

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Ярославский государственный университет им. П.Г. Демидова

Кафедра вычислительных и программных систем

УТВЕРЖДАЮ

Декан факультета ИВТ

 Д.Ю. Чалый

«18» мая 2020 г.

**Рабочая программа дисциплины**  
«Программирование ASP.NET»

**Направление подготовки**  
01.03.02 Прикладная математика и информатика

**Направленность (профиль)**  
«Прикладная математика и информатика»

**Квалификация выпускника**  
Бакалавр

**Форма обучения**  
очная

Программа рассмотрена  
на заседании кафедры  
от 23 апреля 2020 г., протокол № 8

Программа одобрена НМК  
факультета ИВТ  
протокол № 7 от 17 мая 2020 г.

Ярославль

### 1. Цели освоения дисциплины

Целями дисциплины «Программирование ASP.NET» являются студентов с разработкой на языке C# Web-приложений ASP.NET в среде разработки Visual Studio, а также изучение средств создания, отладки и развертывания Web-приложений и служб, предоставляемых средой разработки Visual Studio .NET.

### 2. Место дисциплины в структуре ОП бакалавриата

Дисциплина «Программирование ASP.NET» относится к вариативной части (дисциплина по выбору) ОП бакалавриата.

Для изучения данной дисциплины студентам рекомендуется предварительно изучить язык программирования C#, иметь навыки объектно-ориентированного программирования на этом языке, начальные знания об идеологии среды исполнения .NET Framework и ее архитектуре и основных механизмах управления программами и ресурсами. Знания и навыки, полученные при изучении данной дисциплины, используются учащимися при изучении других дисциплин компьютерного цикла, а также при подготовке курсовых и дипломных работ.

### 3. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОП бакалавриата

Процесс изучения дисциплины направлен на формирование следующих элементов компетенций в соответствии с ФГОС ВО, ОП ВО и приобретения следующих знаний, умений, навыков и (или) опыта деятельности:

Формируемая компетенция (код и формулировка)	Индикатор достижения компетенции (код и формулировка)	Перечень планируемых результатов обучения
<b>Профессиональные компетенции</b>		
ПК-3 Способен к разработке и проектированию программного обеспечения, к использованию современных технологий программирования	ПК – 3.2 Умеет работать с современными средствами разработки программного обеспечения (ПО), имеет навыки разработки ПО с использованием современных инструментальных средств.	<b>Знать:</b> – технологию LINQ и основы работы с базами данных.  <b>Уметь:</b> – проектировать пользовательский интерфейс и функциональность Web-приложений для работы с базами данных.  <b>Владеть навыками:</b> – программирования, отладки и тестирования Web-приложений для обработки данных в инструментальной среде Visual Studio.

#### 4. Объем, структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зач. ед., 108 акад. час.

№ п/п	Темы (разделы) дисциплины, их содержание	Семестр	Виды учебных занятий, включая самостоятельную работу студентов, и их трудоемкость (в академических часах)						Формы текущего контроля успеваемости  Форма промежуточной аттестации (по семестрам)
			Контактная работа						
			лекции	практические	лабораторные	консультации	аттестационные испытания	самостоятельная работа	
1.	Введение.	8	1					3	
2.	Создание Web-приложений.	8	1		1			3	Выполнение учебных упражнений
3.	Проектирование Web-форм.	8	2		2	1		3	Выполнение учебных упражнений
4.	Жизненный цикл Web-страницы.	8	1		1			3	Выполнение учебных упражнений
5.	Развитие интерфейса и функциональности Web-приложения.	8	1		1			3	Выполнение учебных упражнений
6.	Создание и использование пользовательских элементов управления.	8	1		1			3	Выполнение учебных упражнений
7.	Проверка корректности ввода.	8	1		1			3	Выполнение учебных упражнений
8.	Отладка Web-приложений.	8	1		1			3	Выполнение учебных упражнений
9.	Основы работы с базами данных в Web-приложениях.	8	1		2			3	Выполнение учебных упражнений
10.	Программируемый доступ к базе данных.	8	1		2	1		3	Выполнение учебных упражнений
11.	Использование LINQ в Web-приложениях.	8	1		3	1		3	Выполнение учебных упражнений
12.	Работа с динамическими данными в ASP.NET.	8	1		3	1		3	Выполнение учебных упражнений
13.	Использование AJAX в ASP.NET.	8	1		3			3	Выполнение учебных упражнений

14.	AJAX Control Toolkit.	8			3			3	Выполнение учебных упражнений
15.	Использование служб Microsoft Windows Communication Foundation.	8			2			3	Выполнение учебных упражнений
16.	Управление состоянием Web-приложений.	8			2			3	Выполнение учебных упражнений
17.	Конфигурирование Web-приложения.	8	1		2			3	Выполнение учебных упражнений
18.	Безопасность Web-приложений.	8	1		2			4,7	Выполнение учебных упражнений
	<b>Всего за 8 семестр</b>		<b>16</b>		<b>32</b>	<b>4</b>		<b>55,7</b>	<b>Зачет</b>
	<b>Всего</b>		<b>16</b>		<b>24</b>	<b>4</b>		<b>55,7</b>	

Для самостоятельной работы используются учебные упражнения, представляющие собой создание учебного приложения или библиотеки по подробной инструкции. Упражнения доступны в электронной форме в сети факультета. Распределение их по темам соответствует темам лекций.

