


МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ ТАДЖИ-  
КИСТАН  
МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКО-ТАДЖИКСКИЙ (СЛАВЯНСКИЙ) УНИВЕРСИТЕТ»

«Утверждаю»  
Декан естественнонаучного факультета  
Петукович А.И.  
2026 г.



**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

**NET-программирование**

Направление подготовки - 10.03.01 «Информационная безопасность»  
Профиль подготовки – Безопасность компьютерных систем (по отрасли или в  
сфере профессиональной деятельности)  
Форма подготовки – Очная  
Уровень подготовки – Бакалавриат

**ДУШАНБЕ - 2026**

Рабочая программа составлена на основе требований федерального государственного образовательного стандарта высшего образования по направлению подготовки 10.03.01 «Информационная безопасность» (уровень Бакалавриат), утвержденного приказом Министерства образования и науки РФ №524 от 08.06.2017 г., Концепции преподавания NET-программирование для специальностей и направлений подготовки, реализуемых в образовательных организациях высшего образования, утвержденной протоколом Экспертного совета по развитию исторического образования Минобрнауки РФ от 06.08.2024 г. №ВФ/35-ПР

При разработке рабочей программы учитываются

- содержание программ дисциплин, изучаемых на предыдущих и последующих этапах обучения;
- новейшие достижения в данной предметной области.

Рабочая программа обсуждена на заседании кафедры Кафедра информатики и информационных технологий протокол №1 от «\_\_\_» \_\_\_\_\_ 2025 г.

Рабочая программа утверждена УМС естественнонаучного факультета протокол №1 от «\_\_\_» \_\_\_\_\_ 2025 г.

Рабочая программа утверждена Ученым советом естественнонаучного факультета, протокол № 1 от «\_\_\_» \_\_\_\_\_ 2025 г.

## 1. ЦЕЛИ И ЗАДАЧИ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

### Актуальность изучения дисциплины «NET-программирование»

**1.1 Цели изучения дисциплины** Целью освоения дисциплины "NET-программирование" является формирование у студентов теоретических знаний и практических навыков, необходимых для разработки, отладки и сопровождения приложений, функционирующих в среде .NET Framework и .NET Core. Дисциплина направлена на освоение современных подходов к разработке программного обеспечения, включая объектно-ориентированное программирование, многопоточность, работу с данными и сетевое взаимодействие. В результате изучения дисциплины студенты должны быть готовы к самостоятельной разработке и решению задач, связанных с созданием программных продуктов в рамках .NET экосистемы.

**1.2 Задачи изучения дисциплины** 1. Изучение основ языка C# и платформы .NET, включая CLR, BCL и библиотеки классов.

2. Освоение принципов объектно-ориентированного программирования (ООП) и их применение в разработке .NET приложений.

3. Изучение различных подходов к разработке приложений, включая Windows Forms, ASP.NET, WPF и Xamarin.

4. Освоение работы с базами данных, включая ADO.NET, Entity Framework и LINQ.

5. Изучение принципов разработки многопоточных приложений и сетевого взаимодействия в .NET.

**1.3 В результате изучения дисциплины «NET-программирование» у обучающихся формируются следующие универсальные и общепрофессиональные компетенции:**

Код	Результаты освоения ООП	Индикаторы достижения компетенции	Вид оценочного значения
ПК-1.	Способен проводить обследование организаций и	ИПК-1.1 Использует методики обследования организации и выявления	

	формировать требования к информационной системе	<p>информационных потребностей пользователей.</p> <p>ИПК-1.2 Анализирует деятельность предприятия и выявляет участки, нуждающиеся в автоматизации.</p> <p>ИПК-1.3 Выбирает класс ИС, способы автоматизации, оценивает совокупную стоимость владения ИС, планирует стратегическое и оперативное развитие ИС.</p>	
ПК-2.	Способен разрабатывать и адаптировать прикладное программное обеспечение	<p>ИПК-2.1 Применяет современные технологии разработки и адаптации прикладного ПО.</p> <p>ИПК-2.2 Разрабатывает и адаптирует ПО на современных языках программирования.</p> <p>ИПК-2.3 Применяет современные технологии для разработки веб-приложений.</p>	
ПК-3.	Способен проектировать информационные системы по видам обеспечения	<p>ИПК-3.1 Обосновывает выбор проектных решений по видам обеспечения ИС.</p> <p>ИПК-3.2 Участвует в проектировании экономических ИС и их модулей.</p>	

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

**2.1.** Дисциплина «**NET-программирование**» входит в обязательную часть Блока Дисциплины (модули) ОПОП ВО и является её базовой частью **(Б1.В.05)**. В процессе преподавания данного курса учитываются знания студентов по таким дисциплинам, как история таджикского народа, история первобытного общества, история древнего мира и средних веков, изучавшихся ими в общеобразовательной средней школе.

