otus
онлайн-образование

Полная программа

C#-разработчик. Продвинутый уровень

Best Practice по разработке на C# и .NET с практикой Scrum-методики

Длительность курса: 132 часа

Модуль 1. Архитектура проекта и Базы данных

Тема 1

Знакомство, рассказ о формате Scrum, краткий обзор курса // ДЗ

Цель занятия

познакомиться с курсом и командой.

Краткое содержание

знакомство с командой курса;
знакомство со студентам;
цели курса;
формат Scrum-команд;
краткий обзор модулей.

Домашние задания

Знакомимся с командой и гит репозиторием

Цель

проработать организационные моменты и лучше познакомиться друг с другом.

Тема 2

Domain Driven Development: Основы

Цель занятия

изучить теоретическую часть подхода Domain Driven Development.

Краткое содержание

основные определения;
концепция: ограниченные связи, целостность, взаимосвязь;
элементы DDD:
- ограниченный контекст
- сущность
- объект-значение
- агрегат
- службы предметных областей

Anti-corruption Layer.

Тема 3

Domain Driven Development: Практикум

Цель занятия

научиться применять подход самостоятельно на практике.

Краткое содержание

практическое занятие по применению DDD.

Тема 4

Архитектура проекта

Цель занятия

формировать архитектуру приложения;
разделять код на слои и звенья;
принимать более взвешенные решения о структуре проекта.

Краткое содержание

слои;
звенья;
интерфейсы;
API.

Тема 5

Базы данных: организация работы с потоками данных

Цель занятия

объяснить основы работы с базами данных.

Краткое содержание

SQL базы данных;
принципы работы SQL баз данных;
отличие NO SQL от SQL баз данных;
ACID;
ORM.

Тема 6

Базы данных: реляционные базы и работа с ними // ДЗ

Цель занятия

научиться работать с реляционными базами данных.

Краткое содержание

Dapper;
Multiselect, одновременная нагрузка;
архитектура в oltp/olap нагрузке и как нужно когда строить;
в каких случаях использовать for json, а в каких api;
когда нужно делать все на базе, а когда можно и entity.

Домашние задания

Создание виртуальной базы данных магазина в PostgreSQL

Цель

научиться создавать базу данных с таблицами, а также писать скрипты с использованием основных ключевых слов.
Вы создадите два файла со скриптами: один для создания всех необходимых таблиц с ключами и связями, а другой для запросов.

Тема 7

Базы данных: Entity Framework Core

Цель занятия

научиться взаимодействовать с базой данных при помощи фреймворка EF Core.

Краткое содержание

разбор основных элементов и способов построения контекста БД;
практикум на построение запросов к БД;
обзор способов оптимизации автоматически генерируемых запросов.

Тема 8

Базы данных: NoSQL базы и их особенности

Цель занятия

понять, как выбирать между реляционными и NoSQL СУБД в своих проектах;
уметь проектировать оптимальную для хранения в NoSQL СУБД модель данных;
освоить базовые приемы работы с одной из NoSQL СУБД: Redis, MongoDb, или Cassandra.

Краткое содержание

ACID и BASE;
отличия реляционной структуры от агрегатной структуры хранения, паттерн агрегат, как основная единица группировки данных в большинстве NoSql СУБД;
CAP теорема, PACELC
основные подходы к разделению данных в СУБД
шардирование и репликации
обзор основных видов NoSql баз данных: ключ-значение (Redis), документоориентированные (Mongo), семейство столбцов (Cassandra), графоориентированные (Neo4j);
обзор плюсов и минусов различных NoSQL баз и задач, в которых они могут быть полезны;
примеры использования на одной из NoSQL СУБД (Redis, MongoDb или Cassandra) в ASP.NET Core проекте, использование драйверов для NoSQL СУБД под .NET.

Тема 9

Ретроспектива и планирование

Цель занятия

уверенно участвовать в ретроспективах;
планировать объём работ на будущее.

Краткое содержание

проведение ретроспективы прошедшего спринта;
проведение планирования будущего спринта.

Модуль 2. Многопоточность и шаблоны проектирования

Тема 1

Введение в параллелизм в .NET. Отличия процесса, потока, домена и таска // ДЗ

Цель занятия

дать описание разным примитивам параллелизма для лучшего понимания их назначения и отличий;
получить примеры практического использования инструментов параллелизма на практической задаче.

