Contact information:

Phone : +380634438318
E-mail: sj.shaynyuk@gmail.com
Skype :Sh.Sergiy
cvPage: https://SergiGan.github.io

SerhiyShaynyuk

DevOps Engineer

Summary of Qualifications

I have less than a year of experience in this position. During this time i improved my CI/CD pipeline skills based on best practices and my mistakes. Now I develop myself in DevOps area, because DevOps a kind of orchestrator of events in a certain environment.

I can work with a stack of technologies such as Docker, Jenkins, GitLab, etc. I have beginner knowledge in writing scripts on Bash and JS, but I will not stop, because I am worried about what I don't know or do not understand. Recently I finished DevOps course and have general knowledge of how Cloud works. I got experience with deploying web-based infrastructure to GCP and AWS clouds.

I am constantly trying to improve and gain new skills, there are a lot of things that I want to improve in our world. I am self-motivated, with good communication skills, ready to work independently and as a team player. I am insistent, attentive to details and self-organization skills, adaptive to changes. Result-oriented.

Skills

Programming Languages/Technologies

- Bash
- JavaScript
- Python (beginner)
- IaaC
- HTML/CSS

RDBMS

- -

CI Tools

- Jenkins
- GitlabCI
- TravisCI

Version Control

- Git/GitHub/GitLab

Methodologies

- Continuous Integration
- Continuous Delivery

Operating Systems

- Ubuntu, Gentoo, Windows OS

Container Orchestration

- Docker

Monitoring

- Prometheus(beginner)

Application/Web Servers

- Nginx

Cloud

- AWS
- Google Cloud

Experience	
	GCP-based platform-CI-CD-pipeline-to GCP / *ChuckNorris-approves*
Project Description:	Infrastructure deployed on GCP cloud platform. This project was located on Linux-server, with an application written in Java named maze-wars that is running on Nginx web server. We have used load balancing in front of the application servers. System health was monitored with the help of monitoring tools, such as Prometheus. And the notification of condition was sent in Telegram
Customer:	Pet project
Involvement Duration:	5 months in 2020
Project Role:	DevOps Engineer
Responsibilities:	▪ Choose cloud provider; ▪ Expand infrastructure by Terraform (IaaC); ▪ Automating application deployment, scaling, and management via Kubernetes; ▪ Setup Cloud Container Registry and use Docker; ▪ Realize CI/CD (Jenkins); ▪ Notification in Telegram; ▪ Configure monitoring& tracking general monitoring functionalities. ▪ Working with other team members on configuration all system.
Project Team Size:	5 team members
Tools & Technologies:	Linux -Ubuntu, nginx, tomcat:9.0-alpine, Terraform, Ansible, Jenkins, Github, docker, gcloud, bash.

	AWS-based platform -The basic workflow with GitHub_Actions/ *github-actions-cicd-to-aws*
Project Description:	Infrastructure was deployed on AWS cloud platform. In this project was used GitHub Action Workflow to Deploy Phyton App to AWS ElasticBeanstalk. We have used load balancing in front of application servers. System health was monitored with the help of monitoring tools, such as Prometheus.
Customer:	Pet project
Involvement Duration:	One week, 2020
Project Role:	DevOps Engineer
Responsibilities:	▪ Getting up acquainted with AWS cloud platform tools and features; ▪ Environment variables declaration ▪ Triggers the workflow for push-pull request events, but only for the main branch; ▪ The type of runner that the job will run on; ▪ Checking out a repository, so this job can access it; ▪ Cloning repository; ▪ Creating zip deployment package from repo; ▪ Configuring AWS credentials for deployment; ▪ Copying Deployment package to S3 bucket; ▪ Creating new ElasticBeanstalk App version; ▪ Deploy EBAv; ▪ CD Pipeline Finished.

Project Team Size:	1 team member
Tools & Technologies:	Linux - Ubuntu, nginx, Github, bash.

Project Description:	**AWS-based platform - The basic continuous delivery pipeline on AWS/** *aws-elastic-beanstalk-express-js-sample*
	Infrastructure was deployed on AWS cloud platform. In this project we will set up a repository for our code so it can be easily accessed over the Internet. In this project we used GitHub, but there are other Git-compatible options which we can be used, including AWS CodeCommit. In one of the following steps we connect this hosted repo to our pipeline, so every time we push a new commit to it - our build process is starting. So we have used AWS CodePipeline to add a review stage with manual approval to our continuous delivery pipeline. At the end we can see that our pipeline with four stages: "Source," "Build," "Review," and "Deploy."
Customer:	Pet project
Involvement Duration:	One week, 2020
Project Role:	DevOps Engineer
Responsibilities:	▪ Getting up acquainted with AWS cloud platform tools and features; ▪ Set up git repo; ▪ Deploy web app; ▪ Create build project; ▪ Create delivery pipeline; ▪ Finalizing pipeline and test.
Project Team Size:	1 team member
Tools & Technologies:	OS - Amazone Linux, nginx, Github, bash, AWS Elastic Beanstalk, AWS CodeBuild, and AWS CodePipeline

Education	Ivan Franko National University of Lviv Faculty of Physics and Electronics, physics specialist