**2.2** Преподавание данной дисциплины является необходимым для дальнейшего освоения студентами дисциплин в структуре ОПОП бакалавриата по направлению «**Информационная безопасность**».

Таблица 2.

№ п/п	Название дисциплины	Семестр	Место дисциплины в структуре ОПОП
-	—	—	Предшествующая дисциплина
-	—	—	Последующая дисциплина

При освоении данной дисциплины необходимы умения и готовность («входные» знания) обучающегося по дисциплинам, указанных в Таблице 2. Дисциплины взаимосвязаны с данной дисциплиной, они изучаются параллельно. Теоретическими дисциплинами, для которых освоение данной дисциплины необходимо как предшествующее являются:

### **3. СТРУКТУРА И СОДЕРЖАНИЕ КУРСА, КРИТЕРИИ НАЧИСЛЕНИЯ БАЛЛОВ**

Преподавание курса «NET-программирование» планируется студентам Очная формы обучения по направлению «Информационная безопасность».

Объем дисциплины составляет \_\_ зачетные единицы. Всего запланировано 126 часа, из которых: лекции – 16 часов, практические занятия – 14 часов, лабораторные работы 16 часов, иная контактная работа – 32 часа, самостоятельная работа – 62. Всего часов аудиторной нагрузки – 64 часа.

По итогам 5 семестра планируется сдача студентами зачета с оценкой.

#### **3.1 Структура и содержание теоретической части курса**

##### **Лекция 1 Введение в .NET и C#. Обзор платформы .NET.**

История .NET. Архитектура .NET. CLR, BCL. Знакомство с C#. Типы данных, переменные, операторы.

## **Лекция 2 Основы C#. Управляющие конструкции, циклы, методы.**

Условные операторы. Циклы. Методы и функции. Параметры и возвращаемые значения. Область видимости.

## **Лекция 3 Объектно-ориентированное программирование (ООП) в C#. Классы, объекты, наследование.**

Принципы ООП. Классы и объекты. Наследование, полиморфизм, инкапсуляция. Абстрактные классы и интерфейсы.

## **Лекция 4 ООП. Интерфейсы, делегаты и события.**

Интерфейсы. Делегаты и события. Обработка событий. Лямбда-выражения.

## **Лекция 5 Работа с исключениями, обработка ошибок.**

Механизм исключений в C#. Try-catch-finally. Обработка исключений. Создание пользовательских исключений.

## **Лекция 6 Работа со строками и коллекциями.**

Тип string. StringBuilder. Коллекции. Массивы, списки, словари.

## **Лекция 7 Работа с файлами и потоками. Сериализация.**

Ввод-вывод. Работа с файлами. Потоки. Сериализация и десериализация.

## **Лекция 8 Многопоточность и асинхронное программирование.**

Потоки. Задачи. Асинхронное программирование. Async/await.

## **Структура и содержание практической части курса**

### **Практическое занятие 1: Установка среды разработки. Первые программы на C#. (Практика)**

Установка Visual Studio или другого IDE. Создание и запуск простейших программ. Типы данных, переменные.

### **Практическое занятие 2: Управляющие конструкции и циклы. (Практика)**

Решение задач с использованием условных операторов и циклов. Разработка консольных приложений.

### **Практическое занятие 3: Методы и функции. Работа с параметрами. (Практика)**

Создание и вызов методов. Передача параметров по значению и по ссылке. Перегрузка методов.

#### **Практическое занятие 4: Классы и объекты. Создание простых классов. (Практика)**

Создание классов и объектов. Работа со свойствами и методами классов. Инкапсуляция.

#### **Практическое занятие 5: Наследование и полиморфизм. Разработка иерархии классов. (Практика)**

Реализация наследования. Переопределение методов. Использование абстрактных классов и интерфейсов.

#### **Практическое занятие 6: Работа с интерфейсами. Делегаты и события. (Практика)**

Реализация интерфейсов. Работа с делегатами и событиями. Создание обработчиков событий.

#### **Практическое занятие 7: Обработка исключений. (Практика)**

Использование try-catch-finally. Создание и обработка пользовательских исключений.

#### **Практическое занятие 8: Работа со строками и коллекциями. (Практика)**

Работа со строками, StringBuilder. Использование коллекций. Массивы, списки, словари.

