**Docker Compose**

Compose is a tool for defining and running multi-container Docker applications. With Compose, you use a Compose file to configure your application’s services. Then, using a single command, you create and start all the services from your configuration.

Using Compose is basically a three-step process.

* Define your app’s environment with a Dockerfile so it can be reproduced anywhere.
* Define the services that make up your app in docker-compose.yml so they can be run together in an isolated environment.
* Lastly, run docker-compose up and Compose will start and run your entire app.

**Demo.**

In this demo I am going to run a python application on top of docker containers using Docker Compose. So let start deep drive.

My project directory contains following four files.

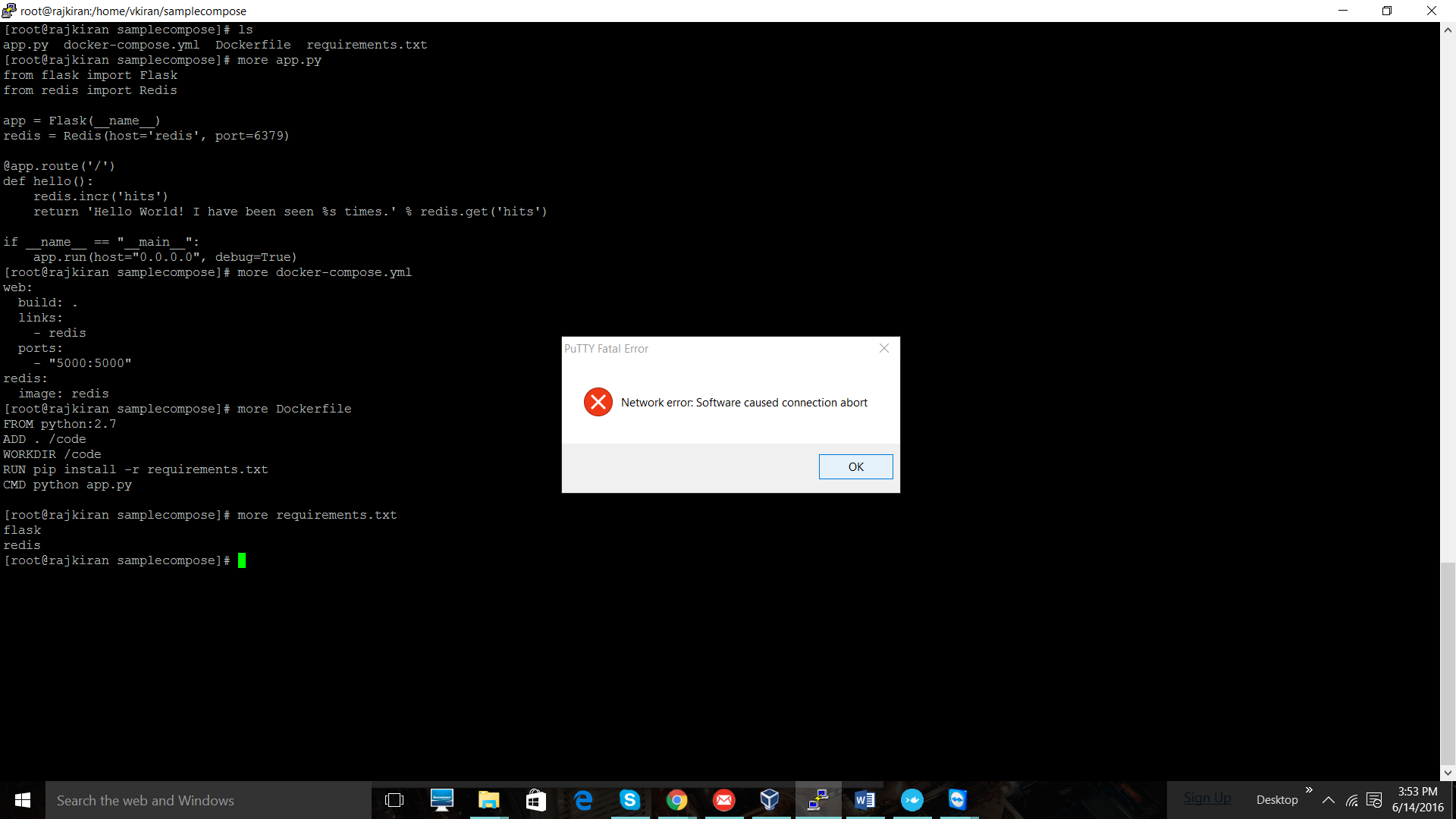


Figure 1 Files inside project directory

Let me show my application code. I stored it in “app.py” file.

