

**Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский технологический
университет «МИСиС»**

УТВЕРЖДАЮ

Проректор по дополнительному образованию

В.Л. Петров

2021 г.



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ОБЩЕРАЗВИВАЮЩАЯ
ПРОГРАММА**

**«Разработка веб-сервисов на платформе.
NET Core на языке C# с нуля»**

НАПРАВЛЕННОСТЬ: ТЕХНИЧЕСКАЯ

Уровень: ознакомительный

Возраст обучающихся 14 - 18 лет

Срок реализации: не менее 3-х месяцев

Составитель (разработчик):

Зорин И.А.

старший преподаватель кафедры АПД

НИТУ «МИСиС»

Москва
2021 год

СОДЕРЖАНИЕ

1.	Пояснительная записка	3
2.	Учебный (тематический) план	5
3.	Содержание учебного (тематического) плана	6
4.	Формы аттестации и контроля	7
5.	Организационно-педагогические условия реализации программы	8
6.	Список литературы	9

1. Пояснительная записка

Характеристика дополнительной общеобразовательной общеразвивающей программы

Дополнительная общеобразовательная (общеразвивающая) программа дополнительного образования детей и взрослых реализуется Федеральным государственным автономным образовательным учреждением высшего образования «Национальный исследовательский технологический университет «МИСиС» (далее – НИТУ «МИСиС», Университет) «Разработка веб-сервисов на платформе .NET Core на языке C# с нуля» (далее - Программа).

Направленность программы - техническая. Программа направлена на передачу учащимся первичных навыков разработки программного обеспечения для сети Интернет.

Уровень освоения – ознакомительный. Программы призвана максимально простыми языковыми и технологическими средствами донести до учащихся основные принципы разработки веб-сервисов.

Актуальность программы. Разработка веб-сервисов в настоящее время – это одно из наиболее популярных направлений программирования. Учитывая широкое использование сети Интернет практически в любой сфере деятельности, разработка веб-сервисов как бэкенда веб-приложений, представляет собой мощнейший ресурс для потенциального программиста.

Новизна программы заключается в том, что используется метод итеративной разработки.

Педагогическая целесообразность. Концептуальная идея предлагаемого курса состоит в формировании у учащихся культуры последовательной разработки программного обеспечения для решения поставленной задачи. При этом одной из особенностей курса является привитие учащемуся навыка движения «маленькими шагами» или, как сейчас принято говорить, итеративной разработки.

Отличительной особенностью программы является то, что она реализуется в короткие сроки за счёт нестандартных методов изучения материала, простого объяснения сложных понятий разработки и при этом подачи только той части программирования, которая необходима для обучающегося в рамках данного предмета.

Цель и задачи

Цель формирование и развитие у обучающихся интеллектуальных и практических компетенций в сфере разработки приложений для сети Интернет.

Задачи

Обучающие:

- знакомство учащихся с техническими основами сети Интернет;
- знакомство учащихся с основами программирования;
- освоение учащимися простых операций с данными;

Развивающие:

- обучение механизмам поиска решений в интернете;
- формирование практических навыков программирования;
- формирование понимания устройства сети Интернет;

Воспитательные:

- формирование навыка преодоления ошибок;
- формирование ответственности за результат;
- формирование творческого отношения к выполняемой работе.

Категория обучающихся

Возраст обучающихся: 14-18 лет.

Сроки реализации: не менее 3-х месяцев, 36 академических часов.

Формы и режим занятий

Формы проведения занятий: лекции, практические занятия, мастер-классы.

Формы организации деятельности: групповые и индивидуально-групповые.

Наполняемость группы: не более 25 человек.

Режим занятий: 1 занятие в неделю по 3-4 академических часа.

Ожидаемые результаты

В результате освоения программы обучающиеся **будут знать:**

- теоретические основы устройства сетей и сети Интернет в частности;
- теоретические основы программирования на языке C#;
- подходы к разработке веб-сервисов;
- правила работы с базой данных в веб-сервисах с использованием MS Entity Framework;

будут уметь:

- разрабатывать простейшие веб-сервисы на языке C# на платформе .NET Core;
- работать с данными в реляционных базах данных из программы на C#;
- вызывать сторонние сервисы из программы на языке C#;