### **Структура и содержание лабораторной части курса**

#### **Лабораторная работа 1 Лабораторная работа 1: Знакомство с IDE. Простейшая программа.**

Установка и настройка IDE. Создание и отладка простейшей программы.

#### **Лабораторная работа 2 Лабораторная работа 2: Управляющие конструкции.**

Разработка программы с использованием условных операторов и циклов.

#### **Лабораторная работа 3 Лабораторная работа 3: Методы и функции.**

Создание и использование методов с параметрами и возвращаемыми значениями.

#### **Лабораторная работа 4 Лабораторная работа 4: Классы и объекты.**

Создание классов и объектов, реализация инкапсуляции.

#### **Лабораторная работа 5 Лабораторная работа 5: Наследование и полиморфизм.**

Реализация наследования и полиморфизма, разработка иерархии классов.

#### **Лабораторная работа 6 Лабораторная работа 6: Работа с интерфейсами. Делегаты и события.**

Реализация интерфейсов, работа с делегатами и событиями.

#### **Лабораторная работа 7 Лабораторная работа 7: Обработка исключений.**

Реализация обработки исключений, создание пользовательских исключений.

#### **Лабораторная работа 8 Лабораторная работа 8: Работа со строками и коллекциями.**

Работа со строками, StringBuilder. Использование коллекций. Массивы, списки, словари.

### **Структура и содержание КСР**

#### **КСР 1: Основы С# и ООП.**

Тестирование знаний по основам С# и принципам ООП.

#### **КСР 2: Наследование, полиморфизм, интерфейсы.**

Тестирование знаний по наследованию, полиморфизму и интерфейсам.

#### **КСР 3: Делегаты, события, обработка исключений.**

Тестирование знаний по делегатам, событиям и обработке исключений.

#### **КСР 4: Строки, коллекции, файлы.**

Тестирование знаний по работе со строками, коллекциями и файлами.

#### **КСР 5: Сериализация и десериализация.**

Тестирование знаний по сериализации и десериализации.

### **КСР 6: Многопоточность и асинхронность.**

Тестирование знаний по многопоточности и асинхронному программированию.

### **КСР 7: Работа с базами данных.**

Тестирование знаний по работе с базами данных ADO.NET и EF Core.

### **КСР 8: Windows Forms и ASP.NET Core.**

Тестирование знаний по разработке Windows Forms и ASP.NET Core приложений.

## **Структура и содержание СРС**

### **СРС 1 Изучение теоретического материала по типам данных и переменным.**

Самостоятельное изучение теоретического материала по типам данных и переменным. Подготовка к практическим занятиям.

### **СРС 2 Изучение материала по управляющим конструкциям и циклам.**

Самостоятельное изучение теоретического материала по управляющим конструкциям и циклам. Подготовка к практическим занятиям.

### **СРС 3 Изучение материала по методам и функциям.**

Самостоятельное изучение теоретического материала по методам и функциям. Подготовка к практическим занятиям.

### **СРС 4 Изучение материала по классам и объектам.**

Самостоятельное изучение теоретического материала по классам и объектам. Подготовка к практическим занятиям.

### **СРС 5 Изучение материала по наследованию и полиморфизму.**

Самостоятельное изучение теоретического материала по наследованию и полиморфизму. Подготовка к практическим занятиям.

### **СРС 6 Изучение материала по интерфейсам, делегатам и событиям.**

Самостоятельное изучение теоретического материала по интерфейсам, делегатам и событиям. Подготовка к практическим занятиям.

### **СРС 7 Изучение материала по обработке исключений.**

Самостоятельное изучение теоретического материала по обработке исключений. Подготовка к практическим занятиям.

### **СРС 8 Изучение материала по работе со строками и коллекциями.**

Самостоятельное изучение теоретического материала по работе со строками и коллекциями. Подготовка к практическим занятиям.

### **СРС 9 Изучение материала по работе с файлами и потоками.**

Самостоятельное изучение теоретического материала по работе с файлами и потоками. Подготовка к практическим занятиям.

### **СРС 10 Изучение материала по сериализации и десериализации.**

Самостоятельное изучение теоретического материала по сериализации и десериализации. Подготовка к практическим занятиям.

### **СРС 11 Изучение материала по многопоточности.**

Самостоятельное изучение теоретического материала по многопоточности. Подготовка к практическим занятиям.

### **СРС 12 Изучение материала по асинхронному программированию.**

Самостоятельное изучение теоретического материала по асинхронному программированию. Подготовка к практическим занятиям.

### **СРС 13 Изучение материала по ADO.NET.**

Самостоятельное изучение теоретического материала по ADO.NET. Подготовка к практическим занятиям.