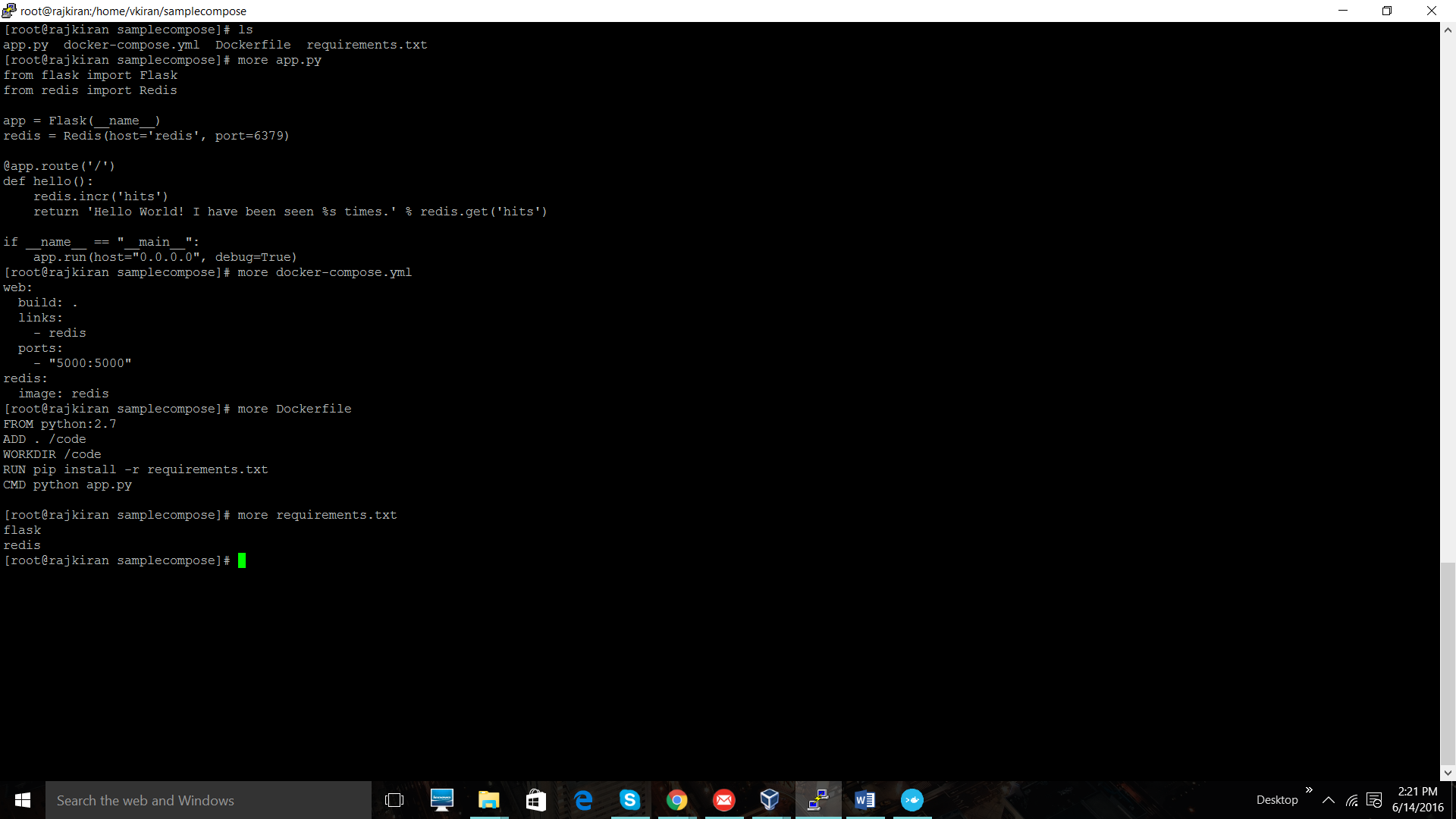


Figure 2 app.py

My application counts number of views and displays onto the webpage. It is a simple, ruby application with “**Hello World**” string and number of hits to that webpage. I store my code as “app.py”. So in-order to run my application, I am using “**Redis**” database and “**Flask**” web framework.