**Зачет** состоит из двух частей. В теоретической части студент должен ответить на вопросы по темам, приведенным в приложении. Практическая часть представляет собой разработку Web-приложения, основанного либо на Web-формах, либо на технологии MVC по усмотрению студента.

#### Содержание разделов дисциплины:

- 1. Введение.**
  - 1.1. Обзор технологии ASP.NET.
  - 1.2. Структура и файлы ASP.NET Web-приложения.
- 2. Создание Web-приложений.**
  - 2.1. Инструменты и шаблоны Microsoft Visual Studio для Web-программирования
  - 2.2. Создание Web-приложений ASP.NET в среде разработки Visual Studio
- 3. Проектирование Web-форм.**
  - 3.1. Способы создания Web-форм.
  - 3.2. Категории элементов управления.
  - 3.3. Атрибуты страниц и элементов управления.
  - 3.4. Методы передачи их состояния на сервер.
  - 3.5. Использование страниц с фоновым кодом.
  - 3.6. Добавление процедур обработки событий для Web-серверных элементов управления.
  - 3.7. Создание и использование классов и компонентов.
- 4. Жизненный цикл Web-страницы.**
  - 4.1. Жизненный цикл Web-страницы.
  - 4.2. Событие Page\_Load.
  - 4.3. Postback-процесс.
- 5. Развитие интерфейса и функциональности Web-приложения.**
  - 5.1. Создание и использование ведущих страниц.
  - 5.2. Контент-страницы.
  - 5.3. Вложенные ведущие страницы.

- 6. Создание и использование пользовательских элементов управления.**
  - 6.1. Проектирование интерфейса и функциональности пользовательских элементов управления.
- 7. Проверка корректности ввода.**
  - 7.1. Проверка корректности пользовательского ввода.
  - 7.2. Стандартные и пользовательские элементы управления для проверки корректности ввода.
- 8. Отладка Web-приложений.**
  - 8.1. Использование объектов Trace и Debug для отладки Web-приложений.
  - 8.2. Удаленная отладка.
- 9. Основы работы с базами данных в Web-приложениях.**
  - 9.1. Работа с реляционной моделью данных в Visual Studio .NET.
  - 9.2. Обзор технологии ADO.NET.
  - 9.3. Установление связи с базой данных.
  - 9.4. Связывание серверных элементов с источниками данных.
- 10. Программируемый доступ к базе данных.**
  - 10.1. Использование объекта DataSet.
  - 10.2. Загрузка нескольких таблиц в DataSet и организация связи между ними.
  - 10.3. Использование объекта SqlDataReader для программируемого доступа к базе данных.
- 11. Использование LINQ в Web-приложениях.**
  - 11.1. Использование LINQ для работы с данными.
  - 11.2. Работа с данными в формате XML с использованием провайдера LINQ для XML.
  - 11.3. Работа с данными через провайдера LINQ для SQL и LINQ для сущностей.
- 12. Работа с динамическими данными в ASP.NET.**
  - 12.1. Введение в динамические данные.
  - 12.2. Применение динамических данных в ASP.NET.
  - 12.3. Настройка динамических данных в ASP.NET.
- 13. Использование AJAX в ASP.NET.**
  - 13.1. Технология AJAX и ее реализация в ASP.NET и VisualStudio.
  - 13.2. Расширение стандартных элементов управления ASP.NET возможностями AJAX.
- 14. AJAX Control Toolkit.**
  - 14.1. Установка AJAX Control Toolkit.
  - 14.2. Специализированные элементы управления AJAX из пакета AJAX Control Toolkit.
- 15. Использование служб Microsoft Windows Communication Foundation.**
  - 15.1. Обзор сервисов Windows Communication Foundation.
  - 15.2. Обращение к службам Windows Communication Foundation.
  - 15.3. Работа со службами WCFData.
- 16. Управление состоянием Web-приложений.**
  - 16.1. Управление состоянием.
  - 16.2. Профили ASP.NET.
  - 16.3. Кэширование ASP.NET.
- 17. Конфигурирование Web-приложения.**
  - 17.1. Конфигурация Web-приложения ASP.NET.
  - 17.2. Настройка и развертывание Web-приложения Microsoft ASP.NET.
- 18. Безопасность Web-приложений.**
  - 18.1. Обзор вопросов безопасности Web-приложений.
  - 18.2. Настройка процессов аутентификации и авторизации.

## **5. Образовательные технологии, используемые при осуществлении образовательного процесса по дисциплине**

В процессе обучения используются следующие образовательные технологии:

**Вводная лекция** – дает первое целостное представление о дисциплине и ориентирует студента в системе изучения данной дисциплины. Студенты знакомятся с назначением и задачами курса, его ролью и местом в системе учебных дисциплин и в системе подготовки в целом. Дается краткий обзор курса, история развития науки и практики, достижения в этой сфере, имена известных ученых, излагаются перспективные направления исследований. На этой лекции высказываются методические и организационные особенности работы в рамках данной дисциплины, а также дается анализ рекомендуемой учебно-методической литературы.