### **СРС 14 Изучение материала по Entity Framework Core.**

Самостоятельное изучение теоретического материала по Entity Framework Core. Подготовка к практическим занятиям.

### **СРС 15 Подготовка к КСР.**

Подготовка к контрольной самостоятельной работе по пройденному материалу.

### **СРС 16 Подготовка к экзамену.**

Подготовка к итоговому экзамену по дисциплине.

### **СРС 17 Разработка учебного проекта (Выбор темы, анализ)**

Самостоятельное выполнение учебного проекта, выбор темы, анализ требований.

**СРС 18 Разработка учебного проекта (Проектирование архитектуры)**

Проектирование архитектуры приложения в рамках учебного проекта.

**СРС 19 Разработка учебного проекта (Реализация)**

Реализация функциональности учебного проекта.

**СРС 20 Разработка учебного проекта (Тестирование)**

Тестирование учебного проекта.

**СРС 21 Разработка учебного проекта (Документирование)**

Документирование учебного проекта.

**СРС 22 Изучение документации по LINQ.**

Самостоятельное изучение документации по LINQ.

**СРС 23 Изучение документации по WPF.**

Самостоятельное изучение документации по WPF.

**СРС 24 Изучение документации по ASP.NET Core MVC.**

Самостоятельное изучение документации по ASP.NET Core MVC.

**СРС 25 Изучение документации по ASP.NET Core Web API.**

Самостоятельное изучение документации по ASP.NET Core Web API.

**СРС 26 Поиск и изучение информации по шаблонам проектирования.**

Поиск и изучение информации по наиболее распространенным шаблонам проектирования.

**СРС 27 Изучение современных подходов к разработке ПО.**

Изучение подходов: Scrum, Kanban, Agile и т.д.

**СРС 28 Изучение документации по LINQ to SQL.**

Самостоятельное изучение документации по LINQ to SQL.

**СРС 29 Изучение основ работы с контейнерами внедрения зависимостей.**

Изучение принципов работы с контейнерами внедрения зависимостей (Dependency Injection).

### СРС 30 Изучение паттернов проектирования.

Изучение популярных паттернов проектирования, их использование в .NET.

### СРС 31 Изучение способов повышения производительности .NET приложений.

Анализ и изучение информации об оптимизации производительности .NET приложений.

## Структура и содержание теоретической, лабораторной части курса, КСР и СРС

Таблица 3.

№ п/п	Наименование темы	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)						Лит-ра	Количество баллов
		Лек	Прак	КСР	Лаб	СРС	ИКР		
1	Введение в .NET и C#. Обзор платформы .NET.	2			2	4		1	12,5
	Установка среды разработки. Первые программы на C#.		2					4	
2	Основы C# и ООП.			2		4		3	12,5
3	Основы C#. Управляющие конструкции, циклы, методы.	2			2			6	12,5
	Управляющие конструкции и циклы.		2			4		5	
4	Наследование, полиморфизм, интерфейсы.			2				4	12,5
5	Объектно-ориентированное программирование (ООП) в C#. Классы, объекты, наследование.	2			2	4		2	12,5
	Методы и функции. Работа с параметрами.		2					5	
6	Делегаты, события, обработка исключений.			2		4		7	12,5
7	ООП. Интерфейсы, делегаты и события.	2			2			2	12,5

	Классы и объекты. Создание простых классов.		2				2	
8	Строки, коллекции, файлы.			2		4		12,5
9	Работа с исключениями, обработка ошибок.	2			2			12,5
	Наследование и полиморфизм. Разработка иерархии классов.		2			2	5	
10	Сериализация и десериализация.			2			3	12,5
11	Работа со строками и коллекциями.	2			2	4		12,5
	Работа с интерфейсами. Делегаты и события.		2				5	
12	Многопоточность и асинхронность.			2		4		12,5
13	Работа с файлами и потоками. Сериализация.	2			2		5	12,5
14	Обработка исключений.		2			4		12,5
15	Работа с базами данных.			2			5	12,5
	Многопоточность и асинхронное программирование.	2			2	2	4	
16	Работа со строками и коллекциями.		2				5	12,5
	Windows Forms и ASP.NET Core.			2		2	2	
<b>Итого:</b>		16	16	16	16	42	0	200

### Формы контроля и критерии начисления баллов

Контроль усвоения студентом каждой темы осуществляется в рамках балльно-рейтинговой системы (БРС), включающей текущий, рубежный и итоговый контроль. Студенты **3-го курса**, обучающиеся по кредитно-рейтинговой системе обучения, могут получить максимально возможное количество баллов - 300. Из них на текущий и рубежный контроль выделяется 200 баллов или 49% от общего количества.