2. Учебный (тематический) план

№ п/п	Раздел / Тема	Количество часов			Форма аттестации/ контроля
		Всего	Теория	Практические занятия	
1.	Блок 1. Краткое устройство современных компьютерных сетей				
1.1	Что такое компьютерная сеть. Как она устроена на физическом уровне. Сетевые протоколы. Модель OSI. Сеть Интернет, её устройство.	2	2	0	Опрос
1.2	Использование Insomnia для работы с http-запросами. Создание простейших запросов.	2	1	1	Опрос
2.	Блок 2. Основы программирования				
2.1	Работа в среде MS Visual Studio 2019 Community. Создание консольного приложения. Типы данных. Арифметические операции.	2	1	1	Практическая работа
2.2	Создание веб-сервиса.	2	1	1	Практическая работа
3	Блок 3. Алгоритмы				
3.1	Условные операторы	2	1	1	Практическая работа
3.2	Циклы	2	1	1	Практическая работа
3.3	Массивы	2	0	2	Практическая работа
4.	Блок 4. Основы Объектно-ориентированного программирования				
4.1	ООП, класс, объект	2	1	1	Практическая работа
4.2	Конструкторы, методы	2	1	1	Практическая работа
4.3	Наследование, инкапсуляция, полиморфизм	4	1	3	
5	Блок 5. Работа с базами данных				
5.1	Создание простейшей реляционной базы данных. Связывание таблиц.	2	0	2	Практическая работа
5.2	Доступ к базе данных из сервиса через Entity Framework.	2	0	2	Практическая работа
6	Блок 6. Методологии для работы с веб-сервисами				
6.1	DDD	2	1	1	Практическая работа
6.2	Модульное тестирование	2	1	1	Практическая работа
7	Блок 7. Выкладка сервиса в интернет				

7.1	Публикация на хостинге	2	0	2	Практическая работа
7.2	Контейнеризация	4	0	4	Практическая работа
Итоговая аттестация проводится на основании совокупности выполненных промежуточных практических работ.					
Всего		36	12	24	

3. Содержание учебного (тематического) плана

Блок 1. Краткое устройство современных компьютерных сетей

1.1 Что такое компьютерная сеть. Как она устроена на физическом уровне. Сетевые протоколы. Модель OSI. Сеть Интернет, её устройство.

1.Теория (2 а.ч.) Сети. Физический уровень сети. Протоколы.

1.2 Использование Insomnia для работы с http-запросами. Создание простейших запросов.

1.Теория (1 а.ч.) Запросы.

2. Практика (1 а.ч.) Создание простейших запросов для сети интернет.

Планируемые результаты. Учащиеся смогут описать путь прохождения сигнала от своего компьютера до нужного сайта, отправить запрос на нужный Uri, рассказать, как устроена сеть Интернет, что такое доменные имена, кто их выдаёт, что такое IP-адрес, DNS, чем отличаются защищённые протоколы передачи данных от открытых, из каких составных частей состоит http-запрос.

Блок 2. Основы программирования в MS Visual Studio 2019

2.1 Работа в среде MS Visual Studio 2019 Community. Создание консольного приложения. Типы данных. Арифметические операции.

1.Теория (1 а.ч.) Среда MS Visual Studio 2019 Community. Типы данных. Арифметические операции

2. Практика (1 а.ч.) Работа с типами данных, запись арифметических операций и их выполнение.

2.2 Создание веб-сервиса.

1.Теория (1 а.ч.) Консольное приложение. Веб-сервис.

2. Практика (1 а.ч.) Создание консольного приложения.

Планируемые результаты. Учащиеся создадут своё первое консольное приложение и первый веб-сервис, который будет выполнять арифметические операции и возвращать результат.

Блок 3. Алгоритмы

3.1 Условные операторы

1.Теория (1 а.ч.) Условные операторы. Основные виды алгоритмов

2. Практика (1 а.ч.) Создание алгоритмов с использованием условных операторов

3.2 Циклы

1.Теория (1 а.ч.) Цикл. Основные структуры циклических конструкций.

2. Практика (1 а.ч.) Создание алгоритмов с использованием циклических конструкций

3.3 Массивы

1. Практика (2 а.ч.) Создание массивов данных. Обработка.

Планируемые результаты. Учащиеся добавят в сервис какую-либо бизнес-логику, научатся обрабатывать входные данные, содержащие массив элементов.

Блок 4. Основы объектно-ориентированного программирования

4.1 ООП, класс, объект

1. Теория (1 а.ч.) Основные понятия объектно-ориентированного программирования (ООП)

2. Практика (1 а.ч.) Знакомство с основными классами, объектами ООП.

4.2 Конструкторы, методы

1. Теория (1 а.ч.) Конструкторы, методы запуска сторонних сервисов

2. Практика (1 а.ч.) Знакомство с конструкторами, создание метода запуска стороннего сервиса

4.3 Наследование, инкапсуляция, полиморфизм

1. Теория (1 а.ч.) Наследование, инкапсуляция, полиморфизм

2. Практика (3 а.ч.) Создание веб-сервиса. Запуск стороннего сервиса

Планируемые результаты. Учащиеся создадут веб-сервис, который парсит входящие наборы данных и обрабатывает их. Также они создадут метод, который запускает сторонний сервис и обрабатывает набор полученных оттуда данных.

Блок 5. Работа с базами данных

5.1 Создание простейшей реляционной базы данных. Связывание таблиц.

