

Департамент информатизации Тюменской области
Государственное автономное учреждение дополнительного образования
Тюменской области
«Региональный информационно-образовательный центр»

СОГЛАСОВАНО

Исполняющий обязанности директора
ГАУ ТО «Западно-Сибирский
инновационный центр»

А.Н. Ермолаев
« 20 » _____ 2022 г.


УТВЕРЖДАЮ

Директор
ГАУ ДО ТО «РИО-Центр»

Т.А. Беляева
« 20 » _____ 2022 г.


УЧЕБНАЯ ПРОГРАММА
«Программирование на языке C#
Для платформы .NET Core»

Трудоемкость программы – 80 академических часа

Форма обучения – очная

Режим занятий – 3 академических часа в день

Начальные навыки: средние навыки высокоуровневого и объектно-ориентированного программирования

Цель обучения:

сформировать у слушателей:

- углублённые знания, умения и навыки объектно-ориентированного программирования, позволяющие разрабатывать функциональные Веб-приложения;
- углублённые знания использования языка программирования C#;
- основные знания, умения и навыки работы в среде Microsoft Visual Studio;
- основные знания, умения и навыки использования языка JavaScript и вспомогательных библиотек для создания клиентской части Веб-приложения.

Компетенции на выходе:**Знания:**

- технология разработки сервисов и клиент-серверных приложений (на примере платформы .NET Core).

Умения:

- использовать возможности платформы и языка C# для создания REST-сервисов и серверной части приложений;
- использовать реляционную СУБД PostgreSQL и технологию ORM для работы с данными приложения;
- использовать язык JavaScript для создания клиентской части Веб-приложений;
- использовать Unit-тестирование для проверки работоспособности приложения.

Приёмы работы:

- со средой разработки;
- с СУБД;
- с REST-сервисами;
- с отладчиком интегрированной среды разработки, браузера.

Программой обучения предусмотрены аудиторные занятия и самостоятельная работа слушателей. Под самостоятельной работой подразумевается самостоятельное выполнение слушателями практических заданий в учебной аудитории и в присутствии преподавателя.

УЧЕБНЫЙ ПЛАН

№ п/п	Наименование тем	Всего часов	в том числе:		Формы контроля
			аудиторн ые занятия	самос- тоятель- ная работа	
1.	Ретроспектива объектно-ориентированного программирования (ООП)	6	6		Выполнение практических заданий
2.	Платформа .NET Core	3	3		
3.	Разработка требований к системе	11	11		
4.	Работа с СУБД	9	9		
5.	Модульное тестирование	5	5		
6.	Использование технологии WebAPI 2 для создания сервисов	13	13		
7.	Разработка клиентской части с использованием JavaScript/TypeScript	17	17		
8.	Работа с геоинформацией	6	6		
9.	Интеграция решения: клиент, сервер, данные	7	7		
10.	Оценка проектов. Итоговое тестирование	3	3		Защита проекта
Итого часов:		80	80	0	

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№п/п	Наименование тем	Всего часов	в том числе:		Формы контроля
			аудитор ная работа	самостояте льная работа	
1.	Ретроспектива объектно-ориентированного программирования (ООП)	6	6		Выполнение практических заданий
1.1.	Основные принципы ООП: инкапсуляция, наследование, полиморфизм, абстракция	3	3		
1.2.	Абстрактные классы. Интерфейсы	3	3		
2.	Платформа .NET Core	3	3		Выполнение практических заданий
2.1.	Сравнение с платформой .NET Framework	1	1		

2.2.	Назначение, новшества	2	2		
3.	Разработка требований к системе	11	11		Выполнение практических заданий
3.1.	Постановка задачи. Функциональная спецификация проекта	3	3		
3.2.	Разработка доменной модели проекта. Предметно-ориентированное проектирование (Domain-Driven Design, DDD)	6	6		
3.3	Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.	2	2		
4.	Работа с СУБД	9	9		Выполнение практических заданий
4.1.	Реляционная модель данных. Основные виды запросов СУБД .PostgreSQL	3	3		
4.2.	Технология ORM для работы с данными	3	3		
4.3.	Использование Entity Framework. Методы «Data First», «Code First»	3	3		
5.	Модульное тестирование	5	5		Выполнение практических заданий
5.1.	Разработка через тестирование (Test-Driven Development, TDD)	1	1		
5.2.	Модульное тестирование на примере NUnit	2	2		
5.3	Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.	2	2		
6.	Использование технологии WebAPI 2 для создания сервисов	13	13		Выполнение практических заданий
6.1.	Введение в Веб-сервисы	2	2		
6.2.	REST-сервисы на основе WebAPI 2	3	3		
6.3.	Создание Self-Host-сервиса	2	2		