На итоговый контроль знаний студентов выделяется 51% или 100 баллов.

Порядок выставления баллов: 1-й рейтинг (1-7 недели до 12,5 баллов+12,5

баллов (8 неделя – Рубежный контроль №1) = 100 баллов), 2-й рейтинг (9-15 недели до 12,5 баллов+12,5 баллов (16 неделя – Рубежный контроль №2) = 100 баллов), итоговый контроль 100 баллов.

К примеру, за текущий и 1-й рубежный контроль выставляется 100 баллов: лекционные занятия – 21 балл, за практические занятия (КСР, лабораторные) – 31,5 балл, за СРС – 17,5 баллов, требования ВУЗа – 17,5 баллов, рубежный контроль – 12,5 баллов.

В случае пропуска студентом занятий по уважительной причине (при наличии подтверждающего документа) в период академической недели деканат факультета обращается к проректору по учебной работе с представлением об отработке студентом баллов за пропущенные дни по каждой отдельной дисциплине с последующим внесением их в электронный журнал.

Итоговая форма контроля по дисциплине (зачет, экзамен) проводится как в форме тестирования, так и в традиционной (устной) форме. Тестовая форма итогового контроля по дисциплине предусматривает: для естественнонаучных направлений – 10 тестовых вопросов на одного студента, где правильный ответ оценивается в 10 баллов, для гуманитарных направлений – 25 тестовых вопросов, где правильный ответ оценивается в 4 балла. Тестирование проводится в электронном виде, устный экзамен на бумажном носителе с выставлением оценки в ведомости по аналогичной системе с тестированием.

**Таблица 4.**

Неделя	Активное участие на лекционных занятиях, написание конспекта и выполнение других видов работ*	Активное участие на практических (семинарских) занятиях, КСР	СРС Написание реферата, доклада, эссе Выполнение других видов работ	Выполнение положения высшей школы (установленная форма одежды, наличие рабочей папки, а также других пунктов устава высшей школы)	РК №1	Всего
1	2	3	4	5	6	7
1	3	4,5	2,5	2,5	-	12,5
2	3	4,5	2,5	2,5	-	12,5
3	3	4,5	2,5	2,5	-	12,5
4	3	4,5	2,5	2,5	-	12,5
5	3	4,5	2,5	2,5	-	12,5

6	3	4,5	2,5	2,5	-	12,5
7	3	4,5	2,5	2,5	-	12,5
8	-	-	-	-	12,5	12,5
Первый рейтинг	21	31,5	17,5	17,5	12,5	100
1	2	3	4	5	6	7
1	3	4,5	2,5	2,5	-	12,5
2	3	4,5	2,5	2,5	-	12,5
3	3	4,5	2,5	2,5	-	12,5
4	3	4,5	2,5	2,5	-	12,5
5	3	4,5	2,5	2,5	-	12,5
6	3	4,5	2,5	2,5	-	12,5
7	3	4,5	2,5	2,5	-	12,5
8	-	-	-	-	12,5	12,5
Второй рейтинг	21	31,5	17,5	17,5	12,5	100
Итого						200

Формула вычисления результатов дистанционного контроля и итоговой формы контроля по дисциплине за семестр для студентов 3 -го курсов:

$$ИБ = \left[ \frac{(P_1 + P_2)}{2} \right] \cdot 0,49 + Эи \cdot 0,51 ,$$

где ИБ –итоговый балл,  $P_1$ - итоги первого рейтинга,  $P_2$ - итоги второго рейтинга, Эи– результаты итоговой формы контроля (экзамен).

#### **4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ**

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Методы обработки информации» включает в себя:

1. план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;
2. характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;
3. требования к представлению и оформлению результатов самостоятельной работы;

