Additional Considerations

Below are the additional considerations that would include in my design to make my application efficient and highly available

* Use **Node Selector constraints** to specify the locations of nodes in the AKS cluster to make sure nodes are located in different zones to improve the availability
* Use **HorizontalPodAutoScaler** to define the average of the metrics that I intend to maintain on each of the pods and then increase/decrease the pod count to maintain those metrics based on the application load.
* Use of **LoadBalancer** to have both the Production and DR clusters in active/active configuration with 100% traffic to Production as default. Would configure the LoadBalancer to route the traffic to DR cluster if the Production cluster is down/unhealthy and revert to Production once it is healthy. The automatic switch significantly reduces the changes of outage when compared to the manual switch-over to the DR instance or in active/passive configuration.
* Automation scripts to run **scheduled database back-ups** during peak hours on a weekly basis.
* Creation of **CI/CD** pipelines in **Azure DevOps** to automate the entire build and deployment process to all the lower environments.
* Usage of **approvals and ServiceNow gates** for Production deployments to automatically check the validity of the Change Request before the deployment to Production.
* Usage of Application Insights and **LogAnalytic Workspaces** to store the application logs and monitor the health of application and add relevant alerting mechanisms.
* Usage of **Azure API Management** instance (external/public facing) to host the API implemented in the web application with enhanced OAuth settings to provide the required security. APIM would re-direct the call to the load balancer after validating the OAuth and the request parameters.
* Utilizing **Azure B2C Active Directory** instance to provide **OAuth** required for authenticating the incoming API request. This instance would be used to create multiple client App Registrations and provide the corresponding clientid/secret to multiple users to secure the application from unauthorized access.
* Utilize the **Azure KeyVaults** to store the passwords/secrets relevant to the application.