

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Проректор по учебной работе

А. Л. Толстий

31 июня

Регистрационный № УД-084 /уч.

ЯЗЫК ПРОГРАММИРОВАНИЯ C#. ОСНОВЫ ASP.NET

Учебная программа учреждения высшего образования
по учебной дисциплине для специальности

1-31 03 08 Математика и информационные технологии

2015 г.

Учебная программа составлена на основе ОСВО 1-31 03 08-2013, введенного в действие 30.08.2013; УП для специальности 1-31 03 08 “Математика и информационные технологии” рег. № G31-134/уч. от 30.05.2013 г.

СОСТАВИТЕЛИ:

Кравчук Анжелика Ивановна, доцент кафедры веб-технологий и компьютерного моделирования Белорусского государственного университета, кандидат физико-математических наук, доцент;

Суздаль Станислав Валерьевич, доцент кафедры веб-технологий и компьютерного моделирования Белорусского государственного университета, кандидат физико-математических наук.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой веб-технологий и компьютерного моделирования
(протокол № 10 от 14.05.2015г.);

Учебно-методической комиссией механико-математического факультета
Белорусского государственного университета
(протокол № 6 от 26.05.2015г.).



С.В.Суздаль



В.С. Романчик

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

.NET Framework представляет собой программную платформу, выпущенную компанией Microsoft в 2002 году, для создания приложений на базе семейства операционных систем Windows, а также некоторых операционных систем производства не Microsoft, таких как Mac OS X и различные дистрибутивы Unix и Linux. Платформа .NET

- обеспечивает возможность взаимодействия с существующим программным кодом;
- позволяет поддерживать многочисленные языки программирования (C#, Visual Basic, F#, S# и т.д.);
- имеет общий исполняющий механизм, используемый всеми поддерживаемыми .NET языками;
- характеризуется полной интеграцией языков – поддерживается межязыковое наследование, обработка исключений и отладка кода;
- имеет обширную библиотеку базовых классов, позволяющую избегать сложностей, связанных с выполнением прямых вызовов к API-интерфейсу, и предлагает согласованную объектную модель, которую могут использовать все поддерживающие .NET языки.

Технология ASP.NET является составной частью платформы Microsoft .NET и развитием Active Server Page (ASP). Технология представляет собой универсальную платформу для разработки веб-приложений корпоративного уровня. На данный момент последней версией этой технологии является ASP.NET 4.5. ASP.NET предлагает новую модель программирования и инфраструктуру, которые позволяют разрабатывать защищенные и масштабируемые решения. В ASP.NET следует отметить следующие функциональные возможности:

- возможности администрирования. Параметры, определяющие конфигурацию ASP.NET, хранятся в текстовом файле. Это позволяет выполнять изменение конфигурации ASP.NET посредством любого текстового редактора. Все произведенные изменения вступают в силу немедленно;
- средства безопасности. ASP.NET предлагает разработчику веб-приложений несколько типовых схем аутентификации и авторизации пользователей. Разработчик может использовать в своем приложении любую из предлагаемых схем или заменить их другими;
- простота развертывания. Развертывание ASP.NET-приложений выполняется путем копирования файлов приложения в специальную папку на веб-сервере. Перезапуск веб-сервера при этом не требуется;
- высокая производительность. ASP.NET имеет дело со скомпилированным кодом. Благодаря этому ASP.NET получает возможность эффективно

использовать различные механизмы оптимизации кода (например, механизмы раннего связывания или оптимизация под конкретную платформу);

- гибкое кэширование. ASP.NET может выполнять кэширование страниц данных (как страницу целиком, так и ее часть) в соответствии с нуждами приложения;

- поддержка национальных языков. Поскольку ASP.NET использует Unicode, разработчики имеют обширные возможности для применения в своих приложениях национальных алфавитов;

- поддержка мобильных устройств. Microsoft заявляет, что ASP.NET поддерживается любым браузером, запущенным на любом устройстве;

- доступность и масштабируемость. ASP.NET разрабатывалась в расчете на использование в крупных кластерных системах. Реализованные в рамках IIS 6.0 механизмы позволяют гарантировать высокую степень доступности приложений. Если с процессом, в рамках которого выполняется приложение, возникнут проблемы, система самостоятельно запустит новый процесс, который возьмет на себя задачи обслуживания запросов пользователей;

- возможности отладки. ASP.NET обеспечивает возможность трассировки и отладки кода приложений. При этом возможна как локальная, так и удаленная отладка при помощи специальных инструментов отладки .NET Framework;

- интеграция с .NET Framework. ASP.NET является частью платформы .NET Framework. Разработчики могут использовать возможности, предоставляемые этой платформой при создании приложений;

- совместимость с существующими ASP-приложениями. Развертывание ASP.NET в рамках IIS не влияет на функционирование уже работающих ASP-приложений. ASP-приложения и ASP.NET-приложения могут сосуществовать на одном сервере, не мешая друг другу.

