Build to Manage - Node.js Observability labs

During this lab we will instrument a simple Node.js application for logging in order to use with log analytics tools like Elastic stack and Humio as well as with metrics for monitoring with Prometheus and Grafana.

Instrumentation of the application code with metrics, logging and tracing is part of the general concept we call **Build to Manage**. It specifies the practice of activities developers can do in order to provide manageability aspects as part of an application release.

Objectives

- Lab 1: Node.js logging with Winston and ELK stack
- Lab 2: Node.js logging with Winston and Humio
- Lab 3: Node.js metrics instrumentation and monitoring with Prometheus and Grafana
- Lab 4: Configure application monitoring with Prometheus on Openshift
- Distributed tracing labs

Prerequisites

Install the following software on your workstation. You may use your laptop for all the labs, but probably a better idea is to use a clean Linux VM.
If you have access and would like to use Fyre, I'd recommend to deploy Ubuntu 20.04 ember with 8 core CPU and 16 GB RAM (for this setup, installation of all prerequisites is as easy as `apt install docker-compose`).

- Docker for Desktop
- Docker Compose
- Openshift `oc` CLI (optional)
- `curl`

Clone the following repository from GitHub.

```
git clone https://github.com/garage-milan/b2m-nodejs-v2
```

Most of the commands should be executed from the `b2m-nodejs-v2/lab-x` directory:

> The solution to the lab is located in the directory b2m-nodejs-v2/lab-x/solution

> Login to Docker Hub using `docker login` in order to avoid the problems with recently introduced limits.