Краткое содержание

определения и понятия параллелизма с точки зрения практической задачи по обработке очереди уведомлений;
описание процессам и потокам ОС, а также рассмотрим их применение в .NET;
работа доменов приложений и размещением CLR в .NET и .NET Core;
понятия синхронности и асинхронности, а также механизм Task.

Домашние задания

Параллельное считывание файлов

Цель

обрести базовые навыки работы с тасками, запустив таски в параллель.

Тема 2

Асинхронные операции

Цель занятия

писать асинхронный код.

Краткое содержание

пул потоков;
машина состояний;
отличие асинхронного подхода от конкурентного;
async/await.

Тема 3

Магические слова async / await

Цель занятия

изучить механизм, скрытый под ключевыми словами async/await;
научиться правильному использованию этих ключевых слов.

Краткое содержание

жизненный цикл Task;
что такое контекст синхронизации;
разбор использования ConfigureAwait(false);
синхронная асинхронность.

Тема 4

Синхронизация доступа к общему ресурсу

Цель занятия

познакомиться с примитивными и гибридными средствами управления доступа к общим ресурсам;
рассмотреть дополнительные инструменты для упрощения организации многопоточной работы;
применить полученные знания в практическом занятии.

Краткое содержание

Monitor и lock;
Mutex;
Semaphore;
ReaderWriterLock;
SpinLock;
Slim-версии примитивов.

Тема 5

Взаимодействие потоков

Цель занятия

познакомиться с примитивами управления потоков;
понять в каких моментах использовать эти примитивы;
применить примитивы на практике.

Краткое содержание

EventWaitHandle, AutoResetEvent и ManualResetEvent;
CountdownEvent;
Barrier;
Interlocked;
SpinWait.

Тема 6

Внутрипроцессное взаимодействие // ДЗ

Цель занятия

применять потоки, задачи, Parallel LINQ;
распараллеливать расчёты для ускорения вычислений;
оценивать целесообразность применения механизмов;
параллельной обработки данных.

Краткое содержание

потоки в операционной системе (на примере Windows API);
поток (Thread);
задача (Task);
Parallel For;
Parallel LINQ.

Домашние задания

Многопоточный проект

Цель

применять разные способы распараллеливания задач и выбирать оптимальный способ реализации.

Тема 7

Межпроцессное взаимодействие

Цель занятия

изучить способы взаимодействия процессов/программ друг с другом.

Краткое содержание

общая память;
сокеты;
именованные каналы;
WCF, REST, etc.
общие файлы;
база данных;
очереди сообщений.

Тема 8

Порождающие шаблоны проектирования // ДЗ

Цель занятия

объяснить назначение изученных шаблонов проектирования;
применять их в своих проектах.

Краткое содержание

шаблоны проектирования Singleton, Prototype, Factory method, Abstract Factory и Builder.

Домашние задания

Реализуем паттерн "Прототип"

Цель

создать иерархию из нескольких классов, в которых реализованы методы клонирования объектов по шаблону проектирования "Прототип".

Тема 9

Структурные шаблоны проектирования

Цель занятия

объяснить назначение изученных шаблонов проектирования и применять их в своих проектах.

Краткое содержание

шаблоны проектирования Proxy, Adapter, Decorator, Facade, Bridge, Composite, Flyweight.

Тема 10

Поведенческие шаблоны проектирования

Цель занятия

объяснить назначение изученных шаблонов проектирования и применять их в своих проектах.

Краткое содержание

шаблоны проектирования Command, Iterator, Mediator, Memento, Observer, State, Template Method;
рассмотрение на конкретном примере шаблонов Strategy и Visitor.

Тема 11

Ретроспектива и планирование

Цель занятия

уверенно участвовать в ретроспективах;
планировать объём работ на будущее.

Краткое содержание

проведение ретроспективы прошедшего спринта;
проведение планирования будущего спринта.

Модуль 3. Клиент-серверная архитектура и микросервисы

Тема 1

Интеграция приложений

Цель занятия

сформулировать основные причины интеграции корпоративных систем и способы их интеграции;
сформулировать какой способ взаимодействия предпочтителен для различных типов задач и в чем причина популярности микросервисов.

Краткое содержание

причины и способы интеграции корпоративных приложений, зачем нужна микросервисная архитектура.

