# Exam: AzE (2022.02) – 2022.04.09 - Variant #1

### Rules

* Be sure to follow exactly the **IP addresses**, **resource names, file names, etc.** as requested in the tasks
* Tasks execution order should not be derived from the order in which they are listed below
* Тhere are tasks that depend on the successful completion of one or more other tasks
* All resources must be created in **one region** (it can be any, for example ***West Europe***)
* For this exam try to use the smallest and cheapest possible resource option that will do the job
* You are free to setup public IP addresses or other additional options to virtual machines or other resources for debugging purposes
* **The aim is** not to build a production-ready solution but a **working one**

### Goal

A screenshot of a cell phone

Description automatically generated

Supporting files set can be downloaded from [**https://zahariev.pro/q/aze-exam-20220409-variant-1.zip**](https://zahariev.pro/q/aze-exam-20220409-variant-1.zip) or from the **Regular Exam** section of the course site

### Tasks

#### Infrastructure - 5 tasks, 13 pts

* (T101, 1 pts) Create a resource group named **RG-Exam**
* (T102, 2 pts) Create an artefact (availability set or virtual machine scale set) that provides high availability for virtual machines in the front-end group and name it **HA-FE**
* (T103, 2 pts) Create an artefact (availability set or virtual machine scale set) that provides high availability for virtual machines in the back-end group and name it **HA-BE**
* (T104, 4 pts) Create a set of **at least one** **Ubuntu 18.04** (or Ubuntu 20.04) virtual machine(s) in the **front-end group** each with password set as authentication method. If created in availability set, name them **FE-VM-x**, where **x** is a sequence number
* (T105, 4 pts) Create a set of **at least one Ubuntu 18.04** (or Ubuntu 20.04)virtual machine(s) in the **back-end group** each with password set as authentication method. If created in availability set, name them **BE-VM-x**, where **x** is a sequence number

#### Networking - 7 tasks, 17 pts

* (T201, 1 pts) Create virtual network named **VN-Net** with address space **20.0.0.0/16**
* (T202, 1 pts) Create a subnet named **SN-FE** with address space **20.0.0.0/24**
* (T203, 1 pts) Create a subnet named **SN-BE** with address space **20.0.1.0/24**
* (T204, 1 pts) Create a network security group **SG-FE** and attach it to the **SN-FE** subnet and create two **inbound** rules:
  + (T204.1, 1 pts) one to allow communication on port **22/tcp**
  + (T204.2, 1 pts) and a second one to allow communication on port **80/tcp**
* (T205, 1 pts) Create a network security group **SG-BE** and attach it to the **SN-BE** subnet and create two **inbound** rules:
  + (T205.1, 1 pts) one to allow communication on port **22/tcp**
  + (T205.2, 1 pts) and a second one to allow communication on port **9000/tcp**
* (T206, 1 pts) Create an external load balancer named **LB-EXT** with the corresponding set of
  + (T206.1, 1 pts) backend pool
  + (T206.2, 1 pts) health probe
  + (T206.3, 1 pts) and load balancing rule that maps external port **80/tcp** to internal port **80/tcp**
* (T207, 1 pts) Create an internal load balancer named **LB-INT** with the corresponding set of
  + (T207.1, 1 pts) backend pool
  + (T207.2, 1 pts) health probe
  + (T207.3, 1 pts) and load balancing rule that maps external port **9000/tcp** to internal port **9000/tcp**

#### Databases - 3 tasks, 7 pts

* (T301, 3 pts) Create an Azure SQL Database resource (incl. a server)
* (T302, 2 pts) Configure the connectivity to the server
* (T303, 2 pts) Initialize the database with the help of the **sql/load-data.sql** file (part of the supporting files set)

#### Organizational - 4 tasks, 7 pts

* (T401, 2 pts) Apply tag **role** with value **front** or **back** (based on its actual role) on every resource of type virtual machine, availability set, and virtual machine scale set depending on your setup
* (T402, 1 pts) Put a resource lock of type **Delete** on the ***two load balancers*** and the ***two network security groups***
* (T403, 3 pts) Define (if needed) and assign a policy that enforces tag **purpose** with value **exam** on every new or updated resource in the resource group
* (T404, 1 pts) Prove that the policy is working by adding a resource of your choice with name **POLICY** (this can be anything, for example a virtual network, network security group, etc.)

#### Software Deployment - 6 tasks, 16 pts

* (T501, 2 pts) Install **NGINX** on all **front-end** servers. For the configuration you can use the **conf/nginx-sample.conf** file (part of the supporting files set). You are free to modify it or use your own
* (T502, 3 pts) Install **PHP-FPM** on all **back-end** servers. Configure it to listen on port **9000**
* (T503, 4 pts) Install **all supplementary software** on all **back-end** servers to allow them to communicate with the SQL Server database
* (T504, 2 pts) Deploy all **three php files** from the **php** folder (part of the supporting files set) to all back-end servers. You can do this either manually or using the ***Cloud Init*** mechanism
* (T505, 2 pts) Configure all **three php files** by adding a connection string to them. This can be done before the upload or after it
* (T506, 3 pts) Have a fully working web application (*when testing you may need to add index.php to the URL*)