**Overview**

Client interacts with the primary services which handles most frequent requests and data. The data is stored within MS SQL Server and Mongo DB. For caching Redis cache is used, for recurring jobs and reminders – Hangfire.

Secondary data is requested via secondary services gateway. Secondary documents service interacts with Mongo DB exclusively, other services work with data from MS SQL Server.

For error monitoring all services utilize error services which report errors to internal monitoring system.

**Primary service**

A monolith app hosted on a server with the functionality regarding entities:

* Users
* Roles
* Skills
* Assessments
* Positions
* Scheduled actions

The app is a REST API that interacts with the fronted via HTTPS protocol. The requests rate should be limited to prevent DDOS attacks (value might vary). The limiting approach – “*fixed window*” (might be changed later). Retry policies will be implemented via Polly for:

* Requests for additional data
* Requests to info database
* Requests to documents database

As *users, skills* and *positions* are most frequently requested entities, they are cached with Redis cache. Reminders and delayed announcements are implemented via Hangfire jobs.

**Secondary service**

Secondary services work with the data which is not used frequently:

* Guides
* old files
* Statistics
* Role specifications
* Rules

Services are hosted in an AKS with the ability of scaling. Additional data is requested from secondary services via secondary services REST API gateway with a load balancer. All services receive requests from gateway and interact with MS SQL Server for text-format data and Mongo DB for binary data.

**Monitoring services**

All services publish error messages to queues and all errors are aggregated in monitoring services. Errors services interact with monitoring services via public gateway which routs requests to the gateway in an internal network. The internal gateway utilizes CORS policies with audience and issuer validations to prevent unauthorized access and rate limiting with “*fixed window*” approach. No retries required here. Communication between internal monitoring gateway and monitoring services is done via HTTP protocol for simplicity of communication.

The access to monitoring data is allowed only from UI hosted in the same private network.