1. Практика (2 а.ч.) Создание таблиц. Связь таблиц. Создание простейшей реляционной базы данных.

5.2 Доступ к базе данных из сервиса через Entity Framework.

1. Практика (2 а.ч.) Присоединение к своему сервису базу данных с использованием Entity Framework, выполнение CRUD-действий с БД.

Планируемые результаты. Учащиеся добавят к своему сервису базу данных с использованием Entity Framework и научатся выполнять простейшие CRUD-действия с БД.

Блок 6. Паттерны для работы с веб-сервисами

6.1 DDD

1. Теория (1 а.ч.) Сервис Domain driven design

2. Практика (1 а.ч.) Перестраивание сервиса в соответствии с DDD

6.2 Модульное тестирование

1. Теория (1 а.ч.) Модульные тесты

2. Практика (1 а.ч.) Создание модульного теста для своего сервиса

Планируемые результаты. Учащиеся перестроят свой сервис в соответствии с паттерном DDD и добавят к нему модульный тест.

Блок 7. Выкладка сервиса в интернет

7.1 Публикация на хостинге

1. Практика (2 а.ч.) Публикация сервиса в интернет

7.2 Контейнеризация

1. Практика (4 а.ч.) Упаковка сервиса в контейнер

Планируемые результаты. Учащиеся выложат свой сервис в сеть интернет на один из бесплатных хостингов.

4. Формы аттестации и контроля

Виды контроля. В образовательном процессе используются следующие методы определения результативности и подведения итогов программы:

Текущий контроль. Проводится с целью непрерывного отслеживания уровня усвоения материала и стимулирования учащихся не отвлекаться. Для реализации текущего контроля в процессе теоретического материала педагог обращается к учащимся с вопросами и дает короткие

задания, в процессе практических занятий по итогам каждой темы обучающийся выполняет практические задания.

Итоговый контроль. Проводится на основании совокупности выполненных промежуточных практических работ.

В процессе обучения будут применяться различные методы контроля, в том числе с использованием современных технологий.

Форма итоговой аттестации – зачет на основании совокупности выполненных работ текущего контроля.

Оценивание: зачтено/не зачтено.

5. Организационно-педагогические условия реализации программы

Организационно-педагогические ресурсы

Специализированные лаборатории и классы, основные установки и стенды

Площадка: г. Москва, Ленинский проспект, дом 6, строение 1 (корпус «Л»)

Компьютерный класс: аудитория Л- 538 кафедры АПид

Оборудование и программное обеспечение:

Операционная система: Windows 8 и Windows 10 (Windows RT не поддерживается).

Аппаратное обеспечение:

1) ПЭВМ по количеству учащихся (желательно ноутбук). Минимальные системные требования:

- Операционная система Windows (7, 8,10) или MacOS (10.6, 10.7, 10.8);
- 8 ГБ оперативной памяти;
- Процессор 1.5 ГГц;
- 750 Мб свободного дискового пространства;
- Разрешение экрана 1024*600;
- 2) Microsoft.NET Core 3.0;
- 3) Среда программирования MS Visual Studio 2019 Community или выше;
- 4) PostgreSQL

Методическое обеспечение программы

Методы обучения, используемые в программе: словесные (устное объяснение материала), наглядные (презентация), практические (обучающиеся решают задачи по программированию), аналитические.

С целью стимулирования творческой активности учащихся используются:

- метод проектов;
- метод погружения;
- методы сбора и обработки данных;
- игровые методики;
- исследовательский и проблемный методы;
- анализ справочных и литературных источников;
- поисковый эксперимент;
- опытная работа;
- обобщение результатов.

Для обеспечения наглядности и доступности изучаемого материала используются:

- наглядные пособия смешанного типа (слайды, видеозаписи);
- дидактические пособия (карточки с заданиями, доска Agile).

Кадровое обеспечение программы

Реализатор программы: Зорин Иван Андреевич – старший преподаватель кафедры АПид НИТУ «МИСиС».

6. Список литературы

- 1) Дж. Рихтер, CLR via C#, Питер 2013г.
- 2) Г. Шилдт, C# 4.0, полное руководство, Вильямс, 2011г.
- 3) Д. Эспозито, Программирование на основе Microsoft ASP.NET, Русская редакция, БХВ-Питер, 2012
- 4) К. Ричардсон, Микросервисы. Паттерны разработки и рефакторинга, Питер, 2019
- 5) Г. Буч, Объектно-ориентированное проектирование и анализ, Санта-Клара, Калифорния, 2013
- 6) Р.С. Мартин, М. Мартин, Принципы, паттерны и методики гибкой разработки на языке C#, Симбо, 2011
- 7) Бондарь А.Г., Microsoft SQL Server 2014, БХВ-Петербург, 2015