

АНДРЕЙ САВИН

Проживание: Россия, Москва

4andreysavin@gmail.com | [@W3RR3R](https://t.me/W3RR3R) | [Хабр Карьера](#) | [GitHub](#)

ЛИЧНАЯ ИНФОРМАЦИЯ

Возраст

19 лет

Проживание

Россия, Москва

ПРОФЕССИОНАЛЬНЫЕ НАВЫКИ

Языки программирования: SQL, Golang, Java, Python, C++

Технологии: Docker, Git, RabbitMQ, Linux, Docker-compose,

Паттерны: SOLID, Clean Architecture, Dependency Injection, Worker Pool

ВЫСШЕЕ ОБРАЗОВАНИЕ

Российский технологический университет МИРЭА

Москва, Россия

Институт информационных технологий, Программная инженерия

Сентябрь 2022 – По настоящее время

СПЕЦИАЛИЗАЦИЯ И ДОСТИЖЕНИЯ

Проекты, разработанные совместно с Киберзона РТУ МИРЭА

- **1 Разработка микросервиса фильмотеки**
- Роль: Бек-энд разработчик в команде
- GitHub репозиторий: FilmotekaService
- Технологии: Docker-compose, RabbitMQ, Паттерны микросервисной архитектуры
- Основные достижения:
 - * Организовал оркестрацию контейнеров на базе Docker-compose.
 - * Построил микросервисную архитектуру по паттерну Service Bus.
 - * Настроил и интегрировал брокер сообщений RabbitMQ.
- **1.1 Разработка CRUD REST API фильмотеки**
- GitHub репозиторий: restApiFilmoteka
- Технологии: net/http, gorilla/sessions, нормализация БД, сидинг данных, миграции БД
- Основные достижения:
 - * Реализовал маршрутизатор с использованием стандартной библиотеки net/http.
 - * Разработал механизм авторизации на основе сессионных куки, что позволило улучшить безопасность и удобство использования системы.
 - * Создал универсальный интерфейс для подключения различных типов баз данных.
 - * Организовал CRUD функционал для полноценной работы сервиса.
 - * Применил принципы SOLID, что улучшило модульность и поддерживаемость кода.
 - * Настроил CI проекта с помощью Docker и Docker-compose.
 - * Разработал легковесный контейнер под проекты на Golang, основанный на образе Ubuntu.
- **1.2 Разработка почтового сервиса рассылки писем зарегистрировавшимся пользователям**
- Технологии: Clean Architecture, Worker Pool, net/smtp, gmail, rabbitmq/amqp091-go, goroutines, примитивы синхронизации
- Основные достижения:
 - * Разработал сервис с четким разделением на слои (используя Clean Architecture), что повысило модульность, тестируемость и поддерживаемость кода.
 - * Организовал структуру пула рабочих потоков для обработки задач отправки писем.
 - * Реализовал механизмы отправки писем через SMTP с помощью стандартной библиотеки net/smtp и внешней библиотеки gmail.
 - * Внедрил RabbitMQ для обработки задач отправки писем в асинхронном режиме.
 - * Использовал горутины для параллельной обработки задач преобразования входящих сообщений и отправки писем, что значительно увеличило скорость выполнения операций.
 - * Предотвратил появление состояний гонки в проекте с помощью примитивов синхронизации.
- **2 Разработка системы организации турниров для ВУЗа**
- Роль: Разработчик в команде

- Технологии: Асинхронное программирование, mockery (Mock тестирование)
- Основные достижения:
 - * Приобрёл опыт командной разработки с использованием GitLab.
 - * Разобрал исходный код библиотеки для взаимодействия со Steam API, и на основе полученных знаний разработал функционал двухэтапной авторизации.
 - * Приобрел опыт асинхронной разработки и работы с событийно-ориентированной архитектурой, что позволило обрабатывать большие потоки данных в реальном времени.
 - * Разработал асинхронный планировщик матчей с применением паттернов Dependency Injection и немного модифицированного Worker Pool.
 - * Изучил и применил примитивы синхронизации, такие как RWMutex, каналы и Wait-группы.
 - * Провёл Unit-тестирование с использованием библиотеки vektra/mockery.

УЧЕБНЫЙ ОПЫТ И ПЕТ ПРОЕКТЫ

Разработка RESTApi-сервиса | *Go, PostgreSQL, Docker, gorilla, logrus*

- Разработка многофункционального RESTApi-сервиса, поддерживающего модульную архитектуру и возможность интеграции с различными базами данных.
- Имплементация системы сессионной авторизации и механизма логирования для обеспечения безопасности.
- Автоматизация процессов развертывания приложений с использованием Docker.
- Реализация интеграционного тестирования методов взаимодействия с БД.
- Автоматизация системы тестирования проекта с помощью встроенных библиотек.

Бэкграунд разработки на других языках

- Консольные приложения и решения алгоритмических задач на C/C++ и Java (1,5 года).
- Создание новостного веб сайта на Python с использованием фреймворка Flask и БД SQLite.
- Разработка мобильного приложения на языке Java.

ДОПОЛНИТЕЛЬНЫЕ НАВЫКИ

- Уверенное владение и понимание внутреннего устройства языка Golang.
- Уверенное использование операционных Unix систем (WSL, Virtual Box, основная ОС).
- Уверенное владение SQL запросами и опыт работы с реляционными базами данных.
- Уверенное написание Docker образов и docker-compose файлов.
- Знание популярных паттернов, таких как Clean Architecture, SOLID, Dependency Injection, Singleton, Worker Pool.

О СЕБЕ

На данный момент являюсь студентом третьего курса специальности "Системная и программная инженерия". Занимаюсь бек-энд разработкой на языке Golang. Я - коммуникабельный и веселый программист с спокойным характером, постоянно изучающий новые технологии в своей сфере и стремящийся применить их на практике. Мои навыки в установлении контактов помогают мне легко находить общий язык и успешно вливаться в новые коллективы. Усидчивость и стремление к нахождению лучших решений являются неотъемлемой частью моего подхода к работе. Мой энтузиазм и собственное виденье задач позволяют мне приносить ценные идеи в рабочей среде. Я готов к диалогу, открыт для обсуждения и всегда готов выслушать мнение коллег. Мои принципы гибки, и я готов принять обоснованные идеи других, если они более эффективны.