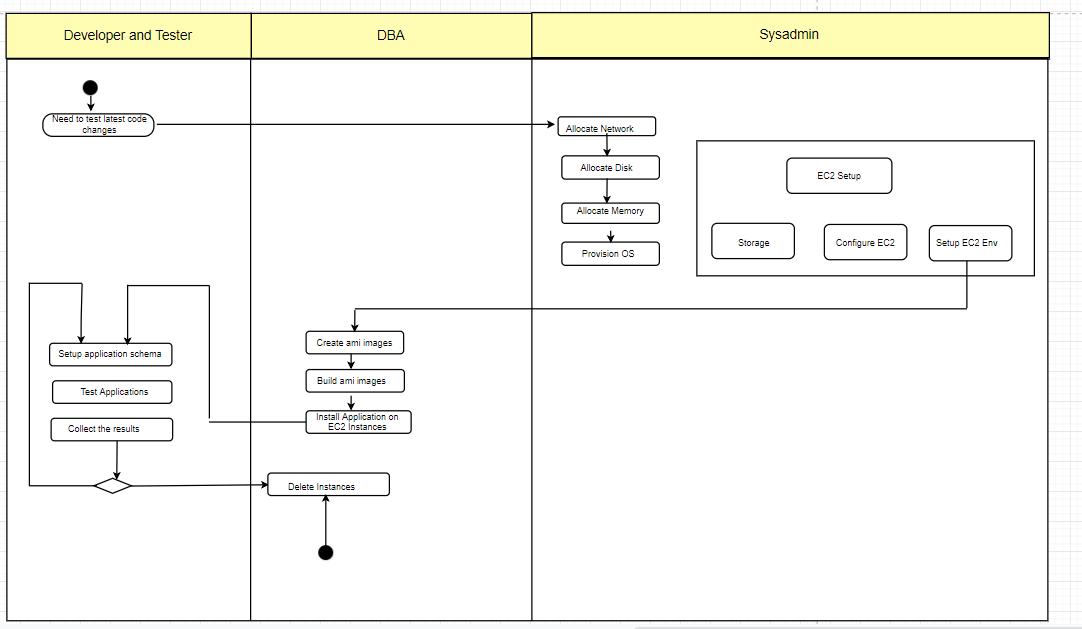
**Ansible to provision infrastructure within AWS Overview:**

Ansible dev-ops tool based on activity diagram, can be very use in any organization to allocate instances for developers.



Standard format in any organizations is there will be 3 entities hence we will have 3 swim lanes to carry out specific tasks.

1. Developer
2. DbAdmin
3. System Admin

Sequence of activities that occur as are below:

* Developer request new environment to test his latest changes. Request is directed to System admin to provision below resources
* allocate network,
* allocate disks
* allocate CPU/memory
* Provision OS

With these EC2 instance will be configured with appropriate security groups.

* Then Next is Database activities follow

DBA configures and sets up the database for developer.

* Then developer sets up application schema and tests it. Once testing is completed instances or containers are deleted.

**Contents/configurations in ansible playbook template:**

Create a ansible playbook, which will be in yaml format for provisioning EC2 instance using ansible. Developers provide the requirements using which we modify the variables in the template and customize and provision the resources.

Run the ansible playbook to automate the app provisioning on AWS

ansible-playbook ec2\_instance.yaml

Various parameters /configurations handled in playbook:

* Necessary Tools to be installed on EC2:

1. Install Ansible Virtual Env

` virtualenv myansible`

source myansible/bin/activate

1. Install ansible

pip install ansible

1. Installl boto boto3

pip install boto boto3

Now to check the installed ansible version on ec2 below commands are used:

ansible --version

ansible 2.7.10

* Important parameters to be changed in ec2\_instance.yaml

instance\_type: t2.micro

security\_group: connectme\_sg

image: ami-0019ef04ac50be30f

region: us-west-1

keypair: drifters

aws\_access\_key: <PASTE your ACCESS KEY>

aws\_secret\_key: <PASTE your SECRET KEY>

count: 2

hosts\_file: /tmp/hosts

Remaining portions mentioned below in ymal is automated with python and ansible:

* Running nginx server
* Configuring gcc packages
* Creating number of host groups for collecting the newly created ec2 instances ip address in a template.
* Playbook contains github url from which Repo will be cloned on AWS EC2 instance with appropriate security settings.
* Add host handler keeps track of creation of EC2 instances and is stored in a list.

For every new box request from developer to test the changes made with 2 instances up and running, Security group must be different and previous, or any other instances should not be modified.

Under Security Group for Newly Created EC2 Instance below variables are configured.

Ports: Accessible and open for developer.

rules\_egress : Allow from which user the resources can be accessed.

rules:

- proto: tcp

from\_port: 22

to\_port: 22

cidr\_ip: 0.0.0.0/0

- proto: tcp

from\_port: 80

to\_port: 80

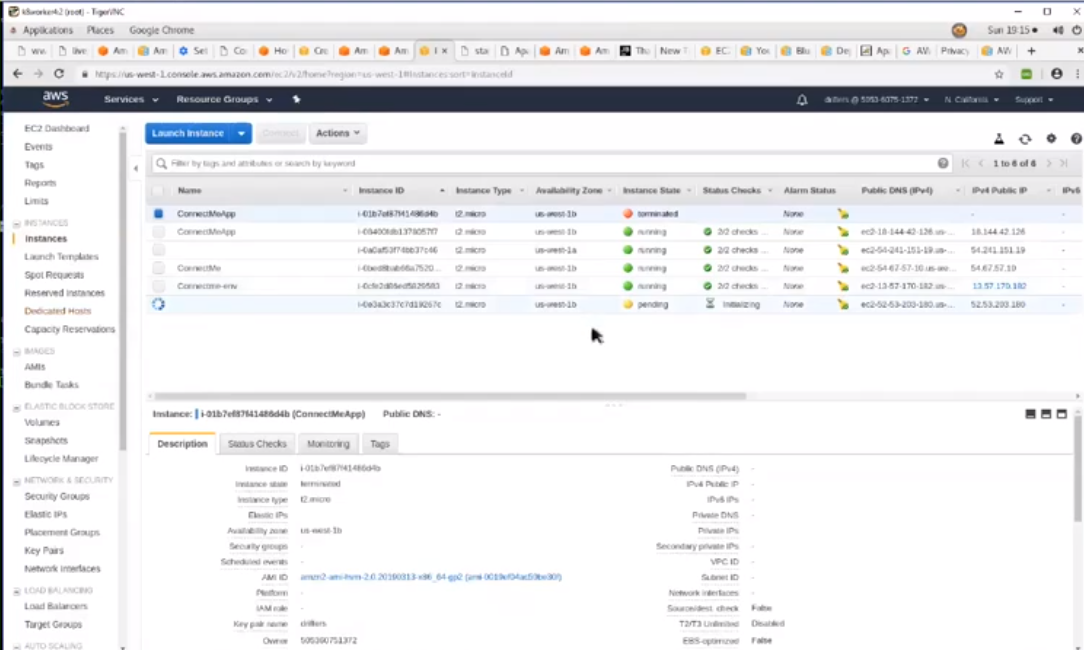
cidr\_ip: 0.0.0.0/0

rules\_egress:

- proto: all

cidr\_ip: 0.0.0.0/0

We can see instance creation on AWS console as shown below:



Once the instance status is changed to running, Open browser and provide the ipaddress to check if the application is up and running.