#### 4. критерии оценки выполнения самостоятельной работы.

План-график выполнения самостоятельной работы по дисциплине

##### 4.1. План-график выполнения самостоятельной работы по дисциплине

№	Объем СРС, ч.	Тема СРС	Форма и вид результатов самостоятельной работы	Форма контроля
1	4	Платформа .NET и среда разработки	Вопросы 1–4. Описание технологии разработки, реферат	Опрос
2	4	Язык программирования C#: синтаксис и структура	Вопросы 5–8. Презентация методов	Выступление
3	6	Типы данных и операторы в C#	Вопросы 8–10. Презентация, доклад	Выступление
4	6	Управляющие конструкции и циклы	Вопросы 11–13. Выполнение задания 1 (1–10)	Защита работы, выступление
5	4	Методы и параметры методов	Выполнение задания 1. Конспект, презентация (вопросы 14–15)	Опрос, выступление
6	4	Классы и объекты	Выполнение задания 2	Защита работы
7	6	Наследование и полиморфизм	Вопросы 16–17. Выполнение задания 3	Защита работы
8	6	Интерфейсы и абстрактные классы	Вопросы 16–17. Выполнение задания 4	Защита работы
9	4	Работа с коллекциями	Выполнение задания 5	Защита работы
10	4	Работа с файлами и потоками	Вопросы 18–25. Выполнение задания 6	Защита работы
11	4	Обработка исключений	Вопросы 26–29. Выполнить задания 2 и описать в терминах классов	Опрос, защита работы
12	4	Технология ADO.NET и работа с базами данных	Вопросы 30–31. Реферат. Выполнение задания 7	Защита реферата, защита работы
13	4	Разработка Windows-приложений (WPF/WinForms)	Вопросы 32–37. Презентация	Опрос, выступление
14	4	Основы ASP.NET и веб-разработка	Вопросы 38–40. Выполнение задания 8 (1–4)	Защита работы
15	4	Создание веб-сервисов и API	Вопросы 41–44. Выполнение задания 9	Защита работы
16	4	Итоговый проект на платформе .NET	Вопросы 45–46. Выполнение задания 8 (4–10)	Защита работы

##### 4.2 Характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;

Для выполнения задания, прежде всего, необходимо ознакомиться и

изучить основные положения теоретических материалов соответствующей темы из литературных источников. Они указаны в разделе «Содержание и структура дисциплины». Конспекты и задания можно выполнить в отдельном тетради или в лекционной (практической) тетради в произвольной форме.

#### **4.3 Критерии оценки выполнения самостоятельной работы.**

Критерии оценки выполнения самостоятельной работы является полнота освещения вопроса, логичность изложения, проявления самостоятельность в обработке материала.

#### **4.4. Критерии оценки выполнения самостоятельной работы.**

Самостоятельная работа прививает студентам навыки работы с источниками и учебной литературой, помогает повысить уровень знаний по предмету, которые можно использовать на практике.

Оценка «отлично» выставляется студенту, если индивидуальное задание выполнено полностью и по данной теме защищена лабораторная работа.

Оценка «хорошо» выставляется студенту, если лабораторная работа по теме индивидуального задания защищена, а само индивидуальное задание выполнено с отдельными замечаниями.

Оценка «удовлетворительно» выставляется студенту, если лабораторная работа по теме индивидуального задания защищена, а само индивидуальное задание выполнено не до конца, т.е. не полностью.

Оценка «неудовлетворительно» выставляется студенту, если лабораторная работа по теме индивидуального задания не защищена, а само индивидуальное задание выполнено не до конца, т.е. не полностью.

### **5. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

#### **5.1. Основная литература:**

1. Троелсен Э. Язык программирования C# 8.0 и платформа .NET Core 3.0. Полное руководство. Вильямс, 2020. 1328 стр.
2. Шилдт Г. C# 8.0: Полное руководство. Вильямс, 2019. 1056 стр.

3. Либерти Д., Кирк Д. Программирование на C#. Специальное издание. Питер, 2018. 768 стр.
4. Назарчук Я. Основы C#. Самоучитель. БХВ-Петербург, 2020. 464 стр.
5. Петцольд Ч. Программирование для Windows на C#. Microsoft Press, 2019. 1008 стр.
6. Албахари Дж., Албахари Б. C# 7.0. Справочник. Питер, 2018. 896 стр.
7. Скит Б. C# in Depth. Manning Publications, 2019. 640 стр.

## **5.2. Учебники и учебные пособия в сети Интернет:**

1. 1. Мартин Р. Чистый код. Создание, анализ и рефакторинг. Питер, 2018.
2. Фаулер М. Рефакторинг. Улучшение существующего кода. Символ-Плюс, 2017.
3. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. Питер, 2018.
4. Макконнелл С. Совершенный код. 2-е издание. Питер, 2018.
5. Hunt A., Thomas D. The Pragmatic Programmer. Addison-Wesley Professional, 2019.
6. Fowler M. Patterns of Enterprise Application Architecture. Addison-Wesley Professional, 2014.
7. Larman C. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. Prentice Hall, 2015.