6.4.	Внедрение зависимостей	2	2		
6.5.	Использование формата JSON для взаимодействия с клиентом	2	2		
6.6	Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.	2	2		
7.	Разработка клиентской части с использованием JavaScript/TypeScript	17	17		Выполнение практических заданий
7.1.	Знакомство с TypeScript, JavaScript. Прототипирование	2	2		
7.2.	Разработка клиентской части приложения с помощью Node.js	2	2		
7.3.	Использование Promises для выполнения асинхронных операций	2	2		
7.4.	Библиотека JQuery	5	5		
7.4.1.	Взаимодействие с пользователем	2	2		
7.4.2.	Взаимодействие с серверной частью	2	2		
7.5	Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.	2	2		
8.	Работа с геоинформацией	6	6		Выполнение практических заданий
8.1.	Картографические проекции и системы координат	3	3		
8.2.	Использование Yandex.Maps. API и приёмы работы	3	3		
9.	Интеграция решения: клиент, сервер, данные	7	7		
10.	Оценка проектов. Итоговое тестирование	3	3		
Итого часов:		80	80		

СОДЕРЖАНИЕ ПРОГРАММЫ

- 1. Ретроспектива объектно-ориентированного программирования (ООП)**
 - 1.1. Основные принципы ООП: инкапсуляция, наследование, полиморфизм, абстракция.
 - 1.2. Абстрактные классы. Интерфейсы.
- 2. Платформа .NET Core**
 - 2.1. Сравнение с платформой .NET Framework.
 - 2.2. Назначение, новшества.
- 3. Разработка требований к системе**
 - 3.1. Постановка задачи. Функциональная спецификация проекта.
 - 3.2. Разработка доменной модели проекта. Предметно-ориентированное проектирование (Domain-Driven-Design, DDD)
 - 3.3. Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.
Разбор сложного материала. Защита промежуточных проектов и контрольных работ. Встреча с работодателями.
- 4. Работа с СУБД**
 - 4.1. Реляционная модель данных. Основные виды запросов. СУБД PostgreSQL.
 - 4.2. Технология ORM для работы с данными.
 - 4.3. Использование Entity Framework. Методы “Data First”, “Code First”.
- 5. Модульное тестирование.**
 - 5.1. Разработка через тестирование (Test-Driven Development, TDD).
 - 5.2. Модульное тестирование на примере NUnit.
 - 5.3. Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.
Разбор сложного материала. Защита промежуточных проектов и контрольных работ. Встреча с работодателями.
- 6. Использование технологии WebAPI 2 для создания сервисов**
 - 6.1 Введение в Веб-сервисы.
 - 6.2 REST-сервисы на основе WebAPI 2.
 - 6.3 Создание Self-Host-сервиса.
 - 6.4 Внедрение зависимостей.
 - 6.5 Использование формата JSON для взаимодействия с клиентом.
 - 6.6. Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.
Разбор сложного материала. Защита промежуточных проектов и контрольных работ. Встреча с работодателями.
- 7. Разработка клиентской части с использованием JavaScript/TypeScript**
 - 7.1 Знакомство с TypeScript, JavaScript. Прототипирование.
 - 7.2 Разработка клиентской части приложения с помощью Node.js.
 - 7.3 Использование Promises для выполнения асинхронных операций.
 - 7.4. Библиотека JQuery.
 - 7.4.1. Взаимодействие с пользователем.
 - 7.4.2. Взаимодействие с серверной частью.
 - 7.5. Рефлексия. Обсуждение выполненных заданий. Консультация по индивидуальному проекту.

Разбор сложного материала. Защита промежуточных проектов и контрольных работ.
Встреча с работодателями.

8. Работа с геоинформацией

8.1 Картографические проекции и системы координат.

8.2 Использование Yandex.Maps. API и приёмы работы.

9. Интеграция решения: клиент, сервер, данные

10. Оценка проектов. Итоговое тестирование