

Gallagher Re

DevOps Engineer - Technical Exercise

Note: This exercise should take you no longer than 3 hours to complete.

Objective: The objective of this task is to ensure provisioning of secure and scalable infrastructure for an internal risk application being developed by the development team. For context, the application's main features will revolve around reading/writing to blob storage, running data transforms, and uploading to a SQL Database. You will design, implement, and document the infrastructure while adhering to best practices.

Requirements:

1. The infrastructure should be hosted entirely on Microsoft Azure.
2. The infrastructure configuration should be in code (Infrastructure-as-Code).
3. The risk application needs to, at least, use the following Azure services:
 - ☐ Azure App Service – UI
 - ☐ Azure Storage Account (Blob) - Blob Storage
 - ☐ Azure Key Vault - Secrets
 - ☐ Azure SQL Database
4. The App Service needs to be able to read/write to storage.
5. The infrastructure should be cost-effective, scalable, and optimized for performance.
6. The infrastructure should use Managed Identity where possible for authentication between services.
7. The infrastructure should have public network access disabled.
8. The application requires two environments – Development and Production.

Tasks:

1. Design the overall architecture for the application with the main features in mind. If anything is unclear, make reasonable assumptions and state them.
2. Identify the Azure services to be used for each aspect of the application – if any additional services (other than those listed above) are used, document reasons why and their benefit.
3. Identify any roles/permissions which are required for the main features to work.

4. Build the Infrastructure-as-Code configuration and pipeline for the application and environments.
5. Provide recommendations for monitoring and troubleshooting the infrastructure.
6. Provide recommendations for how to automatically scale resources in case of high demand.
7. Document how secrets will be managed.

Deliverables

1. High-level architecture design, including diagrams and explanations of the chosen Azure services and their configurations. Diagrams don't need to be complex, simple enough to understand what's going on. See [here](#) for examples.
2. IaC and Pipeline scripts for the application. These aren't expected to be functional or syntactically complete. Main things we want to see from this deliverable are:
 - a. Understanding of requirements and translation into basic architecture
 - b. Understanding of Azure services and dependencies
 - c. Promotion of builds to higher environments
3. Recommendations for monitoring and scaling + how secrets are managed.