Architecture: To-Do Dashboard
Fog and Cloud Computing 2018/2019

Valentino Armani
valearmani95@gmail.com

Marian Alexandru Diaconu
marianalexandrudiaconu@gmail.com

Introduction
The project aims to distribute a web application over a cloud-based infrastructure using Openstack operating system.

App Scenario
The application is basically the classical To-Do List that allows anonymous users to create, store and retrieve lists of items from a remote database.

App Basic Components
The project is composed of two main components:

- **Front-end** - Angular application whose main role is to allow the user to visualize and modify the data.
- **Back-end** - Node Js (or maybe other) REST APIs that exposes data from database

Implementation
The application will be distributed and implemented in the following way:

- The front-end component will reside on an m1.small flavour instance or similar with about 2GB of RAM on which a linux-based image will be running. This instance will also host an NGINX application that will perform Load Balancing between two equal back-ends. This instance will not need any persistent storage because it will host only stateless applications with respect to data storage, so ephemeral storage should be enough.
- In order to scale performance, the back-end component, as mentioned before, will be duplicated. Both copies will reside on distinct ds1G flavour instances or similar with about 1GB of RAM with the same operating system image as for the front-end case. In this case, the instances will need persistent storage, so a Volume will be attached to them in order to not lose data on server interruption. Moreover, we would also like to replicate the volume in order to increase data availability in case of high traffic or hardware failures.
- In order to protect the back-end from unauthorized access, two Networks will be created. The first will be the one that will contain the front-end compute node and which will proxy public traffic from the Internet to the back-end. The

second will be a network that will contain the back-end compute node. Those networks will be interconnected by a router and the first one will be connected by another router to the public network.

- Security Groups will be created in order to allow traffic from instances to Internet and vice-versa and the same will be done in order to allow communication between instances.
- SSH Key Pairs will be used to secure remote connection to the virtual machines and allow deployment of applications. Moreover, in order to allow a collaborative development in the working group, the key pairs will be securely shared between teammates.