**Академическая лекция** (или лекция общего курса) – последовательное изложение материала, осуществляемое преимущественно в виде монолога преподавателя. Требования к академической лекции: современный научный уровень и насыщенная информативность, убедительная аргументация, доступная и понятная речь, четкая структура и логика, наличие ярких примеров, научных доказательств, обоснований, фактов.

Лекции проводятся в компьютерном классе с использованием проектора для демонстрации презентаций и демонстрационных примеров. Такой подход позволяет разобрать достаточно большой объем материала и продемонстрировать весьма объемные примеры кода, что было бы невозможным при использовании традиционных средств. Лекции по времени совмещены с лабораторными занятиями, что дает возможность сразу закрепить прослушанный материал на практике и при необходимости получить консультации лектора.

**Лабораторное занятие** – занятие в компьютерном классе, посвященное освоению конкретных умений и навыков и закреплению полученных на лекции знаний.

Основной формой практической работы студентов по усвоению данного курса является выполнение ими самостоятельных проектов в рамках лабораторных (по каждой теме) и самостоятельных занятий по данному курсу. Все задания, заготовки проектов, демонстрационные и вспомогательные учебные материалы предоставляются учащимся в электронном виде и постоянно доступны в сети факультета. Это позволяет в случае пропуска занятия самостоятельно отработать пропущенную тему.

Промежуточная аттестация производится в форме отчетов студентов по самостоятельно выполненным проектам, окончательная аттестация в форме зачета.

## **6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения и информационных справочных систем (при необходимости)**

В процессе осуществления образовательного процесса используются:

- Windows 7 в качестве основной операционной системы;
- OfficeStd 2013 RUS OLP NL Acdmc 021-10232 для создания слайдов презентаций, используемых для чтения лекций;
- Интегрированная среда разработки Microsoft Visual Studio для подготовки демонстрационных примеров, а также заготовок лабораторных работ и учебных упражнений;
- OfficeStd 2013 RUS OLP NL Acdmc 021-10232 для формирования инструкций по выполнению учебных упражнений, а также текстов материалов для промежуточной и текущей аттестации;
- Microsoft Access, Microsoft SQL Server и учебная БД Northwind в для обучения работе с базами данных.

## **7. Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

а) основная литература:

1. Васильчиков, В. В., Программирование ASP.NET Web Forms [Электронный ресурс] : учебно-методическое пособие / В. В. Васильчиков ; Яросл. гос. ун-т, Ярославль, ЯрГУ, 2021, 207с
2. Васильчиков, В. В. Программирование на языке С# для .NET Framework : курс лекций. Часть 1 : учебное пособие / В. В. Васильчиков ; Яросл. гос. ун-т им. П. Г. Демидова. — Ярославль : ЯрГУ, 2013. — 196 с.
3. Васильчиков, В. В. Программирование на языке С# для .NET Framework : курс лекций. Часть 2 : учебное пособие / В. В. Васильчиков ; Яросл. гос. ун-т им. П. Г. Демидова. — Ярославль : ЯрГУ, 2014. — 200 с.
4. Васильчиков, В. В. Дополнительные вопросы программирования для .NET Framework: учебно-методическое пособие / В. В. Васильчиков ; Яросл. гос. ун-т им. П. Г. Демидова. — Ярославль : ЯрГУ, 2017. — 60 с.

б) дополнительная литература:

5. Эспозито Д. Программирование с использованием Microsoft ASP.NET 3.5 : пер. с англ. / Д. Эспозито. — СПб. : И. Д. Питер, 2013. — 544 с.
6. Эспозито Д. Разработка веб-приложений с использованием ASP.NET и AJAX : пер. с англ. / Д. Эспозито. — СПб. : И. Д. Питер, 2012. — 400 с.
7. Фримен, А. ASP.NET 4.5 с примерами на С# 5.0 для профессионалов : пер. с англ. / А. Фримен. — М. : И. Д. Вильямс, 2014. — 1120 с.
8. Мак-Дональд М., Фримен, А., Шпушта М. Microsoft ASP.NET 4 с примерами на С# 2010 для профессионалов : пер. с англ. / М. Мак-Дональд. — М. : И. Д. Вильямс, 2011. — 1424 с.
9. С# 4.0 и платформа .NET 4 для профессионалов : пер. с англ. / К. Нейгел, Б. Ивсен, Д. Глинн, К. Уотсон. — М. : И. Д. Вильямс, 2011. — 1440 с.
10. Шилдт, Г. С# 4.0 Полное руководство : пер. с англ. / Г. Шилдт. — М. : И. Д. Вильямс, 2011. — 1056 с.
11. Троелсен, Э. Язык программирования С# 2010 и платформа .NET : пер. с англ. / Э. Троелсен. — М. : И. Д. Вильямс, 2011. — 1392 с.
12. Трей, Н. С# 2010 — Ускоренный курс для профессионалов : пер. с англ. / Н. Трей. — М. : И. Д. Вильямс, 2010. — 592 с.
13. Макки, А. Введение в .NET 4.0 и Visual Studio 2010 для профессионалов : пер. с англ. / А. Макки. — М. : И. Д. Вильямс, 2010. — 416 с.

