

# Артемий Поздняков

| Санкт-Петербург, Россия | +7 (922) 249-69-44 | [aapozdnyakove@gmail.com](mailto:aapozdnyakove@gmail.com) | [github.com/aaletov](https://github.com/aaletov)  
| [gitlab.com/aapozd](https://gitlab.com/aapozd) | [linkedin.com/in/artemy-pozdnyakov-b7972b233/](https://linkedin.com/in/artemy-pozdnyakov-b7972b233/) | [t.me/aaletov](https://t.me/aaletov)

## О себе

Люблю находить новые решения к привычным задачам. Я имею опыт в бэкенд разработке и в настоящее время развиваюсь в области машинного обучения и анализа данных

## Образование

**Санкт-Петербургский политехнический университет им. Петра Великого** 2020-2024  
Бакалавр, Программная инженерия

## Опыт

**Лаборатория Dell Technologies - ВШПИ СПбПУ** Февраль 2022 – Август 2022, Санкт-Петербург  
Стажёр (Go)

Участвовал в разработке open-source проекта компании Dell - csi-baremetal, CSI-драйвера для Kubernetes, предназначенного для управления физическими дисками в кластере

- Разработал тестовую реализацию gRPC-сервиса на Go для поддержки Storage Capacity в драйвере, реализация была протестирована в окружении Kind
- Разработал proposal с планом реализации поддержки Storage Capacity в проекте для снижения нагрузки на использовавшийся кастомный планировщик

**Linxdatacenter** Сентябрь 2022 – Июнь 2023, Санкт-Петербург  
Младший разработчик бэкенда (Go)

Участвовал в разработке бэкенда клиентского портала компании Linxdatacenter

- Переработал модуль аутентификации пользователей, что позволило в несколько раз сократить объем кода обработки JWT-токенов, ответственного за ограничение доступа к методам API
- Разработал микросервис для доступа к истории посещения ЦОД
- Реализовал с нуля BDD-тесты для основного микросервиса, в процессе были исправлены и приведены в соответствие между собой OpenAPI-спецификации различных микросервисов, что позволило согласовать логику данных сервисов и фронтенда

## Проекты

**Классификатор болезней растений** Февраль 2024 – Апрель 2024  
Классификатор болезней томатов на основе свёрточной нейронной сети. Курсовая работа по курсу "Глубокое обучение" в СПбПУ

- Построил нейронную сеть с помощью Tensorflow Keras, используется комбинация свёрточной нейронной сети и многослойного перцептрона с признаками Харалика в качестве входа
- В результате работы было установлено, что точность модели (0.84) при совместном использовании подходов на основе CNN и обычной классификации в линейно неразделимой выборке с помощью MLP, оказывается выше, чем при использовании данных моделей по отдельности

**Анализ стационарных решений мат. модели** Февраль 2022 – Май 2022  
Программа для анализа динамической математической модели, заданной системой дифференциальных уравнений

- Получены стационарные точки модели, исследована зависимость стационарных решений от заданного параметра
- Определены точки вещественной и комплексной бифуркации для различных параметров модели
- Библиотека numpy использовалась для вычислительной части работы - вычисления собственных значений матрицы Якоби системы и нахождения корней уравнения, задающего стационарные точки

## План реализации Storage Capacity в драйвере csi-baremetal

Март 2022 – Август 2022

## Моделирование СМО

Сентябрь 2022 – Ноябрь 2022

Модель системы массового обслуживания с заданными дисциплинами отказа и постановки на обслуживание

- Реализовал модель СМО, REST API сервис (Golang) на основе модели, а также фронтенд на React

## Система автоматизации работы больницы

Февраль 2022 – Май 2022

Курсовая работа по курсу "Объектно-ориентированное программирование" в СПбПУ

- Разработал веб-приложение на языке Java, позволяющее автоматизировать работу больницы. Бэкенд написан с использованием паттерна MVC и фреймворка Spring, фронтенд использует фреймворк Vaadin

## Технические навыки

---

**Back-end development:** Microservices, REST, OpenAPI, gRPC, JWT, OAuth, Git

**ML + Analytics:** Sklearn, Tensorflow, Keras, pandas, numpy, scipy, matplotlib

**Database Management:** SQL, PostgreSQL, Redis

**Testing:** TDD, BDD, Unit testing

**Teamwork:** Agile, Scrum, Confluence, Jira

**Programming languages:** Python, Go, C++, Bash, LaTeX