (VMs) are multiplexed in the same hardware machine. The idea of VMs can be dated back to the 1960s. The purpose of a VM is to enhance resource sharing by many users and improve computer performance in terms of resource utilization and application flexibility.

## 23. What are the implementation levels of virtualization?

The virtualization types are following

1. OS-level virtualization
2. ISA level virtualization
3. User-Application Level virtualization
4. hardware level virtualization
5. library level virtualization

## 24. List the requirements of VMM?

There are three requirements for a VMM.

First, a VMM should provide an environment for programs which is essentially identical to the original machine.

Second, programs run in this environment should show, at worst, only minor decreases in speed.

Third, a VMM should be in complete control of the system resources.

## 25. Explain Host OS and Guest OS?

A comparison of the differences between a host system, a guest system, and a virtual machine within a virtual infrastructure.

A host system (host operating system) would be the primary & first installed operating system. If you are using a bare metal Virtualization platform like Hyper-V or ESX, there really isn't a host operating system besides the Hypervisor. If you are using a Type-2 Hypervisor like VMware Server or Virtual Server, the host operating system is whatever operating system those applications are installed into.

A guest system (guest operating system) is a virtual guest or virtual machine (VM) that is installed under the host operating system. The guests are the VMs that you run in your virtualization platform.

## 26. Write the steps for live VM migration?

The five steps for live VM migration is

Stage 0: *Pre-Migration*

Active VM on Host A

Alternate physical host may be preselected for migration

Block devices mirrored and free resources maintained

Stage 1: *Reservation*

Initialize a container on the target host

Stage 2: *Iterative pre-copy*

Enable shadow paging

Copy dirty pages in successive rounds.

Stage 3: *Stop and copy*

Suspend VM on host A

Generate ARP to redirect traffic to Host B

Synchronize all remaining VM state to Host B

Stage 4: *Commitment*

VM state on Host A is released

Stage 5: *Activation*

VM starts on Host B

Connects to local devices

Resumes normal operation

### **27..Define Globus Toolkit: Grid Computing Middleware**

- ✓ Globus is open source grid software that addresses the most challenging problems in distributed resources sharing.
- ✓ The Globus Toolkit includes software services and libraries for distributed security, resource management, monitoring and discovery, and data management.

### **28. Define Blocks in HDFS**

- ✓ A disk has a block size, which is the minimum amount of data that it can read or write. Filesystems for a single disk build on this by dealing with data in blocks, which are an integral multiple of the disk block size. Filesystem blocks are typically a few kilobytes in size, while disk blocks are normally 512 bytes. This is generally transparent to the filesystem user who is simply reading or writing a file—of whatever length.

### **29. Define Namenodes and Datanodes**

- ✓ An HDFS cluster has two types of node operating in a master-worker pattern:
  - a *namenode* (the master) and
  - a number of *datanodes*(workers).
- ✓ The namenode manages the filesystem namespace. It maintains the filesystem tree and the metadata for all the files and directories in the tree. This information is stored persistently on the local disk in the form of two files: the namespace image and the edit log.
- ✓ The namenode also knows the datanodes on which all the blocks for a given file are located, however, it does not store block locations persistently, since this information is reconstructed from datanodes when the system starts.

### **30. Define HADOOP.**

Hadoop is an open source, Java-based programming framework that supports the processing and storage of extremely large data sets in a distributed computing environment. It is part of the Apache project sponsored by the Apache Software Foundation.

### **31. Define HDFS.**

Hadoop Distributed File System (HDFS) is a Java-based file system that provides scalable and reliable data storage that is designed to span large clusters of commodity servers. HDFS, MapReduce, and YARN form the core of Apache™ Hadoop®.

### **32. Write about HADOOP.**

Hadoop was created by Doug Cutting and Mike Cafarella in 2005. Cutting, who was working at Yahoo! at the time, named it after his son's toy elephant. It was originally developed to support distribution for the Nutch search engine project.

### **33. Definition of Grid Portal:**

A *Grid Portal* provides an efficient infrastructure to put Grid-empowered applications on corporate Intranet/Internet.

### **34. Define GAE.**

Google App Engine (often referred to as GAE or simply App Engine) is a Platform as a Service and cloud computing platform for developing and hosting web applications in Google-managed data centers.

Applications are sandboxed and run across multiple servers. App Engine offers automatic scaling for web applications—as the number of requests increases for an application, App Engine automatically allocates more resources for the web application to handle the additional demand.

### **35. What is Cloudsim?**

CloudSim is a simulation toolkit that supports the modeling and simulation of the core functionality of cloud, like job/task queue, processing of events, creation of cloud entities(datacenter, datacenter brokers,

etc), communication between different entities, implementation of broker policies, etc. This toolkit allows to:

- Test application services in a repeatable and controllable environment.
- Tune the system bottlenecks before deploying apps in an actual cloud.
- Experiment with different workload mix and resource performance scenarios on simulated infrastructure for developing and testing adaptive application provisioning techniques

### **36. Core features of CloudSim are:**

- The Support of modeling and simulation of large scale computing environment as federated cloud data centers, virtualized server hosts, with customizable policies for provisioning host resources to virtual machines and energy-aware computational resources
- It is a self-contained platform for modeling cloud's service brokers, provisioning, and allocation policies.
- It supports the simulation of network connections among simulated system elements.
- Support for simulation of federated cloud environment, that inter-networks resources from both private and public domains.
- Availability of a virtualization engine that aids in the creation and management of multiple independent and co-hosted virtual services on a data center node.
- Flexibility to switch between space shared and time shared allocation of processing cores to virtualized services.

### **37. Uses of Cloudsim.**

- Load Balancing of resources and tasks
- Task scheduling and its migrations
- Optimizing the Virtual machine allocation and placement policies
- Energy-aware Consolidations or Migrations of virtual machines
- Optimizing schemes for Network latencies for various cloud scenarios

### **38. Define OpenStack.**

OpenStack is a cloud operating system that controls large pools of compute, storage, and networking resources throughout a datacenter, all managed and provisioned through APIs with common authentication mechanisms. A dashboard is also available, giving administrators control while empowering their users to provision resources through a web interface.

### **39. Define Trystack.**

TryStack is a great way to take OpenStack for a spin without having to commit to a full deployment.

This free service lets you test what the cloud can do for you, offering networking, storage and compute instances, without having to go all in with your own hardware.

It's a labor of love spearheaded by three Red Hat OpenStack experts Will Foster, Kambiz Aghaiepour and Dan Radez.

TryStack's set-up must bear the load of anyone who wants to use it, but instead of an equally boundless budget and paid staff, it was originally powered by donated equipment and volunteers from Cisco, Dell, Equinix, NetApp, Rackspace and Red Hat who pulled together for this OpenStack Foundation project.

### **40. Define Hadoop.**

Hadoop is an open-source software framework for storing data and running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs.