в) программное обеспечение и Интернет-ресурсы:

- Visual Studio 2010.
- Информация по платформе Microsoft .NET Framework, языкам программирования, технологиям, справочная система: <https://msdn.microsoft.com/ru-ru/vstudio/default.aspx>
- Электронно-библиотечная система «Юрайт»(<https://urait.ru/>).
- Электронно-библиотечная система «Лань»(<https://e.lanbook.com/>).

**8. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине**

Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине включает в свой состав специальные помещения:

-учебные аудитории для проведения занятий лекционного типа и практических занятий (семинаров);

-учебные аудитории для проведения лабораторных занятий;

- учебные аудитории для проведения групповых и индивидуальных консультаций,

- учебные аудитории для проведения текущего контроля и промежуточной аттестации;

-помещения для самостоятельной работы;

-помещения для хранения и профилактического обслуживания технических средств обучения.

Специальные помещения укомплектованы средствами обучения, служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа предлагаются наборы демонстрационного оборудования и учебно-наглядных пособий, хранящиеся на электронных носителях и обеспечивающие тематические иллюстрации, соответствующие рабочим программам дисциплин.

Помещения для лабораторных занятий и самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.

Число посадочных мест в лекционной аудитории больше либо равно списочному составу потока, а в аудитории для практических занятий (семинаров) – списочному составу группы обучающихся.

**Автор(ы) :**

Зав. кафедрой

вычислительных и программных систем, к.т.н. В.В. Васильчиков



**Приложение №1**  
**к рабочей программе дисциплины**  
**"Программирование ASP.NET"**

**Фонд оценочных средств**  
**для проведения текущей и промежуточной аттестации студентов**  
**по дисциплине**

**1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

**1.1. Контрольные задания и иные материалы, используемые в процессе текущей аттестации**

**Задания для самостоятельной работы**

Для самостоятельной работы используются учебные упражнения, по каждой теме данного учебного курса предлагается несколько. Большинство из них являются обязательными для выполнения. Заготовки проектов, сценарии работы и пошаговые инструкции для выполнения каждого из упражнений выложены в локальной сети факультета.

Упражнения заключаются в создании проекта Web-приложения и последовательном развитии его функциональности. По завершении всех этапов развития приложения готовый проект сдается преподавателю. Для прохождения аттестации в течение семестра студенты должны выполнить все обязательные упражнения и сдать заверченный проект.

Для оценки качества сданного проекта применяются следующие критерии оценивания

<b>Оценка</b>	<b>Критерии</b>
Отлично	ОПК-3: Умеет выбрать подходящий тип проекта, разработать его архитектуру и грамотно реализовать ее в виде отдельных компонентов на языке C#. Умеет спроектировать Web-интерфейс и реализовать его с использованием подходящих элементов управления. Для добавления необходимой функциональности использует подходящие события и реализует их обработчики. Умеет использовать механизм исключений, грамотно пользуется отладчиком. Программа выполняется для всех тестов. ПК-1: Создает полноценное Web-приложение. Хорошо владеет средствами LINQ для работы с данными. Поясняет код и изменяет его при необходимости
Хорошо	ОПК-3: Умеет выбрать подходящий тип проекта, разработать его архитектуру и реализовать ее в виде отдельных компонентов на языке C#. Умеет спроектировать Web-интерфейс и реализовать его с использованием подходящих элементов управления. Для добавления необходимой функциональности использует события и реализует их обработчики. Умеет использовать механизм исключений, пользуется отладчиком. Программа выполняется для всех тестов, если какой-либо из тестов не выполняется программа дорабатывается. ПК-1: Создает Web-приложение. Владеет средствами LINQ для работы с данными. Поясняет код и изменяет его при

	необходимости с небольшими неточностями
Удовлетворительно	<p>ОПК-3: Умеет выбрать подходящий тип проекта, использовать его для отдельных компонентов на языке C#. В основном может спроектировать Web-интерфейс и реализовать его с использованием подходящих элементов управления. Понимает связь между событием и обработчиком, с некоторыми подсказками может выбрать подходящее событие. Программа выполняется для всех тестов, возможно кроме одного-двух крайних случаев.</p> <p>ПК-1: Создает Web-приложение. Не в полной мере владеет средствами LINQ. С трудом поясняет код, не может изменить код при усложнении или существенном дополнении задачи</p>
Неудовлетворительно	<p>ОПК-3: Не может спроектировать Web-интерфейс и реализовать его. Программа не выполняется для большинства тестов.</p> <p>ПК-1: Не может создать приложение с Web-интерфейсом в среде разработки или создает неработающий проект. Не владеет средствами LINQ. Не может пояснить код и изменить его.</p>