Наличие перечисленных функциональных возможностей делает технологию ASP.NET сверхмощным инструментом для разработки веб-приложений корпоративного уровня, разработчиков на основе ASP.NET востребованными на рынке труда, а предлагаемый учебный курс актуальным для изучения студентами.

Дисциплина специализации *«Язык программирования C#. Основы ASP.NET»* разработана для студентов III курса очной формы обучения по специальности 1-31 03 08 «Математика и информационные технологии» механико-математического факультета Белорусского государственного университета.

Изучение дисциплины *«Язык программирования C#. Основы ASP.NET»* базируется на знаниях, полученных при изучении университетских курсов «Методы программирования и информатика», «Веб-программирование», «Информационные технологии», «Базы данных».

Специальный курс «Язык программирования С#. Основы ASP.NET» имеет прикладную направленность. **Основными целями** курса «Язык программирования С#. Основы ASP.NET» являются:

- дать теоретическую подготовку по языку С#;
- дать теоретическую подготовку по базовым технологиям .NET Framework;
- дать практические навыки создания программного обеспечения на примере разработки Web-приложений с использованием новейших версий фреймворка ASP.NET MVC;
- познакомить с принципами первичного тестирования разрабатываемого продукта.

Основными задачами курса «Язык программирования С#. Основы ASP.NET» являются:

- научить строить архитектуру разрабатываемого приложения в соответствии с принципами ООП и SOLID;
- научить пользоваться сторонними программами и инструментами в процессе разработки реальных программных продуктов;
- научить студентов создавать приложения на языке программирования С#;
- научить студентов проектировать и разрабатывать web-системы с использованием технологий ASP.NET;

В результате изучения дисциплины «Язык программирования С#. Основы ASP.NET» студенты будут

знать:

- синтаксис простых конструкций языка С#;
- основные концепции платформы .NET и ее базовой части Framework;
- базовые принципы разработки web-приложений на основе технологии ASP.NET;

уметь:

- создавать и вызывать методы;
- перехватывать, обрабатывать и выбрасывать исключения;
- создавать и использовать новые типы (перечисления, классы и структуры) и понимать различия между ссылочными и значимыми типами данных;
- выполнять простые операции для взаимодействия с файловой системой;
- контролировать область видимости и время жизни членов типа;
- использовать наследование для разработки новых ссылочных типов данных;
- управлять временем жизни объектов и контролировать использование ресурсов;
- определять свойства и индексы для инкапсулирования данных и определять операторы для этих данных;

- отделять действие от метода, реализующего это действие, и использовать отделенные операции для обработки асинхронных событий;
- использовать коллекции для объединения данных и использовать обобщения для реализации типизированных классов, структур, интерфейсов и методов;
- реализовывать собственные классы коллекций, поддерживающие возможность перебора элементов;
- создавать запросы к данным, находящимся в памяти, с использованием LINQ;
- выполнять первичное тестирование проекта на этапах разработки;
- планировать и внедрять архитектуру проекта в соответствии с принципами ООП;
- создавать веб-приложения в интегрированной среде разработки и развертывания Microsoft Visual Studio на языке C#;
- создавать веб-формы ASP.NET;
- создавать веб-системы на основе фреймворка ASP.NET MVC, реализующий шаблон “Модель-Представление-Контроллер”;
- решать задачи, требующие доступа к данным, используя LINQ;
- управлять состоянием в веб-приложениях;
- настраивать и развертывать веб-приложения ASP.NET на веб-сервере IIS;

владеть:

- навыками работы в среде разработки Microsoft Visual Studio;
- навыками разработки web-приложений в среде Microsoft Visual Studio.

