**Deployment of application for Nordcloud.**

Purpose for this project is to create infrastructure to meet business requirements of customer.

Customer has provided detailed requirements:

1. The Application must serve variable amount of traffic. Most users are active during business hours. During big

events and conferences the traffic could be 4 times more than typical.

2. The Customer takes guarantee to preserve your notes up to 3 years and recover it if needed.

3. The Customer ensures continuity in service in case of datacenter failures.

4. The Service must be capable of being migrated to any regions supported by the cloud provider in case of emergency.

5. The Customer is planning to have more than 100 developers to work in this project who want to roll out multiple deployments a day without interruption / downtime.

6. The Customer wants to provision separated environments to support their development process for development, testing, production in the near future.

7. The Customer wants to see relevant metrics and logs from the infrastructure for quality assurance and security

purposes.

To allocate all those needs, we are planning to deploy application with following steps:

1. Register application in catalog. This will allow us to manage and maintain application in faster more reliable way.
2. Define firewall to allow only traffic on port 443. No others would be required.
3. Create Azure load balancer. To allocate customer need increased traffic. Assumption here is that application is scalable and can communicate to at least second node (possibly more).
4. Prepare template for VM with SQLite. As it would be required for allocating all entries which users will create.
5. Deploy 2 VM’s in different Azure regions. This step allows to mitigate two things, first load balancing of workload and possible attacks/disasters hitting Azure DC will not cause interruption of customer application.
6. Create blob as destination for data copy. Assumption here will be that either application can directly write to blob all data which users are bringing or there would be daily export from app/db which can be scheduled to make a copy. Data would remain at REST allowing additional layer of security. No user interaction would be possible as only APPLICATION would be allowed to write to this blob (using service account).
7. Create secondary blob on archive tier. As standard in backup and disaster recovery environment 95% of restores are happening within 6 months. It would be wise to have layering here as well. It would allow to minimize cost for customer as well putting yet another layer of possible data/hack breach.
8. End of deployment.

To allocate future of customers where test and development environments would be required, we could simply use template machines and deploy them in new environments. Possibly machines with lower specs not to impose unnecessary costs.

As per customer access to relevant metrics we can either allow them to see their infrastructure using our own access portal (I can only imagine company have one). If not there’s an alternative solution utilizing Azure Lighthouse whereabouts, we can simply delegate certain rights to users/groups, hence allowing particular people to access monitoring and security center in Azure portal.