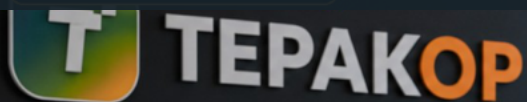


ТЕРАКОР • ПРОФЕССИЯ В IT



ОБУЧЕНИЕ. РАЗРАБОТКА. РЕЗУЛЬТАТ.



ПОДРОБНАЯ ПРОГРАММА ОБУЧЕНИЯ

Профессия C# / .NET-разработчик

От C# и ASP.NET Core до Kafka и микросервисов

C# • ASP.NET Core • EF Core • PostgreSQL • Kafka • Docker

Практика на проектах для портфолио

После каждого модуля - законченная работа и code review.

5 месяцев

полный маршрут

456 ак. ч.

общая нагрузка

152 ак. ч.

живые занятия

КАФКА - 8 АК. Ч.

4 ПРОЕКТА + ИНТЕРВЬЮ

Преподаватель

Сергей Валерьевич Герасименко

Основатель ТЕРАКОР • практикующий full-stack разработчик

90 000 ₽

полная программа

terakor.ru

Лицензированный учебный центр • обучение онлайн

1. О профессии

Профессия C# / .NET-разработчик - последовательная подготовка от основ C# и PostgreSQL до ASP.NET Core, микросервисов и Apache Kafka. Программа сохраняет структуру четырех разделов: после каждого студент выполняет проект, получает code review и дорабатывает результат для портфолио.

Формат: онлайн-занятия с преподавателем в группе до 8 человек, самостоятельная практика и защита итоговой платформы OrderFlow. Цель программы - подготовка к задачам .NET backend и техническому интервью. Грейд зависит также от опыта коммерческой разработки.

2. Результаты обучения

- Писать C#-код, работать с коллекциями, LINQ, исключениями, файлами и JSON.
- Применять ООП, SOLID, паттерны GoF, dependency injection и async/await.
- Проектировать схему PostgreSQL и работать с данными через EF Core.
- Разрабатывать и защищать ASP.NET Core Web API.
- Проверять код unit- и интеграционными тестами и запускать систему в Docker Compose.
- Строить синхронное и событийное взаимодействие сервисов, обрабатывать сбои и дубли.
- Защищать архитектуру OrderFlow и отвечать на вопросы технического собеседования.

3. Нагрузка и формат

Раздел	Живые занятия	Самостоятельная практика
I. C# и основы .NET	48 ак. ч.	88 ак. ч.
II. ООП и платформа .NET	46 ак. ч.	88 ак. ч.
III. ASP.NET Core и production	42 ак. ч.	80 ак. ч.
IV. Kafka и паттерны микросервисов	16 ак. ч.	48 ак. ч.
ИТОГО	152 ак. ч.	304 ак. ч.

Всего - 456 академических часов. Ориентировочный срок полного маршрута - 5 месяцев. Подготовка к техническому собеседованию: 4 ак. ч. внутри 42 часов раздела III; занятия проходят после защиты OrderFlow. Эти часы не прибавляются к итогу.

4. Подробная программа

Раздел I. C# и основы .NET - 48 ак. ч.

Результат: студент пишет приложения на C#, использует Git и PostgreSQL, проверяет основные сценарии тестами.

- Алгоритмы и сложность в прикладных задачах; установка .NET SDK, Visual Studio / Rider, отладка.
- C# 12 и .NET 8: типы значений и ссылки, nullable, операторы, условия, циклы.
- Массивы, List, Dictionary, строки, StringBuilder, методы и параметры.
- Делегаты и события; основы LINQ to Objects.
- Исключения, работа с файлами, потоками и System.Text.Json.
- Введение в классы, конструкторы, свойства и слои приложения.
- PostgreSQL: DDL/DML, JOIN, ограничения, транзакции и ACID.
- Git: ветки, pull request; xUnit/NUnit и Moq - введение.

ПРОЕКТ ПОСЛЕ МОДУЛЯ: StockDesk CLI

Консольная система учета товаров и движения остатков на C# с PostgreSQL.

Для портфолио: код, схема БД, обработка ошибок, тесты, демонстрационные данные и README с запуском.

Контроль: промежуточная защита StockDesk CLI и разбор ошибок на code review.

Раздел II. ООП и платформа .NET - 46 ак. ч.

Результат: студент проектирует расширяемое ядро приложения и аргументирует выбор подходов.

- Наследование, полиморфизм, интерфейсы, abstract class, record types.
- Generics, constraints, covariance/contravariance и производительность коллекций.
- async/await, Task, отмена операций; потоки, lock и concurrent collections.
- SOLID и рефакторинг кода с тестами.
- GoF: Factory Method, Strategy, Observer; Singleton и его ограничения.
- Встроенный DI-контейнер .NET; время жизни зависимостей.
- LINQ, IQueryable и обзор expression trees.
- HttpClient, REST-клиент, сериализация; ILogger и Serilog.
- HTTP и основы веба для backend-разработчика.

ПРОЕКТ ПОСЛЕ МОДУЛЯ: Order Rules Engine

Библиотека правил обработки заказов: разные стратегии, внешние адаптеры и асинхронные вызовы.

Для портфолио: предметная модель, SOLID, паттерны GoF, DI, тесты и описание решений до/после рефакторинга.

Контроль: промежуточная защита библиотеки с объяснением компромиссов архитектуры.

Раздел III. ASP.NET Core и production - 42 ак. ч.

Результат: работающий ASP.NET Core API с БД, авторизацией, тестами и воспроизводимым запуском.