Курс «Язык программирования C#. Основы ASP.NET» рассчитан на 76 часов в 6 семестре, из них 50 аудиторных часов, в том числе 20 часов лекций, 28 часов лабораторных занятий и 2 часа аудиторного контроля УСР.

	Экзамен, семестр	Зачет, семестр	Всего часов	В том числе аудиторных	Из них			
					лекций	семинар- ских занятий	лабораторных занятий	УСР
III курс		6	76	50	20		28	2

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Язык C# и основы платформы .NET.

Тема 1.1. Введение в .NET Framework. Синтаксис и основные программные конструкции языка C#. Цели .NET Framework и возможности, доступные при создании приложений с помощью C# и Visual Studio. Знакомство с платформой .NET Framework новейшей версии. Сборки в .NET, локальные и глобальные сборки, GAC. Создание проектов с Visual Studio новейшей версии. Типы проектов. Написание приложений на C#. ILDASM, .NET Reflector. Построение графических приложений. Отладка исходного кода, документирование приложения.

Основные типы данных и программные конструкции языка C#, их синтаксис и семантику. Приведение типов. Литералы. Объявление переменных и присваивание значений. Использование выражений и операций языка в C#. Использование операторов выбора, цикла, условного оператора. Массивы. Создание и использование массивов.

Тема 1.2. Классы и объекты в C#. Синтаксис объявления классов. Конструкторы. Поля и методы, их определение и использование. Спецификация необязательных и выходных параметров, методы с переменным числом параметров. Свойства, их объявление и использование.

Тема 1.3. Инкапсуляция данных и методов. Модификаторы доступа в C# и реализация принципов инкапсуляции. Модификатор static и его использование. Статические конструкторы. Индексаторы. Перегрузка операторов. Операторы преобразования типов.

Тема 1.4. Наследование в C#. Интерфейсы и абстрактные классы. Наследование классов. Виртуальные методы и полиморфизм. Интерфейсы и абстрактные классы в .NET Framework, а также возможности их использования для упрощения сложных проблем сокращения дублирования кода и ускорения разработки. Создание и использование структур и перечислений.

Тема 1.5. Исключения и обработка исключений. Знакомство с исключениями и их классами. Как можно реализовывать в приложениях их эффективную обработку. Обработка исключений. Возникновение исключений. Пользовательские исключения.

Тема 1.6. Работа с файловой системой. Как читать из файлов и записывать в файлы с помощью классов .NET Framework. Возможные подходы, которые можно при предпринимать, читая и записывая различные форматы данных. Доступ к файловой системе. Чтение и запись файлов с помощью потоков.

Тема 1.7. Делегаты и события. Объявление и использование делегатов. Ковариация и контравариация делегатов. События и их обработка. Как использовать события для информирования приложений об изменении или заметном явлении в типе. Лямбда-выражения. Анонимные методы.

Тема 1.8. Коллекции и обобщенные типы. Введение в LINQ. Обобщения и возможности их использования для поддержания целостности типа и избегания проблем, связанных с отсутствием безопасности типов базовых классов коллекций. Знакомство с обобщенными интерфейсами и классами коллекциями .NET Framework, и возможностью разработки пользовательских классов коллекций, поддерживающих конструкции языка C#, обеспечивающих перечисление и инициализацию. Использование коллекций. Создание и использование обобщенных типов. Определение обобщенных интерфейсов и понимание вариантности. Использование обобщенных методов и делегатов. Реализация пользовательского класса коллекции.

Знакомство с LINQ (Language-Integrated Query) запросами и возможностями их использования для обработки данных в приложениях Microsoft .NET Framework. Разница между статическими и динамическими запросами LINQ, возможности динамического LINQ для создания гибких запросов во время выполнения. Использование LINQ методов расширения и выражений запросов. Создание динамических запросов LINQ.

Раздел 2. Технологии ASP.NET.

Тема 2.1. Введение в ASP.NET, Web формы и ASP.NET MVC. Общие концепции ASP.NET. Клиент-серверное взаимодействие в ASP.NET. Компоненты приложения ASP.NET. Модель компиляции ASP.NET. Структура приложений ASP.NET Web форм и ASP.NET MVC.