### Примерный список дополнительных вопросов при защите собственного проекта

1. В чем состоит назначение Common Language Runtime?
2. В чем состоит назначение Common Type System?
3. Какие элементы описания метода образуют его сигнатуру?
4. Что такое полиморфизм? Какое он имеет отношение к раннему и позднему связыванию?
5. Перечислите способы, с помощью которых можно определить тип объекта
6. Что такое подписка на событие и как она происходит?
7. Напишите строчку кода для создания объекта WebRequest для URI:  
`http://www.microsoft.com/default.htm?foo=bar`
8. В чем состоит назначение проху применительно к использованию WCF служб?
9. Как в обработчике события Page\_Load определить, в первый раз запрашивается страница или это результат обратной послылки?
10. В чем разница в использовании объектов Trace и Debug?
11. Сколько объектов DataTables может содержать DataSet?
12. Как после создания объектов SqlConnection, SqlDataAdapter, и DataSetobjectотобразить данные в объекте DataGridView?
13. В чем разница между объектами DataSet и DataView?
14. В чем разница между объектами DataSet и SqlDataReader?
15. Как получить доступ к XMLWeb службе или WCF службе из Web формы?
16. Назовите три категории событий, которые обрабатываются в коде файла global.asax
17. В чем разница между объектами Cache и кэшем вывода?
18. Для чего нужны мастер-страницы?
19. Что такое контент-страница?
20. Для чего нужны заказные элементы управления?
21. Какие файлы используются для конфигурирования ASP.NETWeb приложения?
22. Почему можно удалить файлы с фоновым кодом после развертывания ASP.NETWeb приложения?
23. В чем разница между аутентификацией и авторизацией?
24. Назовите три метода проведения аутентификации, поддерживаемых ASP.NET

## **1.2. Список вопросов по темам для проведения зачета по курсу "Программирование ASP.NET"**

### **1. Обзор технологии ASP.NET**

Структура и файлы ASP.NET Web-приложения. Обзор среды разработки Visual Studio .NET. Создание Web-приложений ASP.NET. Создание компонента и его использование на Web-форме.

### **2. Создание ASP.NET Web-форм**

Способы создания ASP.NET Web-форм. Категории элементов управления. Атрибуты страниц и элементов управления. Методы передачи их состояния на сервер. Использование страниц с фоновым кодом. Добавление процедур обработки событий.

### **3. Обработка событий Web-страницы**

Жизненный цикл Web-страницы. Событие Page\_Load, Postback-процесс. Связывание элементов управления на странице. Обработка событий Web-страницы. Использование объектов Trace и Debug для отладки ASP.NET Web-приложений. Удаленная отладка.

### **4. Проверка корректности пользовательского ввода**

Создание и использование элементов управления для проверки корректности ввода. Создание и использование пользовательских элементов управления.

### **5. Технология ADO.NET**

Работа с реляционной моделью данных в Visual Studio .NET. Обзор технологии ADO.NET.

### **6. Отображение данных на Web-форме**

Отображение данных на Web-форме с использованием List-bound элементов управления. Доступ к данным с использованием технологии ADO.NET. Программируемый доступ к базе данных, Использование объектов DataSet и DataView. Загрузка нескольких таблиц в DataSet и организация связи между ними. Использование объекта DataReader для программируемого доступа к базе данных. Хранимые процедуры, типы их параметров, их использование с объектом DataAdapter и объектом Command.

### **7. Работа с XML данными**

Чтение и запись XML данных. XML и объект DataSet. Использование XML WEB-серверных элементов управления. Обзор XML WEB-служб. Доступ к XML WEB-службе посредством протокола Http. Программируемый доступ к XML WEB-службе. Создание и использование XML WEB-служб.

### **8. Управление состоянием Web-приложения**

Файл Global.asax. Использование и хранение переменных приложения и сеанса. Использование cookies.

### **9. Оптимизация, конфигурирование и развертывание ASP.NET Web-приложений**

Оптимизация и конфигурирование ASP.NET Web-приложений. Развертывание ASP.NET Web-приложений.

### **10. Безопасность ASP.NET Web-приложений**

Обеспечение безопасности ASP.NET Web-приложений. Основные методы и сценарии проведения аутентификации.

### **1.3. Правила проведения зачета в восьмом семестре по курсу "Программирование ASP.NET"**

Для получения оценки "зачтено" студент должен выполнить следующие требования:

- в течение семестра выполнить и сдать все обязательные упражнения для самостоятельной работы;
- разработать и защитить собственный проект ASP.NET-приложения, созданного в среде Microsoft Visual Studio, требования к содержанию проекта перечислены ниже;
- в процессе защиты своего проекта правильно ответить на вопросы по данному проекту, а также на вопросы для проведения экзамена, перечисленные выше.

Основные требования к содержанию самостоятельного проекта для сдачи экзамена

- Тема для приложения выбирается студентом самостоятельно.
- Приложение должно быть создано в среде Microsoft Visual Studio на языке C#, причем разработка ведется "с нуля", не допускается переделка ранее созданных кем-либо приложений. Приложение может иметь WebForms(рекомендуется) или MVC.
- Приложение должно иметь не менее трех страниц.
- Должны удовлетворяться следующие обязательные требования:
  - о Работа любым типом базы данных: SQL-сервер, файл SQL, XML и т.п. Автоматически создаваемая база данных для аутентификации пользователей не считается.
  - о Наличие аутентификации пользователей и разных ролей (аутентифицированный пользователь, анонимный пользователь, администратор). Разные возможности для разных ролей.
  - о Использование LINQ для работы с данными.
- Дополнительные требования, не являющиеся обязательными, но влияющие на оценку качества проекта:
  - о Использование AJAX (хотя бы частично-страничной отправки).
  - о Использование мастер-страниц.
  - о Использование заказных контролов.
  - о Использование контролов валидации.
  - о Использование WCF-служб.
  - о Использование средств управления состоянием приложения и/или сеанса.
- Оценка за проект зависит от степени его развитости, качества интерфейса и полезности для пользователя.

