Proposition of azure solution for Drone Shuttles Ltd.

Date: 10.03.2023
Authors: Lukasz Przybyl
Version: 0.1

Client requirements

The customer Drone Shuttles Ltd. is currently running their website on an outdated platform hosted in their own data-center. They are about to launch a new product that will revolutionize the market and want to increase their social media presence with a blogging platform. During their ongoing modernization process, they decided they want to use the Ghost Blog platform for their marketing efforts.

1. **Scalability**:
   1.1. unknown traffic - solution should be able to adapt to traffic spikes,
   1.2. up to 4 times the typical load.
2. **Resiliency**:
   2.1. platform remains online even in case of a significant geographical failure
   2.2. disaster recovery capabilities in case of a region failure
3. **Operations**:
   3.1. Continuous Integration and Delivery*:*
      3.1.1. release new versions of the application multiple times per day, without requiring any downtime
      3.1.2. multiple separated environments to support their development efforts
   3.2. Monitoring*:*
      3.2.1. tools to support their operations and help them with visualizing and debugging the state of the environment
   3.3. Administration:
      3.3.1. server-less function to delete all posts at once
4. **Security**
   4.1. website will be exposed to the internet

Proposed solution

Use docker image of ghost web platform. Based on application complexity level it can be hosted using Azure Web App Service.

To address scalability requirements (Ad. 1) use Web App Service plan that supports scaling out should be used. In case of used MySQL database scalability can be achieved with Azure MySQL Flexible service with enabled auto-grow and IOP auto scaling (currently in preview)

Regarding resiliency requirements (Ad. 2) - especially point no. 2.1- there should be automatic fail-over mechanism in place for all platform crucial component:

- container registry with app images,
- web app service plan + web app,
- database,
- azure storage (static files for web app),
- key vault,
- azure function.

Fail-over could be triggered by Azure Front Door based on defined health-probe settings.

In terms of operations requirements (Ad. 3) there should be three different stage related environments: *test*, *stage* and *production* (Ad 3.1.2). Given environment could be linked with related code branches in git repository in Azure DevOps or similar. Test environment could be used

at early stages of development. Stage environment could use whole or some portion of production data to simulate production-like behavior without affecting actual production. Each environment could be controlled via Azure Pipelines to deploy images, switching web app slots without down time and in general alter configuration i.e. via terraform (Ad. 3.1.1). For monitoring resources and web app Azure Monitoring could be utilized (Ad. 3.2.1). Since Ghost App provides admin API, it could be utilized by Azure Functions with for example http trigger to delete all posts (Ad 3.3.1).

Security of platform could be delivered by:

- Azure Front Door - SSL/TLS certificat, HTTPs enforce, WAF
- Azure Key - vault to store CI/CD secrets
- AzureAD - provide user authentication and authorization with least required access policies assigned to given users/groups
- Internal firewall/VNET settings

Please see below *Figure 1.* for infrastructure visualization.

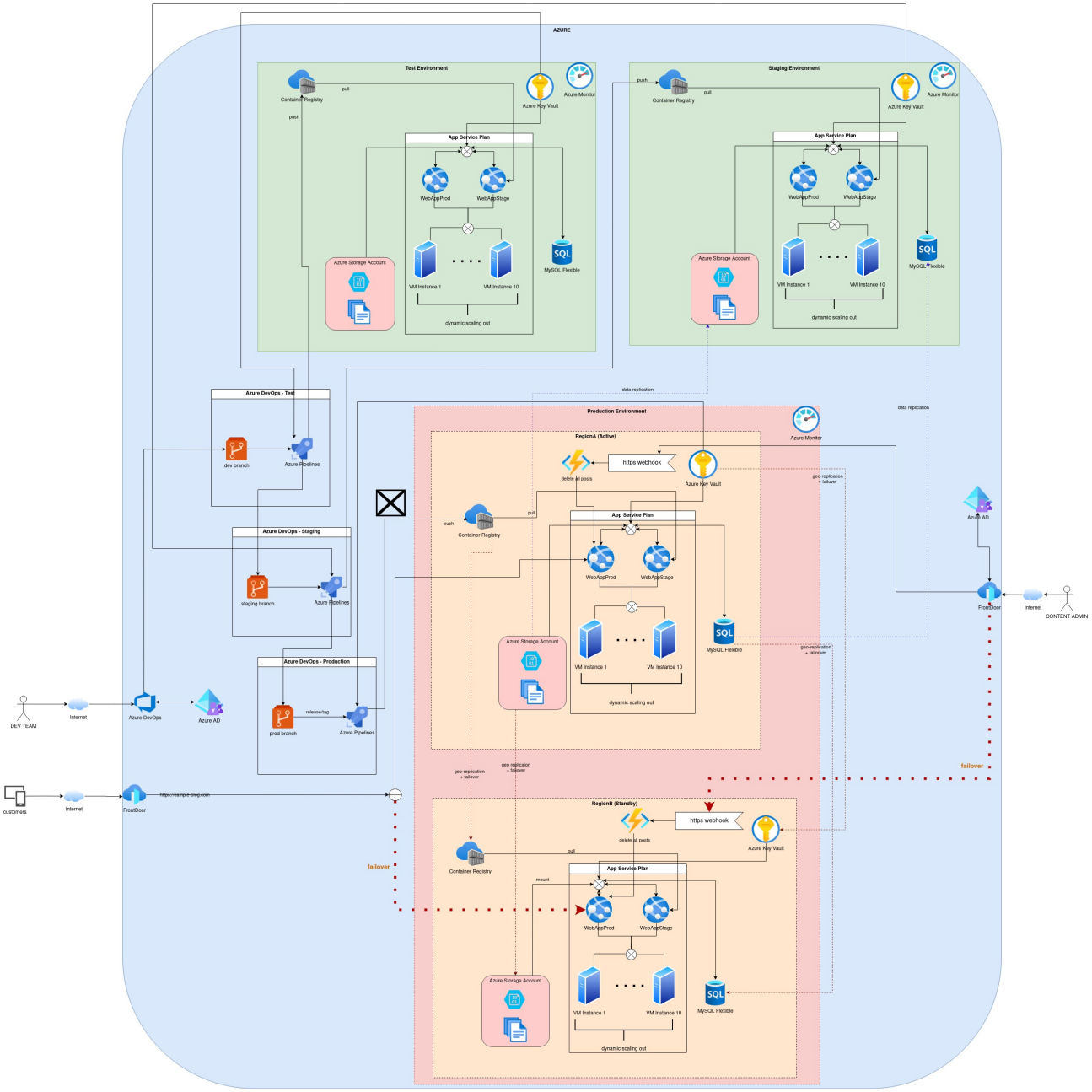

Figure 1: Proposed solution in azure infra