Тема 2.2. Контроллеры в ASP.NET MVC. Класс контроллера и методы действий (Action методы). Маршрутизация, типы запросов для Action методов (атрибуты [HttpGet], [HttpPost], [HttpDelete] или [HttpPut]). Параметры Action методов и способы их передачи в метод. Получение данных из контекста запроса. Встроенные классы, наследники ActionResult, возвращаемые Action методами. Ошибки и обработка ошибок. Отправка файлов. Асинхронные контроллеры и методы.

Тема 2.3. Представления в ASP.NET MVC. Реализация представлений в ASP.NET MVC, фреймворк Razor. Шаблоны представлений и их типы (Scaffold template). Передача данных из представления в контроллер. Типизированные представления. Основы синтаксиса фреймворка представлений Razor. Встроенные html-хелперы и их применения (helper methods). Формы, элементы управления.

Тема 2.4. Модели в ASP.NET MVC. Модели и доступ к данным. Создание классов моделей. Аннотации данных для отображения свойств. Основы фреймворка Entity. Шаблонные хелперы для работы с данными. Основы ADO.NET. Выполнение SQL операторов в ADO.NET, хранимые процедуры в ADO.NET.

Тема 2.5. Проверка пользовательского ввода. Валидация данных. Авторизация и аутентификация. Серверная и клиентская валидации данных. Атрибуты валидации. Валидация модели в контроллере. Обработка ошибок валидации. Пользовательская валидация. Фильтры и их базовые типы. Фильтры авторизации. Атрибуты аутентификации.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	Язык C# и основы платформы .NET	12			14	1		
1.1	Введение в .NET Framework. Синтаксис и основные программные конструкции языка C#	2			2			Отчет по лабораторной работе
1.2	Классы и объекты в C#	2			2			Отчет по лабораторной работе
1.3	Инкапсуляция данных и методов в C#	2			2			Отчет по лабораторной работе
1.4	Наследование в C#. Интерфейсы и абстрактные классы.	2			2			Отчет по лабораторной работе
1.5	Исключения и обработка исключений	1			1			Отчет по лабораторной работе
1.6	Работа с файловой системой.	1			1			Отчет по лабораторной работе
1.7	Делегаты и события.	1			2			Отчет по лабораторной работе
1.8	Коллекции и обобщенные типы. Введение в LINQ	1			2		1	Отчет по лабораторной работе

2	Технологии ASP.NET	8				14	1	
2.1	Введение в ASP.NET, Web формы и ASP.NET MVC	1				2		Отчет по лабораторной работе
2.2	Контроллеры в ASP.NET MVC	1				2		Отчет по лабораторной работе
2.3	Представления в ASP.NET MVC	2				4	1	Отчет по лабораторной работе
2.4	Модели в ASP.NET MVC	2				4		
2.5	Проверка пользовательского ввода. Валидация данных. Авторизация и аутентификация.	2				2		Опрос
	ВСЕГО	20				28	2	

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

ЛИТЕРАТУРА

Основная:

- 1st Andrew Troelsen. Pro C# 5.0 and the .NET 4.5 Framework. Sixth Edition. Apress – 2012, 1487 p.
- 2nd Albahari J., Albahari B. C# 5.0 in a Nutshell. Fifth Edition. O'Reilly Media – 2012, 1062 p.
- 3rd Jeffrey Richter. CLR via CSharp. 3rd Edition. Microsoft Press – 2010, 896 p.
- 4-й М. Мак-Дональд, А. Фримен, М. Шпушта. Microsoft ASP.NET 4 с примерами на C# 2010 для профессионалов/ Пер. с англ. – М. : ООО «И.Д. Вильямс», 2011. – 1424 с.
- 5th Dino Esposito. Programming Microsoft ASP.NET 4.: Microsoft Press, 2011. – 992 p.
- 6-й Дино Эспозито. Программирование с использованием Microsoft ASP.NET 3.5/ Пер. с англ. – М.: «Русская редакция», СПб.: Питер, 2009. – 1008 с.
- 7-й Дж. Гленн, Н. Тонн. Разработка клиентских веб-приложений на основе платформы Microsoft .NET Framework. Учебный курс Microsoft/ Пер. с англ. – М.: «Русская редакция», СПб.: Питер, 2007. – 768 с.
- 8-й Д. Рейли. Создание приложений Microsoft ASP.NET/ Пер. с англ. – М.: «Русская редакция», 2002 – 480 с.
- 9-й Дж. Шеперд. Microsoft ASP.NET 3.5: Step by Step/ Пер. с англ. – ЭКОМ Паблишерз, 2009. – 720 с.