## **5.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

1. 1. Microsoft .NET Documentation: [<https://learn.microsoft.com/>](<https://learn.microsoft.com/>)
2. C# Language Specification: [<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/language-specification/>](<https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/language-specification/>)
3. Stack Overflow: [<https://stackoverflow.com/>](<https://stackoverflow.com/>)
4. GitHub: [<https://github.com/>](<https://github.com/>)

5. .NET Foundation: [<https://dotnetfoundation.org/>](<https://dotnetfoundation.org/>)

#### **5.4. Перечень информационных технологий и программного обеспечения**

Используются лицензионное программное обеспечение ОС Windows -/11 и программное обеспечение открытого доступа (Open source), среды программирования (Denwer, CodeBlock, Dev\_C++ и др.). Для разработки моделей проекта ИС используются CASE – средства: ERWin, Visual UML, Rational Rose и т.д.

### **6. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

Для обеспечения систематической и регулярной работы по изучению дисциплины «NET-программирование» и успешного прохождения текущих и итоговых контрольных испытаний студенту рекомендуется придерживаться следующего порядка обучения:

1. Самостоятельно определить объем времени, необходимого для проработки каждой темы.
2. Регулярно изучать каждую тему дисциплины, используя различные формы индивидуальной работы.
3. Согласовывать с преподавателем виды работы по изучению дисциплины.
4. По завершении отдельных тем передавать выполненные работы (рефераты) преподавателю.

Обучение по дисциплине «NET-программирование» строится следующим образом. На лекциях преподаватель дает общую характеристику рассматриваемого вопроса, различные научные концепции или позиции, которые есть по данной теме. Во время лекции рекомендуется составлять конспект, фиксирующий основные положения лекции и ключевые определения по пройденной теме. Во время лекционного занятия необходимо фиксировать все спорные моменты и проблемы, на которых останавливается преподаватель. Потом именно эти аспекты станут предметом самого пристального внимания и изучения на практических занятиях.

При подготовке к практическому занятию обязательно требуется изучение дополнительной литературы по теме занятия. Без использования нескольких

источников информации невозможно проведение дискуссии на занятиях, обоснование собственной позиции, построение аргументации. Если обсуждаемый аспект носит дискуссионный характер, следует изучить существующие точки зрения и выбрать тот подход, который вам кажется наиболее верным. При этом следует учитывать необходимость обязательной аргументации собственной позиции. Во время практических занятий рекомендуется активно участвовать в обсуждении рассматриваемой темы, выступать с подготовленными заранее рефератами, докладами и презентациями.

Самостоятельная работа должна соответствовать графику прохождения программы дисциплины. Самостоятельная работа по дисциплине «NET-программирование» включает:

- а) работу с литературой;
- б) подготовку устного выступления на практическом занятии;
- в) подготовку к занятию в интерактивной форме;
- г) подготовку реферата с презентацией;
- д) подготовку к дискуссии;
- е) заполнение хронологической таблицы;
- ж) подготовку к текущей и итоговой аттестации по дисциплине.

Для теоретического и практического усвоения дисциплины большое значение имеет самостоятельная работа студентов, которая может осуществляться студентами индивидуально и под руководством преподавателя.

Самостоятельная работа студентов предполагает самостоятельное изучение отдельных тем, дополнительную подготовку студентов к каждому практическому занятию.

Самостоятельная работа студентов является важной формой образовательного процесса. Она реализуется непосредственно в процессе аудиторных занятий, в контакте с преподавателем, а также в библиотеке, дома, при выполнении студентом учебных и творческих задач.

Цель самостоятельной работы студентов - научить студента осмысленно и самостоятельно работать сначала с учебным материалом, затем с научной

информацией, заложить основы самоорганизации и самовоспитания с тем, чтобы привить умение в дальнейшем непрерывно повышать свою квалификацию.

При изучении дисциплины организация самостоятельной работы студентов форм представлена следующим образом:

- 1) внеаудиторная самостоятельная работа;
- 2) аудиторная самостоятельная работа, которая осуществляется под непосредственным руководством преподавателя.

Аудиторная самостоятельная работа может реализовываться при проведении практических занятий и во время чтения лекций.

На практических занятиях различные виды самостоятельной работы позволяют сделать процесс обучения более интересным и поднять активность значительной части студентов в группе.

Для проведения занятий необходимо иметь большой банк заданий, причем эти задания могут быть дифференцированы по степени сложности.

На каждом этапе самостоятельной работы следует разъяснять цели работы, контролировать понимание этих целей студентами, постепенно формируя у них умение самостоятельной постановки задачи и выбора цели.

При чтении лекционного курса непосредственно в аудитории необходимо контролировать усвоение материала основной массой студентов путем проведения экспресс-опросов по конкретным темам.