- ASP.NET Core: hosting, конфигурация и middleware pipeline.
- Контроллеры и Minimal API, маршрутизация, HTTP-коды, OpenAPI.
- EF Core: Code First, связи, миграции, транзакции и границы Repository/Unit of Work.
- ASP.NET Core Identity, JWT, авторизация и защита маршрутов.
- Валидация, FluentValidation и Problem Details.
- Unit-тесты и интеграционные тесты через WebApplicationFactory.
- Кэширование IMemoryCache; Redis - обзор.
- Dockerfile, Docker Compose, PostgreSQL, переменные окружения, health checks.
- gRPC и SignalR - обзор применимости.

ПРОЕКТ ПОСЛЕ МОДУЛЯ: SupportDesk API

API обращений: создание заявки, назначение исполнителя, изменение статуса, поиск и история действий.

Для портфолио: ASP.NET Core Web API, EF Core, миграции, JWT, тесты, Swagger/OpenAPI и Docker Compose.

Подготовка к интервью: 4 ак. ч. включены в часы этого раздела, но проходят по расписанию после защиты итогового OrderFlow. Содержание: C#/ООП/async; SQL/ASP.NET Core/EF; live coding; пробное интервью и обратная связь.

Раздел IV. Kafka и паттерны микросервисов - 16 ак. ч.

Результат: студент строит событийный сценарий, диагностирует задержки и повторы, объясняет согласованность данных между сервисами.

Подраздел	Часы	Темы и практика
Apache Kafka	8	Топики, партиции, ключи, consumer groups и offsets. Confluent.Kafka producer/consumer, BackgroundService, retry, DLQ, идиоматичная обработка.
Паттерны и запуск	8	Границы сервисов, Docker Compose, синхронные вызовы, timeout; Outbox, Saga, Circuit Breaker с Polly. API Gateway и CQRS - обзор.

Восемь часов Kafka дают практическую интеграцию; глубокие темы администрирования, Schema Registry и Streams не заявляются как полностью освоенные в этом блоке.

ПРОЕКТ ПОСЛЕ МОДУЛЯ: OrderFlow Platform

Итоговая событийная платформа заказов на ASP.NET Core: сервис заказов, обработчик событий и уведомления.

Для портфолио: Web API, EF Core, PostgreSQL, Kafka, Outbox, Docker Compose, тесты, схема событий и демонстрация восстановления после сбоя.

5. Проекты в портфолио

После раздела	Проект	Что проверяется
I	StockDesk CLI	C#, SQL, Git, основы тестирования.
II	Order Rules Engine	ООП, SOLID, GoF, DI, async.
III	SupportDesk API	ASP.NET Core, EF Core, авторизация, интеграционные тесты.
IV - итог	OrderFlow Platform	Сервисы, Kafka, Outbox, Docker и сценарий сбоя.

У каждого проекта есть репозиторий, README, инструкция запуска, тесты и краткое описание решений. После каждого раздела - code review и исправление замечаний. Проекты выполняются преимущественно в часах самостоятельной работы, указанных в разделе 3.

6. Итоговая работа OrderFlow Platform

Платформа из трех компонентов: Order API принимает и хранит заказы; обработчик читает события Kafka и обновляет процессы; сервис уведомлений реагирует на смену статуса. Отдельные компоненты могут запускаться как сервисы ASP.NET Core или фоновые worker-процессы .NET по назначению.

- Order API: REST, JWT, PostgreSQL, EF Core и миграции.
- Kafka: топики и ключи по идентификатору заказа, producer/consumer и consumer group.
- Outbox: согласованная публикация событий после изменения данных.
- Повторная обработка, защита от дублей, retry и DLQ.
- Docker Compose, интеграционные тесты и базовое логирование.
- Защита: нормальный сценарий заказа и один воспроизводимый сценарий сбоя.

Для портфолио: схема сервисов и событий, README, демоданные, API-примеры, тесты и короткое видео. Техническую глубину реализации студент обосновывает на защите.

7. Техническое собеседование - 4 ак. ч.

Час	Содержание
1	C#, ООП, SOLID, generics, LINQ и async/await.
2	SQL, ASP.NET Core, EF Core, безопасность, Kafka и паттерны.
3	Live coding, чтение и исправление чужого кода.
4	Пробное интервью по OrderFlow, обратная связь и план развития.

Эти 4 ак. ч. входят в 42 ак. ч. раздела III и в общий итог 152 ак. ч. Их проводят в конце полного маршрута, после защиты проекта. Прохождение интервью не гарантирует трудоустройство или присвоение грейда.

8. Для кого программа

- Начинающим: полный маршрут от C# и базы данных до законченной backend-системы.
- Разработчикам с опытом C#: систематизация ASP.NET Core, EF Core, Kafka и архитектуры.
- Специалистам из Java/Python: переход на .NET и практика в новой экосистеме.

Если часть тем уже освоена, точку входа можно определить после теста и разговора о предыдущем опыте. Условия сокращенного маршрута уточняются отдельно.

9. Формат и преподаватель

Живые онлайн-занятия, небольшая группа до 8 человек, самостоятельные задания, code review и защита проектов. Сергей Валерьевич Герасименко - практикующий full-stack разработчик, технический директор и основатель ТЕРАКОР и ТЕРАКОР Digital. На занятиях разбирает инженерные подходы и объясняет решения на практических примерах.

10. Стоимость и следующий шаг

Полный маршрут	Стоимость
----------------	-----------

5 месяцев; 456 ак. ч. общей нагрузки; 152 ак. ч. занятий

90 000 ₽

Программа включает четыре проекта, code review, итоговую защиту и подготовку к техническому собеседованию.

Актуальные условия оплаты и даты набора проверяйте на странице профессии:

terakor.ru/catalog/dotnet/dotnet-profession/.

Пройдите бесплатный тест по C# / .NET или обсудите с преподавателем точку входа, если уже знаете часть программы.