В случае неудовлетворительной защиты проекта, либо его отсутствия, студент должен сдать зачет в специально назначенное для этого время. Ему потребуется в учебном классе в отведенный временной промежуток создать и отладить в среде Microsoft Visual Studio Web-приложение в соответствии с полученным заданием.

В случае сдачи экзамена в такой форме оценка "зачтено" выставляется только в случае удовлетворения приложения всем требованиям, перечисленным в задании.

## **2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкалы оценивания**

### **2.1. Шкала оценивания сформированности компетенций и ее описание**

Оценивание уровня сформированности компетенций в процессе освоения дисциплины осуществляется по следующей трехуровневой шкале:

*Пороговый уровень* - предполагает отражение тех ожидаемых результатов, которые определяют минимальный набор знаний и (или) умений и (или) навыков, полученных студентом в результате освоения дисциплины. Пороговый уровень является обязательным уровнем для студента к моменту завершения им освоения данной дисциплины.

*Продвинутый уровень* - предполагает способность студента использовать знания, умения, навыки и (или) опыт деятельности, полученные при освоении дисциплины, для решения профессиональных задач. Продвинутый уровень превосходит пороговый уровень по нескольким существенным признакам.

*Высокий уровень* - предполагает способность студента использовать потенциал интегрированных знаний, умений, навыков и (или) опыта деятельности, полученных при освоении дисциплины, для творческого решения профессиональных задач и самостоятельного поиска новых подходов в их решении путем комбинирования и использования известных способов решения применительно к конкретным условиям. Высокий уровень превосходит пороговый уровень по всем существенным признакам.

## 2.2. Перечень компетенций, этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

Код компетенции	Форма контроля	Этапы формирования (№ темы (раздела))	Показатели оценивания	Шкала и критерии оценивания компетенций на различных этапах их формирования		
				Пороговый уровень	Продвинутый уровень	Высокий уровень
Общепрофессиональные компетенции						
ОПК-3	Проверка выполнения учебных упражнений по темам № 1-18 Зачет	1-18	<b>Знать:</b> – технологию ASP.NET, ее архитектуру и основные механизмы управления; – технологию ADO.NET. – технологию LINQ и основы работы с базами данных. <b>Уметь:</b> – создавать Web-приложения, Web-службы WCF и отдельные компоненты ASP.NET; – оптимизировать и конфигурировать ASP.NET Web-приложения. <b>Владеть:</b> – навыками программирования, отладки и	1. Знание основы технологии ASP.NET.  2. Знание основных технологий работы с данными.  3. Знание основы технологии LINQ для работы с данными.  4. Умение проектировать пользовательский интерфейс и функциональность Web-	1. Знание основы технологии ASP.NET. Понимание основных механизмов управления Web-приложения.  2. Знание основных технологий работы с данными, преимущества ADO.NET по сравнению с прежними технологиями.  3. Знание технологии LINQ для работы с данными, конструкции и вариации LINQ.  4. Умение проектировать пользовательский интерфейс и функциональность Web-	1. Знание основы технологии ASP.NET. Понимание основных механизмов функционирования и управления Web-приложения.  2. Знание технологий работы с данными, преимущества ADO.NET по сравнению с более ранними технологиями. Понимание ее архитектуры, устройства и назначения основных классов.  3. Знание технологии LINQ для работы с данными, конструкции и вариации LINQ. Умение использовать динамические запросы LINQ.  4. Умение проектировать пользовательский интерфейс и функциональность Web-приложений ASP.NET.

			тестирования ASP.NET Web-приложений в инструментальной среде Visual Studio.	приложений ASP.NET.  5. Умение реализовывать функциональность Web-служб ASP.NET.  6. Знание основных приемов отладки и тестирования Web-приложений и Web-служб.	приложений ASP.NET. Создание собственных элементов управления, работа с мастер-страницами.  5. Умение реализовывать функциональность Web-служб. Развертывание Web-служб.  6. Знание приемов отладки и тестирования Web-приложений и Web-служб, средств Visual Studio для этих целей.	Создание собственных элементов управления, работа с мастер-страницами. Знание и умение использовать технологии AJAX.  5. Умение реализовывать функциональность Web-служб. Развертывание Web-служб. Знание технологий и методов обеспечения безопасности.  6. Знание приемов отладки и тестирования Web-приложений и Web-служб, средств Visual Studio для этих целей. Конфигурирование Web-приложений.
--	--	--	---	---	---	---