Lets start our three steps of Docker compose. Well, first step is to create a Docker file. Docker file should always named as “**Dockerfile**” , i.e Capital “**D**”. As we know Docker file consist of plain text, that contains instructions to build Docker image. In order to run my application, I need to install python onto my server. So I am going to define it in **Dockerfile**.

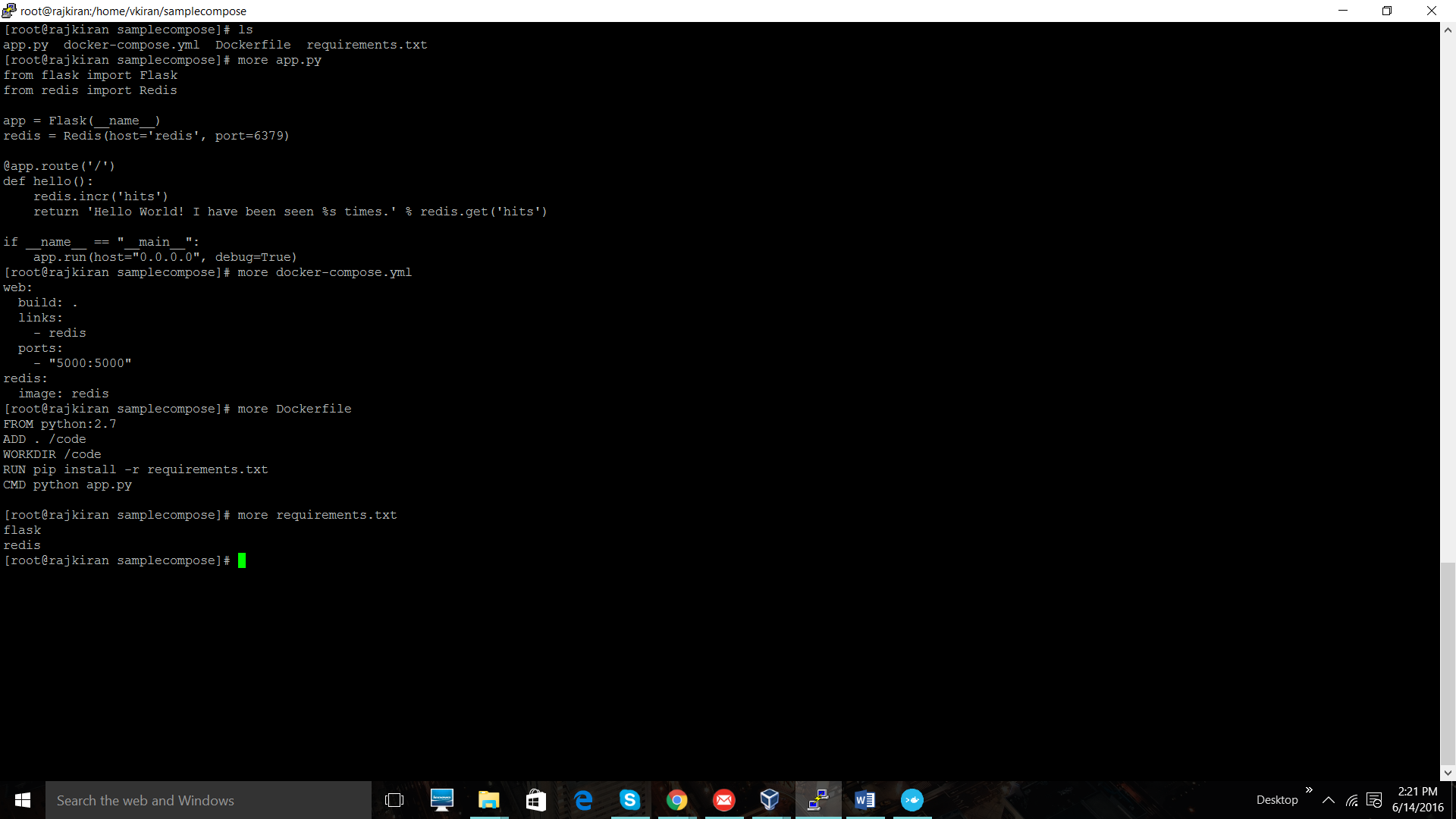


Figure 3Docker file

You can see in above picture that we are using Python:2.7 image. On top of the image we are installing “Redis” and “Flask”, by refereeing **requirements.txt** file.

