

Ярослав Волков

DevOps Инженер

Лазоревый проезд, д. 1, кв.

58, Москва

+7 (916) 004-73-75

Tg: @Minivolk

volkov.yarolsav0214@gmail.com

ОПЫТ РАБОТЫ

Home Credit Bank, Москва – Старший DevOps Инженер

Январь 2022 – НАСТОЯЩЕЕ ВРЕМЯ

- Разрабатываю скрипты автоматизации процессов настройки и обновления систем (Bash, Python, Groovy и др.)
- Разворачиваю и настраиваю тестовые и продакшн среды
- Создаю CI/CD-пайплайны для 15 команд (команда тестирования, продуктовая команда и т.д.) в Jenkins для стека Java / Spring Framework / npm/ Golang / Django
- Развиваю системы мониторинга и логирования (Prometheus stack)
- Автоматизирую сбор и визуализацию технических и бизнес-метрик систем
- Участвую в траблшутинг-процессах, создаю системные решения для повторяющихся проблем
- Участвую во 2-й линии поддержки систем

FINCH, Москва – DevOps Инженер

Сентябрь 2021 – Декабрь 2022

- Отвечал за создание и поддержку инфраструктуры для проекта РосСтат (обработка и агрегация BigData)
- Разработал микросервисы (отказоустойчивые StateFul приложения) для проекта
- Разработал и успешно внедрил клиентоориентированную инфраструктуру в условиях изолированной сетевой среды заказчика
- Отвечал за поддержку серверного и сетевого оборудования компании
- Проводил обучение в “школе”, организованной компанией
- Успешно обучил 5 новых сотрудников

FINCH, Москва – Python Developer

Ноябрь 2020 – Сентябрь 2021

- Отвечал за разработку микросервисов на основе машинного обучения на Python (OpenCV, FastApi) для проекта оптимизации затрат электроэнергии на складе (заказчик – IKEA)
- Реализовал сбор и обработку Big Data для оптимизации технических работ на складах заказчика

- Отвечал за установку и конфигурацию сетевого оборудования и программного обеспечения для него на складах

ОБРАЗОВАНИЕ

Школа программирования Яндекс, Москва

Май 2016 – Май 2019

Итоговая работа по теме “Сервис для бронирования билетов PkCinema на базе фреймворка Django с использованием системы управления базами данных MySQL”

Навыки

Архитектура: Микросервисная, Монолитная

Паттерны и практики разработки: Интеграционные (EIP), Микросервисные (MSA), GoF, GRASP, SOLID

Фреймворки: Fastapi, Django, DjangoRestApi, Flask

Конфигурация: Spring Cloud Config, Ansible, Vault

Системы контроля версий: git, gitflow

Сборка проектов: maven, make, cmake

Аутентификация и авторизация: SSO, OAuth 2.0, Keycloak, JWT, OpenID, LDAP & Active Directory

CI/CD конвейеры: Gitlab, Jenkins

Базы данных SQL: PostgreSQL, MySQL

Базы данных NO-SQL: Redis, MongoDB, Elasticsearch

Методологии разработки: Agile, Scrum, Kanban

Мониторинг: Zabbix, Grafana, Prometheus, VictoriaMetrics

Тестирование: UnitTest

Управление проектом: Jira, Confluence

Протоколы и интеграционные подходы: http/s, tcp, udp, REST, RESTful API, gRPC, WebServices

Сериализация и десериализация: JSON, Protobuf

Технологии контейнеризации: Docker, Podman, docker-compose, containerd

Оркестраторы: Kubernetes, OpenShift

Облака: Yandex Cloud

Логирование: Loki, Vector, Graylog

Брокеры сообщений: RabbitMQ

ETL & Stream processing: Dagster

BI: Superset

API management: Swagger, OpenAPI

Операционные системы: macOS, Windows, Linux

Вспомогательные языки программирования: Bash, SQL, Groovy, Python, JavaScript, C++, Scala

Анализ кода: Sonarqube

Методологии: Объектно-ориентированное программирование (ООП), Функциональное, Процедурное