Дополнительная:

- 1-й Эндрю Троелсен. Язык программирования C# 2010 и платформа .NET 4. 5-е издание. Apress – 2011, 1392 с.
- 2-й Нейгел, Кристиан, Ивбен, Билл, Глинн, Джей, Уотсон, Карли. C# 4.0 и платформа .NET 4 для профессионалов.: Пер. с англ. – М. : ООО «И.Д. Вильямс», 2011. – 1440 с.
- 3rd Л. Пауэре, М. Снелл. Microsoft Visual Studio 2008. СПб.: БХВ-Петербург, 2009. – 1200 с.
- 4th <http://msdn.microsoft.com/en-us/library>.
- 5th <http://msdn.microsoft.com/ru-ru/library/ms123401>.
- 6-й <http://www.rsdn.ru>.
- 7-й Алекс Макки. Введение в .NET и Visual Studio 2010 для профессионалов.: Пер. с англ. – М. : ООО «И.Д. Вильямс», 2011. – 416 с.
- 8-й Jesse Liberty, Brian MacDonald. Learning C# 3.0: Master the fundamentals of C# 3.0. O'Reilly Media. – 2008, 704 p.
- 9-й Judith Bishop. C# 3.0 Design Patterns. O'Reilly Media. – 2007, 292 p.
- 10th Jeffrey Richter. CLR via CSharp. 3rd Edition. Microsoft Press – 2010, 896 p.

СРЕДСТВА ДИАГНОСТИКИ РЕЗУЛЬТАТОВ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ

Рекомендуются следующие формы диагностики компетенций.

Устная форма

1. Опрос.

Устно-письменная форма

1. Отчеты по лабораторным работам с их устной защитой.
2. Курсовые работы с их устной защитой.
3. Зачет.

ПЕРЕЧЕНЬ ЗАДАНИЙ И КОНТРОЛЬНЫХ МЕРОПРИЯТИЙ УПРАВЛЯЕМОЙ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Контрольные мероприятия УСР по дисциплине «Язык программирования C#. Основы ASP.NET» проводятся преподавателем, как правило, во время аудиторных занятий.

Полученные студентом количественные результаты УСР учитываются как составная часть итоговой оценки по дисциплине в рамках рейтинговой системы.

ПРИМЕРНЫЙ ПЕРЕЧЕНЬ заданий для лабораторных работ

Задание 1.

Разработать тип, позволяющий вычислять корень n -ой степени из числа методом Ньютона с заданной точностью.

Предусмотреть возможность сравнения полученного результата со значением, рассчитываемым с помощью метода `Math.Pow` библиотеки классов .NET Framework.

В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 2.

Разработать тип, позволяющий конвертировать неотрицательное десятичное значение целого числа в строку, содержащую двоичное представление этого значения.

В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 3.

Разработать тип, реализующий алгоритм Евклида для вычисления НОД двух целых чисел (http://en.wikipedia.org/wiki/Euclidean_algorithm). Метод должен принимать выходной параметр, содержащий значение времени, необходимое для выполнения расчета. Добавить к разработанному типу дополнительную функциональность в виде перегруженных методов вычисления НОД для трех и т.д. целых чисел.

Добавить к разработанному типу метод, реализующий алгоритм Стейна (бинарный алгоритм Эвклида) для расчета НОД двух целых чисел (http://en.wikipedia.org/wiki/Binary_GCD_algorithm). Метод должен принимать выходной параметр, содержащий значение времени, необходимое для выполнения расчетов. Добавить к разработанному типу дополнительную функциональность в виде перегруженных методов вычисления НОД для трех и т.д. целых чисел.

Разработать метод, позволяющий сравнивать результаты работы разработанных методов на одинаковых наборах данных. (Разработать метод для отображения результатов работы с использованием гистограммы. Метод должен принимать в качестве двух обязательных параметров время выполнения, и в качестве необязательных параметров – ориентацию гистограммы (вертикальная или горизонтальная) и цвет для отображения ее полос).

Создать unit-тесты для тестирования разработанных методов.

В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 4.

Разработать класс «вектор» для работы с трехмерными векторами.

Перегрузить для класса операции, допустимые для работы с трехмерными векторами.