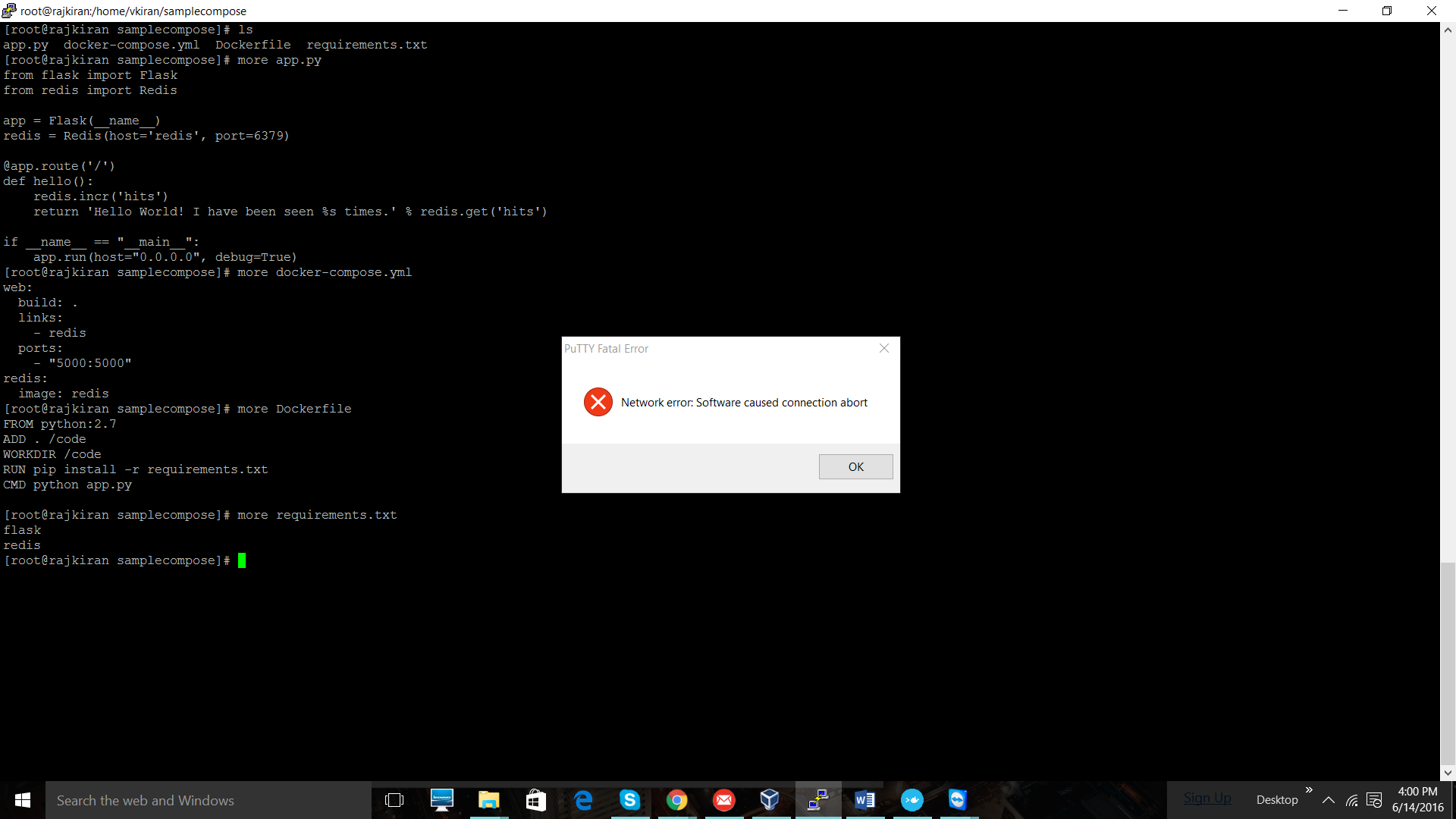