#### Профессиональные компетенции

ПК-1	Проверка выполнения учебных упражнений по темам № 9 - 12 Экзамен	9 - 12	<b>Знать:</b> – способы хранения в памяти больших объемов данных – технологию LINQ и основы работы с базами данных. <b>Уметь:</b> – проектировать пользовательский интерфейс и функциональность Web-приложений, обрабатывающих	1. Знание способов хранения в памяти больших объемов данных. Знание конструкций языка C# для работы с массивами и коллекциями и основных классов.  2. Знание технологии LINQ и основ работы с базами данных.	1. Знание способов хранения в памяти больших объемов данных. Знание конструкций языка C# для работы с массивами и коллекциями, основных классов и интерфейсов.  2. Знание технологии LINQ, основных вариаций LINQ и основ работы с базами данных.	1. Знание способов хранения в памяти больших объемов данных. Знание конструкций языка C# для работы с массивами и коллекциями, основных классов и интерфейсов, в том числе параметризованных.  2. Знание технологии LINQ, основных вариаций LINQ и основ работы с базами данных. Умение работать как с реляционными, так и с нереляционными источниками данных.
------	---	--------	--	--	---	---

			<p>большой объем данных</p> <p><b>Владеть:</b></p> <p>– навыками программирования, отладки и тестирования Web-приложений для работы с большим объемом данных.</p>	<p>3. Программирование интерфейса и требуемой функциональности Web-приложений для работы с большим объемом данных.</p>	<p>3. Программирование интерфейса и требуемой функциональности Web-приложений для работы с большим объемом данных.</p>	<p>3. Программирование интерфейса и требуемой функциональности Web-приложений для работы с большим объемом данных разного типа.</p>
--	--	--	---	--	--	---



### **3. Методические рекомендации преподавателю по процедуре оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Целью процедуры оценивания является определение степени овладения студентом ожидаемыми результатами обучения (знаниями, умениями, навыками и (или) опытом деятельности).

Процедура оценивания степени овладения студентом ожидаемыми результатами обучения осуществляется с помощью методических материалов, представленных в разделе «Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций»

#### **3.1 Критерии оценивания степени овладения знаниями, умениями, навыками и (или) опытом деятельности, определяющие уровни сформированности компетенций**

Пороговый уровень (общие характеристики):

- владение основным объемом знаний по программе дисциплины;
- знание основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы без существенных ошибок;
- владение инструментарием дисциплины, умение его использовать в решении стандартных (типовых) задач;
- способность самостоятельно применять типовые решения в рамках рабочей программы дисциплины;
- усвоение основной литературы, рекомендованной рабочей программой дисциплины;
- знание базовых теорий, концепций и направлений по изучаемой дисциплине;
- самостоятельная работа на практических и лабораторных занятиях, периодическое участие в групповых обсуждениях, достаточный уровень культуры исполнения заданий.

Продвинутый уровень (общие характеристики):

- достаточно полные и систематизированные знания в объеме программы дисциплины;
- использование основной терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать выводы;
- владение инструментарием дисциплины, умение его использовать в решении учебных и профессиональных задач;
- способность самостоятельно решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в базовых теориях, концепциях и направлениях по изучаемой дисциплине и давать им сравнительную оценку;
- самостоятельная работа на практических и лабораторных занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

Высокий уровень (общие характеристики):

- систематизированные, глубокие и полные знания по всем разделам дисциплины;

- точное использование терминологии данной области знаний, стилистически грамотное, логически правильное изложение ответа на вопросы, умение делать обоснованные выводы;
- безупречное владение инструментарием дисциплины, умение его использовать в постановке и решении научных и профессиональных задач;
- способность самостоятельно и творчески решать сложные задачи (проблемы) в рамках рабочей программы дисциплины;
- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой дисциплины;
- умение ориентироваться в основных теориях, концепциях и направлениях по изучаемой дисциплине и давать им критическую оценку;
- активная самостоятельная работа на практических и лабораторных занятиях, творческое участие в групповых обсуждениях, высокий уровень культуры исполнения заданий.

### **3.2 Описание процедуры выставления оценки**

В зависимости от уровня сформированности каждой компетенции по окончании освоения дисциплины студенту выставляется оценка. Для дисциплин, изучаемых в течение нескольких семестров, оценка может выставляться не только по окончании ее освоения, но и в промежуточных семестрах. Вид оценки («отлично», «хорошо», «удовлетворительно», «неудовлетворительно», «зачтено», «не зачтено») определяется рабочей программой дисциплины в соответствии с учебным планом.

Высокий уровень формирования компетенций соответствует оценке «отлично» за самостоятельные, контрольные работы и экзаменационную работу.

Продвинутый уровень формирования компетенций соответствует оценке «хорошо» за самостоятельные, контрольные работы и экзаменационную работу.

Пороговый уровень формирования компетенций соответствует оценке «удовлетворительно» за самостоятельные, контрольные работы и экзаменационную работу.

Оценка «отлично» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована на высоком уровне.

Оценка «хорошо» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на продвинутом уровне.

Оценка «удовлетворительно» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «неудовлетворительно» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

Оценка «зачет» выставляется студенту, у которого каждая компетенция (полностью или частично формируемая данной дисциплиной) сформирована не ниже, чем на пороговом уровне.

