Prerequisites:

- Docker
- k8s Minikube
- DockerHub access (or any other service that allows to store images)
- Azure DevOps access (optional)

The test application consists of following parts:

- Nginx powered file server that responds to each request with a static json file.

- WebApi applications that can work simultaneously and make request to the file server.

Task

Your task is to build and deploy into k8s cluster a distributed web api application.

WebAPI Application

There is a .net core web api (version 3.1.1) application, that should be built in a container and deployed into k8s cluster. The minimum amount of instances to run simultaneously is 2. The source files are in the attachment (TestApi.zip). It's a simple web application that returns a random weather forecast, some simple stats (count of received requests) and a static file from the file server.

Example of the API response:

```
GET https://IP:5001/weatherforecast - ([{"date":"2020-03-28T16:29:25.3487316+01:00","temperatureC":-9,"temperatureF":16,"summary":"Sweltering"},....

GET https://IP:5001/weatherforecast/stats - MachineName – 9

GET https://IP:5001/weatherforecast/fetch - {"version": "1.1.132", "buildId": "2703"}
```

Configuration

By default the web application doesn’t know the URL of the file server. It’s dynamically taken from the environment variable “NGINX”.

File server

Use an official nginx image and update it so that it will return a static file. The file is named static.json and could be found in the zipped folder (TestApi.zip). The file should be accessible by GET request to URL which is nginx container IP address within k8s.

Deployment

Create a deployment for the application into the k8s Minikube.

Result

Send us all scripts, links to containers, files, etc. Include, please, a step-by-step manual how could we launch your solution locally.

(Optional)

If you have an access to Azure DevOps, we would be happy to see the steps from above included into a build pipeline.