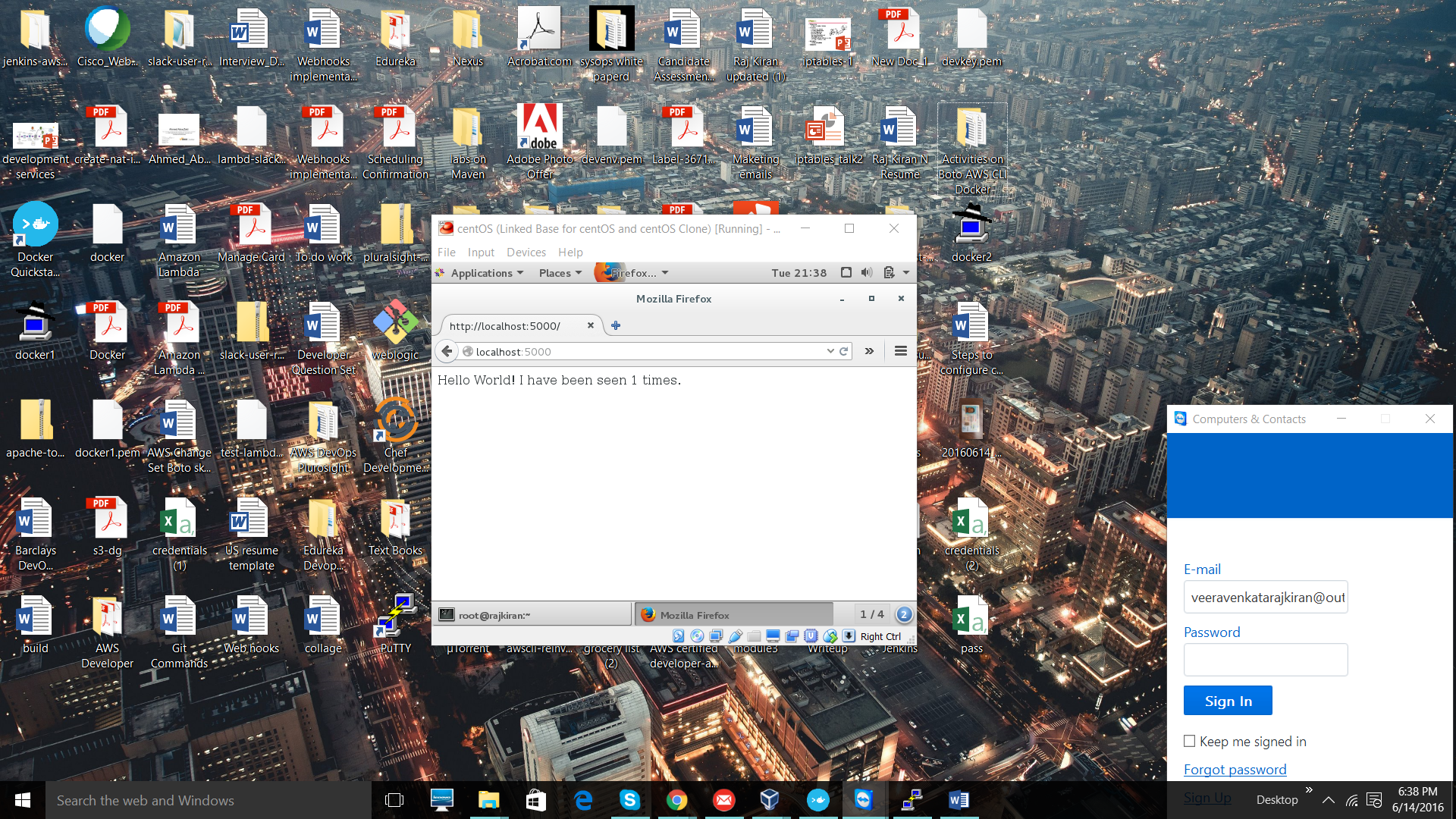