Тема 2

Принципы SOLID // ДЗ

Цель занятия

изучить значение каждой буквы акронима;
узнать как применять эти принципы на практике.

Краткое содержание

принцип единственной ответственности;
принцип открытости/закрытости;
принцип подстановки Барбары Лисков;
принцип разделения интерфейса;
принцип инверсии зависимостей.

Домашние задания

Демонстрация SOLID принципов

Цель

практическое применение SOLID принципов.

Тема 3

Брокеры сообщений

Цель занятия

разобраться в сходствах и отличиях четырёх наиболее популярных брокеров сообщений;
научиться писать код, получающий сообщения из очереди.

Краткое содержание

что такое "Брокер сообщений";
обзор нескольких брокеров сообщений;
создание кластера RabbitMQ в облаке;
знакомство с интерфейсом панели администратора RabbitMQ:
создание, просмотр, настройка и удаление очередей.
самостоятельная реализация отправки и получения сообщений из очереди облачного RabbitMQ.

Тема 4

CI/CD

Цель занятия

разобраться в принципах CI/CD;
понимать как организовать правильный CI/CD;
сформулировать основные инструменты для CI и CD, их отличия;
настроить собственный CI/CD.

Краткое содержание

принципы CI/CD;
правильная настройка и организация;
основные инструменты и их отличия;
настройка собственного CI/CD.

Тема 5

Ретроспектива и планирование

Цель занятия

уверенно участвовать в ретроспективах;
планировать объём работ на будущее.

Краткое содержание

проведение ретроспективы прошедшего спринта;
проведение планирования будущего спринта.

Модуль 4. C# Advanced

Тема 1

Отражение (Reflection) // ДЗ

Цель занятия

обрабатывать экземпляры разных или неизвестных заранее классов;
создавать универсальные тесты;
писать свои компоненты.

Краткое содержание

использование механизмов доступа к параметрам типов через механизмы отражения;
динамическая загрузка сборок и позднее связывание.

Домашние задания

Рефлексия и её применение

Цель

написать свой класс-сериализатор данных любого типа в формат CSV, сравнить его с типовыми механизмами сериализации. Изучить возможности Reflection.

Тема 2

Сериализация

Цель занятия

использовать механизмы сериализации и результирующие форматы сериализации;
применять стандартные способы сериализации.

Краткое содержание

что такое сериализаци;
какие бывают форматы, их плюсы и минусы;
интерфейс ISerializable и IFormatter;
стандартные средства сериализации.

Тема 3

Атрибуты

Цель занятия

использовать существующие атрибуты;
проверять наличие атрибутов на классах, функциях, полях, и т.д.;
создавать свои собственные атрибуты.

Краткое содержание

что такое атрибут;
как применяются атрибуты;
где они уже используются;
стандартные и кастомные атрибуты;
как создавать свои атрибуты.

Тема 4

Исключения и нюансы работы с ними

Цель занятия

объяснить, что из себя представляет исключение;
разобраться как исключение устроено "под капотом";
научиться бросать и перехватывать исключения;
познакомиться с Best Practice в работе с исключениями.

Краткое содержание

исключение;
иерархии исключений;
перехват исключений;
обработка исключений.

Тема 5

Работа с методами как с переменными (delegates, events) // ДЗ

Цель занятия

создавать делегаты для передачи методов в функции;
создавать события и подписки на них;
писать менее связный код и упростить расширение функционала ПО.

Краткое содержание

делегаты и их использование;
применение событий;
отличия событий и делегатов;
стандартизация событий в .NET

Домашние задания

Делегаты и события

Цель

попрактиковаться в реализации механизмов делегатов и событий.

Тема 6

Дженерики, их реализация и ограничения

Цель занятия

работать с обобщениями.

Краткое содержание

обобщённые методы;
обобщённые классы:
- значения по умолчанию: default(T);
- несколько обобщённых параметров: class Transaction<U, V>.
- во что разворачиваются в момент выполнения;
ограничения универсальных типов:
- стандартные ограничения: class, struct, new;
- гибкие ограничения: классы и интерфейсы.

Тема 7

Сборщик мусора, деструкторы и финализаторы, Disposable Pattern

Цель занятия

разобраться, как хранятся объекты в памяти в .NET;
познакомиться с алгоритмом выделения физической памяти для приложений;
понять алгоритм работы Сборщика Мусора (поколения, стратегии, карточный стол);
начать отличать Деструкторы от Финализаторов;
научиться использовать Disposable Pattern.

