**Update the local APT repository**

sudo apt-get update && sudo apt-get -y upgrade

**Check the installed version**

python3 -V

**Install pip which helps in installing packages from PyPi(python package repository)**

sudo apt-get install -y python3-pip

**Check the version of pip**

pip3 -V

**Install virtual environment**

sudo apt install virtualenv

**Check version of virtualenv**

virtualenv --version

**Create virtualenv**

virtualenv -p /usr/bin/python3.8 venvselfyogi

**Activate virtualenv**

source venvselfyogi/bin/activate

**Install requirements.txt**

pip install -r requirements.txt

**Installing Gunicorn**

pip install gunicorn

**Test it by running**

gunicorn selfyogi.wsgi:application --bind 0.0.0.0:8000

**Scripting a bash file for gunicorn**

nano selfyogi\_start.bash

Copy the following script

#!/bin/bash

NAME=“hri”

DJANGODIR=/home/ubuntu/HRI\_yash\_stg\_be

SOCKFILE=/home/ubuntu/venvhri/run/hri.sock

USER=ubuntu

GROUP=ubuntu

NUM\_WORKERS=3

DJANGO\_SETTINGS\_MODULE=hri.settings

DJANGO\_WSGI\_MODULE=hri.wsgi

echo “Starting $NAME as ‘whoami’”

#Activate virtual environment

cd $DJANGODIR

source /home/ubuntu/venvhri/bin/activate

export DJANGO\_SETTINGS\_MODULE=$DJANGO\_SETTINGS\_MODULE

export PYTHONPATH=$DJANGODIR:$PYTHONPATH

#Creating Run directory

RUNDIR=$(dirname $SOCKFILE)

test -d $RUNDIR || mkdir -p $RUNDIR

#Start Django Unicorn

exec gunicorn ${DJANGO\_WSGI\_MODULE}:application \

--name $NAME \

--workers $NUM\_WORKERS \

--user=$USER --group=$GROUP \

--bind=unix:$SOCKFILE \

--log-level=debug \

--log-file=-

**Now make this script executable**

sudo chmod u+x selfyogi\_start.bash

**Run this script now**

./selfyogi\_start.bash

**Install Supervisor**

sudo apt-get install supervisor

**Create a configuration file inside /etc/supervisor/conf.d/**

sudo nano /etc/supervisor/conf.d/selfyogi.conf

Write the following script

[program:selfyogi]

command = /home/ubuntu/selfyogi/selfyogi\_start.bash ;

user = ubuntu ;

stdout\_logfile = /home/ubuntu/logs/gunicorn\_supervisor.log ;

redirect\_stderr = true ;

environment=LANG=en\_US.UTF-8,LC\_ALL=en\_US.UTF-8 ;

**Now that log directory needs to be created**

mkdir -p /home/ubuntu/logs/

touch /home/ubuntu/logs/gunicorn\_supervisor.log

**Now start the supervisor**

sudo systemctl enable supervisor

sudo systemctl restart supervisor

**Check status of supervisor**

sudo supervisorctl status selfyogi

**Install nginx**

sudo apt-get install nginx

**Configure file for nginx**

sudo nano /etc/nginx/sites-available/hri.conf

Paste the following

upstream hri\_server {

server unix:/home/ubuntu/venvhri/run/hri.sock fail\_timeout=0;

}

server {

listen 80;

server\_name prodsybe.hri.com;

client\_max\_body\_size 4G;

access\_log /home/ubuntu/logs/hri/nginx-access.log;

error\_log /home/ubuntu/logs/hri/nginx-error.log;

root /home/ubuntu/hri;

location / {

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header Host $http\_host;

proxy\_redirect off;

proxy\_pass [http://unix:/home/ubuntu/venvhri/run/hri.sock](about:blank);

}

}

**Create the symbolic link for it**

sudo ln -s /etc/nginx/sites-available/hri.conf /etc/nginx/sites-enabled/hri.conf

**Test your configuration with**

sudo nginx -t

**Now start the nginx**

sudo service nginx start

**Creating selfyogi\_server.service at /etc/systemd/system**

cd /etc/systemd/system

sudo nano hri\_server.service

Add the following scripts

[Unit]

Description=hri Server gunicorn daemon

After=network.target

[Service]

User=ubuntu

Group=ubuntu

WorkingDirectory=/home/ubuntu/HRI\_yash\_stg\_be

ExecStart=/home/ubuntu/venvhri/bin/gunicorn --workers 3 --bind unix:/home/ubuntu/venvhri/run/hri.sock hri.wsgi:application

[Install]

WantedBy=multi-user.target

**Reload Daemon**

sudo systemctl daemon-reload

**Restart selfyogi\_server.service**

sudo systemctl restart selfyogi\_server.service

**Restart Supervisor**

sudo supervisorctl restart selfyogi

**Restart nginx**

sudo service nginx restart