Figure 4 requirements.txt

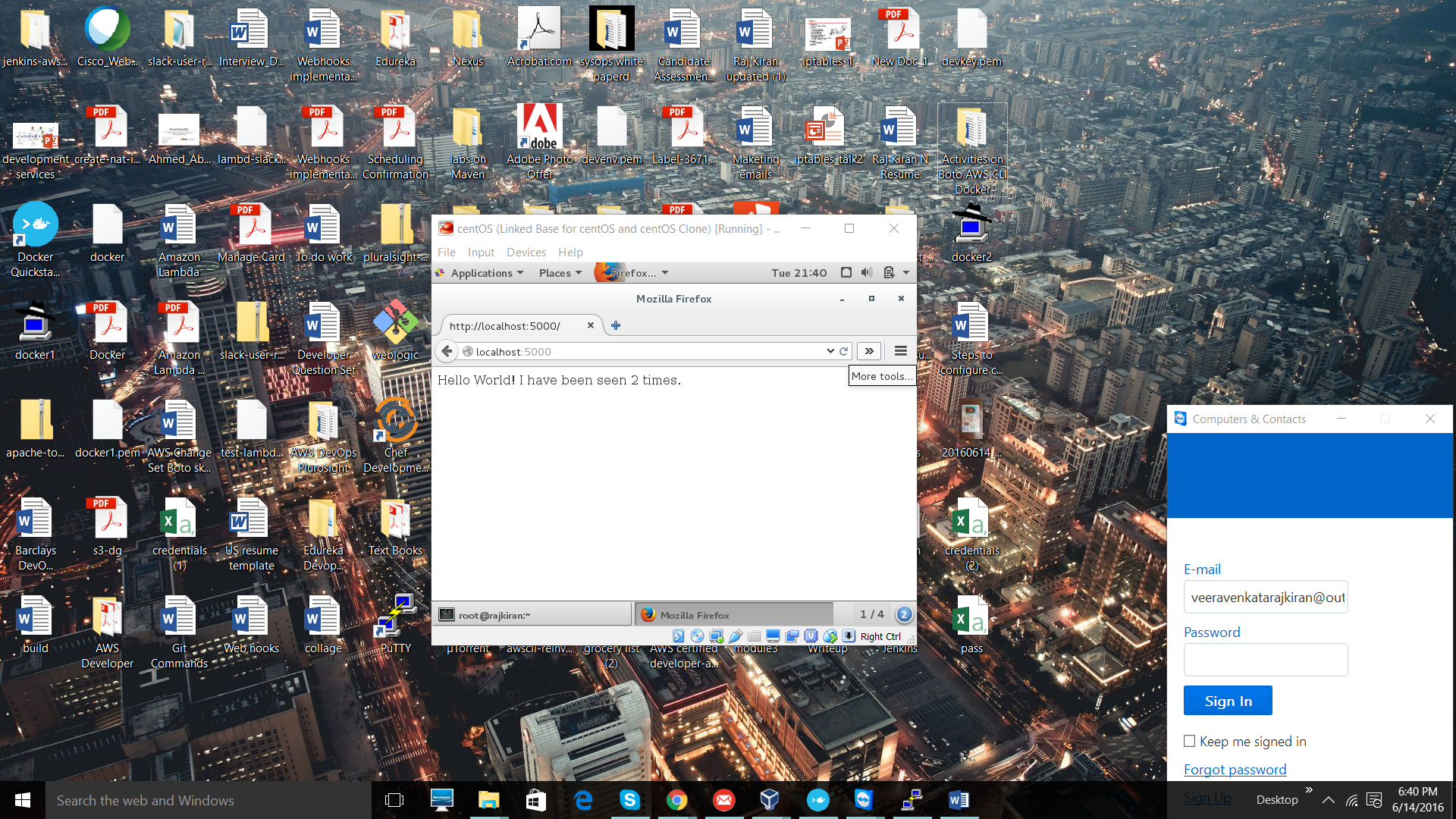
Final step is to run “**docker-compose up**”. By doing so you can deploy your application.



You can check whether you application is up and running. I am running my application on port 5000. So I will go to “localhost:5000”, which gives you following result.



If you referesh the page you should see.



This is a simple example, that shows how you can use docker compose. In this similar passion you can deploy application across cluster of servers using simple docker compose command.