Mratyunjay Kaushik

Mrityunjaykaushik.02@gmail.com

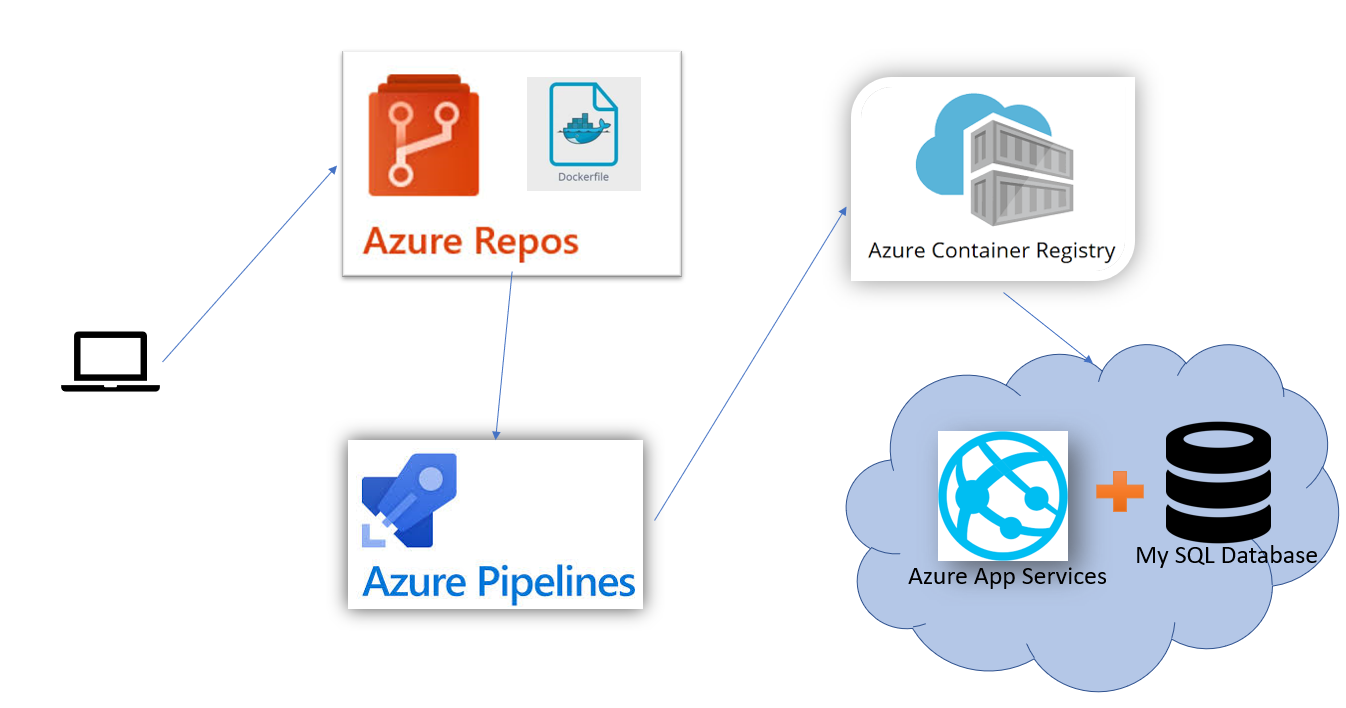
Design Document

For Deployment of Maven Application using Docker Container and Azure Infrastructure

1. **Introduction:** Create a CI/CD pipeline which should be able to Build, test, push Docker image in Azure Container Registry. Also create infrastructure in Azure to deploy the application.
2. **System Overview:** Developers can commit the changes in Azure repo using suitable IDE. As soon as code is changed and pushed into Azure repo in master branch, pipeline will trigger automatically.

Pipeline will be responsible for build, test, create Docker container registry in Azure cloud, push the Docker image in registry and create Azure infrastructures for Dev, Test and Prod environment.