На практических занятиях различные виды самостоятельной работы позволяют сделать процесс обучения более интересным и поднять активность значительной части студентов в группе.

На практических занятиях нужно не менее 1 часа из двух (50% времени) отводить на самостоятельное рассмотрение заданий.

По результатам самостоятельного рассмотрения задания следует выставлять по каждому занятию оценку. Оценка предварительной подготовки студента к практическому занятию может быть сделана путем экспресс-опроса в течение 5, максимум - 10 минут.

По материалам раздела целесообразно выдавать студенту домашнее задание

и на последнем практическом занятии по разделу подвести итоги его изучения (например, провести контрольную работу), обсудить оценки каждого студента, выдать дополнительные задания тем студентам, которые хотят повысить оценку.

Результативность самостоятельной работы студентов во многом определяется наличием активных методов ее контроля. Существуют следующие виды контроля:

- входной контроль знаний и умений студентов при начале изучения очередной дисциплины;
- текущий контроль, то есть регулярное отслеживание уровня усвоения материала на лекциях, практических занятиях;
- самоконтроль, осуществляемый студентом в процессе изучения дисциплины при подготовке к контрольным мероприятиям;
- итоговый контроль по дисциплине в виде зачета, зачета с оценкой (в устной форме).

## **7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Для реализации дисциплины при кафедре информатики и ИТ РТСУ имеются 4 компьютерных классов. Для занятий используются лицензионное программное обеспечение ОС Windows -7/8/10/11 и программное обеспечение открытого доступа (Open source), среды программирования (Denwer, CodeBlock, Dev\_C++ и др.). Для разработки моделей проекта ИС используются CASE – средства: ERWin, Visual UML, Rational Rose и т.д.

В Университете созданы специальные условия для обучающихся с ограниченными возможностями здоровья - специальные учебники, учебные пособия и дидактические материалы, специальные технические средства обучения коллективного и индивидуального пользования, предоставление услуг ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, проведение групповых и индивидуальных коррекционных занятий, обеспечение доступа в здания организаций и другие условия, без которых невозможно или затруднено освоение дисциплины обучающимися с ограниченными возможностями здоровья.

Обучающимся с ограниченными возможностями здоровья предоставляются бесплатно специальные учебники и учебные пособия, иная учебная литература, а также обеспечивается:

- наличие альтернативной версии официального сайта организации в сети "Интернет" для слабовидящих;
- присутствие ассистента, оказывающего обучающемуся необходимую помощь;
- обеспечение выпуска альтернативных форматов печатных материалов (крупный шрифт или аудиофайлы);
- возможность беспрепятственного доступа обучающихся в учебные помещения, столовые, туалетные и другие помещения организации, а также пребывания в указанных помещениях (наличие пандусов, поручней, расширенных дверных проёмов, лифтов).

## **8. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ**

Промежуточная аттестация осуществляется: для зачета – контрольная работа и опрос. Экзамен проводится в форме тестирования. Защита курсового проекта: представляется пояснительная записка и презентация выступления.

Текущий контроль студентов осуществляется путем защиты лабораторных работ, выполнения самостоятельного задания, обсуждения теоретических вопросов.

Контролирующие материалы по дисциплине содержат:

Контрольные вопросы и задания для текущего контроля знаний по дисциплине.

Тестовые задания для промежуточного контроля знаний по дисциплине;

Методические рекомендации и тематика курсового проектирования.

Также указаны критерии оценки курсового проекта.

**Итоговая система оценок по кредитно-рейтинговой системе с использованием буквенных символов**

<b>Оценка по буквенной системе</b>	<b>Диапазон соответствующих наборных баллов</b>	<b>Численное выражение оценочного балла</b>	<b>Оценка по традиционной системе</b>
<b>A</b>	10	95-100	Отлично
<b>A-</b>	9	90-94	
<b>B+</b>	8	85-89	Хорошо
<b>B</b>	7	80-84	
<b>B-</b>	6	75-79	
<b>C+</b>	5	70-74	Удовлетворительно
<b>C</b>	4	65-69	
<b>C-</b>	3	60-64	
<b>D+</b>	2	55-59	
<b>D</b>	1	50-54	
<b>Fx</b>	0	45-49	Неудовлетворительно
<b>F</b>	0	0-44	

Содержание текущего контроля, промежуточной аттестации, итогового контроля раскрываются в фонде оценочных средств, предназначенных для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС ВО.

ФОС по дисциплине является логическим продолжением рабочей программы учебной дисциплины. ФОС по дисциплине прилагается.