Краткое содержание

как выделяется память;
сборщик мусора;
деструкторы и финализаторы;
Disposable Pattern.

Тема 8

Дополнительные возможности языка: от директив препроцессора до указателей

Цель занятия

писать небезопасный код и создавать указатели, если, столкнётесь с тем, что это понадобится;
использовать механизмы условной компиляции, чтобы обеспечить зависимость поведения проекта от окружения или обеспечить большее удобство работы с кодом;
применять динамические объекты и заготовки кода,
чтобы ускорить написание проектов.

Краткое содержание

директивы препроцессора;
Unsafe code, pointers;
DLR - Dynamic & ExpandoObject.

Тема 9

Что полезного в новых версиях C#?

Цель занятия

проанализировать нововведения в новых стандартах.

Краткое содержание

ключевые нововведения версий;
полезные и часто используемые новшества .Net.

Тема 10

Ретроспектива и планирование

Цель занятия

уверенно участвовать в ретроспективах.

Краткое содержание

проведение ретроспективы прошедшего спринта.

Модуль 5. Процессы и подходы

Тема 1

Waterfall, Scrum, Kanban и прочие методологии

Цель занятия

разобраться в современных методологиях, необходимости и особенностям их применения.

Краткое содержание

что было до Agile;
базовые Agile фреймворки;
проблемы внедрения и как с ними бороться;
фреймворки масштабирования.

Тема 2

Unit, Sandbox, Blackbox, Whitebox, Integration tests

Цель занятия

разбираться во всём этом многообразии типов тестов;
понимать когда и какие нужно применять.

Краткое содержание

unit тестирование;
blackBox тестирование;
whiteBox тестирование;
модульное тестирование;
интеграционное тестирование;
регрессионное тестирование.

Тема 3

Логирование, метрики, трассировка и ведение документации

Цель занятия

разбираться в инструментах исследования и анализа работы программ;
выбирать и настраивать подходящую систему логирования;
объяснять что из себя представляют метрики и как устроена трассировка;
перечислить минимум три системы для ведения программной документации.

Краткое содержание

логирование:
- функции систем логирования
- что необходимо логировать
- распространённые ошибки логирования
- выбор правильных уровней логирования
- структурные логи
- рекомендации по настройке

метрики;
трассировка;
документация.

Тема 4

Реактивное программирование

Цель занятия

изучить основные паттерны работы с последовательностями данных в режимах pull и push;
проанализировать интерфейс IObservable<T> и связанные с ним соглашения;
изучить основные возможности LINQ для IObservable<T>;
получить пример работы с подписками на данные в .net-приложении.

Краткое содержание

устройство IObservable<T> и IObserver<T>, параллели с IEnumerable<T> и IEnumerator<T>;
synchronized observables;
unicast/multicast observables;
hot/cold observables;
основные методы LINQ to observables;
связь IObservable<T> с Task<T>.

Тема 5

Функциональное программирование

Цель занятия

понимать принципы функционального программирования и их отличия от других парадигм;
применять основные техники функционального программирования;
эффективно использовать лямбда-выражения и LINQ для обработки коллекций;
разрабатывать функции высших порядков и использовать их для обработки данных.

Краткое содержание

введение в функциональное программирование и его преимущества;
принципы функционального программирования: неизменяемость, чистые функции, функции первого класса;
применение лямбда-выражений и LINQ8;
создание и использование функций высших порядков

Модуль 6. Проектная работа

Тема 1

Консультация по проектам и домашним заданиям

Цель занятия

получить ответы на вопросы по проекту, ДЗ и по курсу.

Краткое содержание

вопросы по улучшению и оптимизации работы над проектом;
затруднения при выполнении ДЗ;
вопросы по программе.

Домашние задания

Проектная работа

Цель

В этой работе вы продемонстрируете знания и навыки, которые приобрели на курсе, а также умение командного взаимодействия при создании информационной системы по одной из тем ниже:
- Обучающая платформа
- Планирование отпусков
- Телеграмм-бот для музыкальной студии
- Игра с сервером
- Своя тема

Тема 2

Защита проектных работ

Цель занятия

защитить проект и получить рекомендации экспертов.

Краткое содержание

презентация проектов перед комиссией;
вопросы и комментарии по проектам.