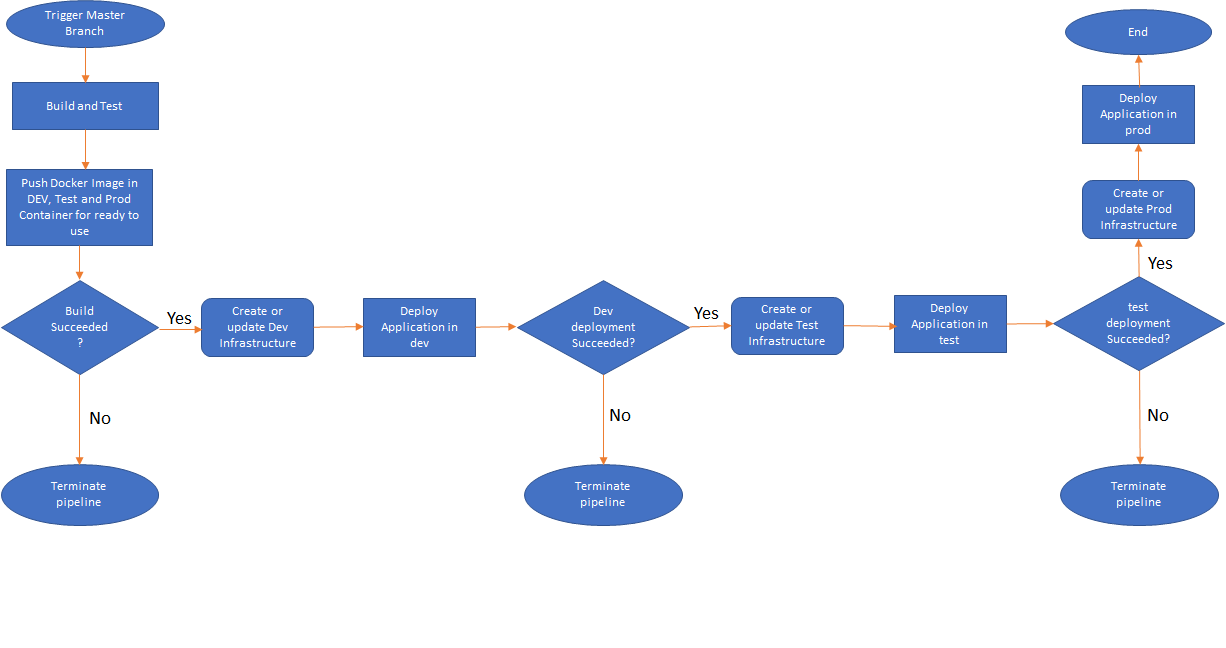
Pipeline will deploy the application in all three environments. We can also configure the approval for pre and post deployment in any environment. We must configure this in YAML file schema.



1. **Configure CI/CD pipeline:**

To configure pipeline for our application a YAML file schema has to prepared. This YAML file will trigger pipeline if any changes happen in master branch.

Using **Ubuntu** because Docker is more compatible with Ubuntu.



1. **Azure Infrastructure:**

To deploy our application in Dev, Test and Prod, we need infrastructures. Below resources are involved for this project:

* 1. **Resource Group:** Group a collection of assets in logical groups for easy or even automatic provisioning, monitoring, and access control, and for more effective management of their costs.
  2. **Container Registry:** ACR (Azure Container Registry) will be used to host the docker image of the spring-boot application. ARM template reference is used inside the pipeline to create the registry for the assignment. *(It is being created or updated through pipeline code.)*
  3. **App Service:** A hosting environment for our Maven application. As soon as application is built and Docker image is pushed in registry, it will automatically be hosted through App service. *(It is being created or updated through pipeline code.)*
  4. **Application Insight:** Application Insights, a feature of Azure Monitor, is an extensible Application Performance Management (APM) service for developers and DevOps professionals. Use it to monitor your live applications. It will automatically detect performance anomalies, and includes powerful analytics tools to help you diagnose issues and to understand what users do with your app. *(It is being created or updated through pipeline code.)*
  5. **SQL Database:** My SQL database will be created for storing application data. *(It is being created or updated through pipeline code.)*

1. **Monitoring and Auditing:**

Application hosted in App Service is subject to certain limits on the resources they can use. There are some features that helps to monitor the resources.

**Alert**: Configure alert rules and attend to fired alerts to efficiently monitor your Azure resources.

**Metric**: Metrics provide information about the app or App Service plan's behaviour. So, Metrics are provided at two levels.

* One is at the App service plan level, which will include CPU and Memory percentage, Data in and out, Disc queue length, and HTTP queue length.
* The second level of metrics provided at an application level, which includes average response time, average memory working set, CPU time, Data in and out, etc. All of these metrics are essential. There are a lot more metrics available when we go for application insight, which is an advanced monitoring tool using which we can monitor web apps.

**Logs**: Log Analytics is a tool in the Azure portal to edit and run log queries from data collected by Azure Monitor Logs and interactively analyse their results. You can use Log Analytics queries to retrieve records matching criteria, identify trends, analyse patterns, and provide a variety of insights into your data.

**Health Check**: Heath check increases your application's availability by removing unhealthy instances.

**Diagnostic settings:** Provide detailed diagnostic and auditing information for Azure resources and the Azure platform they depend on.

**Application Insight** is also created to check performance of your resources. It is a service available in the Microsoft Azure portal that is designed to help development teams get detailed insights into their applications’ availability, performance and usage in near real time. With Application Insights you can view your applications’ telemetry data in one place via user-friendly, interactive graphs.

1. **Scalable Infrastructure:**

Developer can host as many as applications they want, for all three environments. They only need to change the values of variables for Dev, Test and Prod respectively. The separate Infrastructure will be ready automatically.

I am assuming developers are going to use existing Resource Group, Service Connection name and location for all three environments.