Оценка «незачтено» выставляется студенту, у которого хотя бы одна компетенция (полностью или частично формируемая данной дисциплиной) сформирована ниже, чем на пороговом уровне.

**Приложение №2**  
**к рабочей программе дисциплины**  
**"Программирование ASP.NET"**

**Методические указания для студентов по освоению дисциплины**

Основной формой изложения учебного материала по дисциплине "Программирование ASP.NET" являются лекции, проводимые в виде электронных презентаций, что позволяет сделать материал лекций более наглядными, улучшает информативность и понимание изучаемого курса.

По большинству тем предусмотрены учебные упражнения, на которых происходит закрепление лекционного материала путем применения его к конкретным задачам и получение навыков разработки и отладки Web-приложений ASP.NET. Выполнение этих упражнений, а также выполнение индивидуальных заданий позволяет не только понять и закрепить теоретический материал, но и приобрести навык анализа предметной области и построения Web-приложений ASP.NET на языке C#.

Для успешного освоения дисциплины очень важно решение достаточно большого количества задач, как в аудитории, так и самостоятельно в качестве домашних заданий. Примеры решения задач разбираются и обсуждаются на лекциях и лабораторных занятиях. Основная цель решения задач – помочь усвоить фундаментальные понятия и основы концепции объектно-ориентированного анализа и проектирования. Для решения всех задач необходимо знать и понимать лекционный материал. Поэтому в процессе изучения дисциплины рекомендуется регулярное повторение пройденного лекционного материала. Материал, законспектированный на лекциях, необходимо дома еще раз прорабатывать и при необходимости дополнять информацией, полученной при выполнении лабораторных работ или из учебной литературы.

Большое внимание должно быть уделено выполнению домашней работы. В качестве заданий для самостоятельной работы дома студентам предлагаются задачи, аналогичные разобранным на лекциях и лабораторных занятиях или немного более сложные, которые являются результатом объединения нескольких базовых задач.

Для проверки и контроля усвоения теоретического материала, приобретенных практических навыков Web-программирования, в течение всего периода обучения проводятся консультации и разбор самостоятельного выполнения индивидуальных заданий.

В конце семестра студенты сдают зачет. Зачет предполагает самостоятельную разработку приложения и ответы на теоретические вопросы.

Освоить вопросы, излагаемые в процессе изучения дисциплины "Программирование ASP.NET" самостоятельно студенту достаточно сложно. Это связано с отсутствием опыта Web-программирования и с особенностями исполнения такого рода приложений. Поэтому посещение всех аудиторных занятий является совершенно необходимым. Без упорных и регулярных занятий в течение семестра сдать зачет по итогам изучения дисциплины студенту будет сложно.

### **Учебно-методическое обеспечение самостоятельной работы студентов по дисциплине**

Для самостоятельной работы особенно рекомендуется использовать учебную литературу, указанную в разделе № 7 данной рабочей программы.

Также для подбора учебной литературы рекомендуется использовать широкий спектр интернет-ресурсов:

1. Электронно-библиотечная система «Университетская библиотека online» ([www.biblioclub.ru](http://www.biblioclub.ru)) - электронная библиотека, обеспечивающая доступ к наиболее востребованным материалам-первоисточникам, учебной, научной и художественной литературе ведущих издательств (\*регистрация в электронной библиотеке – только в сети университета. После регистрации работа с системой возможна с любой точки доступа в Internet.).

2. Для самостоятельного подбора литературы в библиотеке ЯрГУ рекомендуется использовать:

1. Личный кабинет ([http://lib.uniylar.ac.ru/opac/bk\\_login.php](http://lib.uniylar.ac.ru/opac/bk_login.php)) дает возможность получения on-line доступа к списку выданной в автоматизированном режиме литературы, просмотра и копирования электронных версий изданий сотрудников университета (учеб. и метод. пособия, тексты лекций и т.д.) Для работы в «Личном кабинете» необходимо зайти на сайт Научной библиотеки ЯрГУ с любой точки, имеющей доступ в Internet, в пункт меню «Электронный каталог»; пройти процедуру авторизации, выбрав вкладку «Авторизация», и заполнить представленные поля информации.

2. Электронная библиотека учебных материалов ЯрГУ ([http://www.lib.uniylar.ac.ru/opac/bk\\_cat\\_find.php](http://www.lib.uniylar.ac.ru/opac/bk_cat_find.php)) содержит более 2500 полных текстов учебных и учебно-методических материалов по основным изучаемым дисциплинам, изданных в университете. Доступ в сети университета, либо по логину/пароллю.

3. Электронная картотека «Книгообеспеченность» ([http://www.lib.uniylar.ac.ru/opac/bk\\_bookreq\\_find.php](http://www.lib.uniylar.ac.ru/opac/bk_bookreq_find.php)) раскрывает учебный фонд научной библиотеки ЯрГУ, предоставляет оперативную информацию о состоянии книгообеспеченности дисциплин основной и дополнительной литературой, а также цикла дисциплин и специальностей. Электронная картотека «Книгообеспеченность» доступна в сети университета и через Личный кабинет.