Создать unit-тесты для тестирования разработанных методов.

В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 5.

Разработать класс «многочлен» для работы с многочленами от одной переменной.

Перегрузить для класса операции, допустимые для работы с многочленами.

Создать unit-тесты для тестирования разработанных методов.

В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Задание 6.

Разработать тип для работы с матрицами.

Реализовать методы, позволяющие выполнять основные операции над матрицами, предусмотрев возможность их выполнения, в противном случае должно генерироваться исключение.

В качестве UI-интерфейса использовать консольное приложение с интерфейсом командной строки, WinForms или WPF-приложение.

Создать unit-тесты для тестирования методов разработанного типа.

Задание 7.

Создать сайт ASP.NET, состоящий из одной страницы. На странице проводится проверка того, что пользователь – человек. Проверка осуществляется при помощи ввода кода с картинки.

Задание 8.

Расширить предыдущее задание, добавив к сайту возможность оставлять комментарии. Голосовать могут только люди.

Задание 9.

Расширить предыдущее задание, добавив страничку регистрации новых пользователей. Информацию о пользователях и о голосовании сохранять в базе данных.

Задание 10.

Расширить предыдущее задание, используя в качестве слоя доступа к данным Entity Framework.

Задание 11.

Расширить предыдущее задание таким образом, чтобы анонимные пользователи могли просматривать голосование, но не могли голосовать; зарегистрированные пользователи могли голосовать; администраторы могли объявлять новое голосование.

ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ

Курсовая работа, как форма текущей аттестации студентов, является видом самостоятельной работы студентов, носит учебно-исследовательский характер и представляет собой решение учебной задачи по изучаемой учебной дисциплине в соответствии с установленными требованиями.

Порядок организации курсового проектирования и защиты курсовых работ определяется учреждением высшего образования.

Студент вправе выбрать тему курсовой работы из числа утвержденных на кафедре или самостоятельно предложить тему курсовой работы с обоснованием ее целесообразности. Для формирования умений и навыков работы в команде возможно выполнение группового задания, предусматривающего работу нескольких обучающихся над одной курсовой работой. В этом случае каждому из них устанавливается индивидуальный объем задач в соответствии с объемом и уровнем общих требований.

Цель курсовых работ

по дисциплине *«Язык программирования C#. Основы ASP.NET»*:

- а) закрепить, углубить и расширить полученные по дисциплине знания с учетом современных тенденций в разработке программного обеспечения;
- б) овладеть навыками самостоятельной работы с различными информационными источниками (литература, интернет);
- в) выработать умение публичной защиты.

На выполнение курсовой работы *«Язык программирования C#. Основы ASP.NET»* отводится 40 часов внеаудиторной управляемой самостоятельной работы студента на III курсе в 5–6 семестрах или на IV курсе в 7–8 семестрах.

Примерная тематика курсовых работ

1. Используя технологию WPF разработать Windows-приложения

- приложение для просмотра фотографий;
- медиа плеер;
- игру.

При реализации проекта использовать

- многоуровневую архитектуру (Business Logic Layer, Data Access Layer, Presentation Layer);
- ADO.NET (подключенный уровень) или Entity Framework для работы с данным;
- пользовательские элементы управления.

2. Используя технологию ASP.NET разработать web-приложения

- Простая социальная сеть. Регистрация пользователей. Настройка персональной информации. «Друзья» пользователя. Обмен сообщениями с другими пользователями. Поиск пользователей по определенным критериям. Управление пользователями. Модерирование сообщений.
- Интернет-фотоальбом. Регистрация, загрузка фотографий. Возможность просматривать и оценивать фотографии других пользователей. Поиск фотографий. Управление пользователями.
- Персональный блог/блогхост. Регистрация пользователей. Создание блога (блогов). Создание и редактирование статей блога. Тэги статей. Поиск по тэгам, тексту. Комментирование статей. Управление пользователями. Модерирование статей и комментариев.
- Интернет-аукцион. Просмотр, поиск, и «покупка» лотов. Регистрация и выставление лотов на аукцион. Управление пользователями и моделирование списка лотов.
- Система тестирования знаний. Регистрация, выбор (поиск) теста. Прохождение теста с контролем времени. Статистика тестирования. Управление пользователями. Редактирование тестов. Расширенная статистика тестирования.
- Файловое хранилище. Доступ к файлу по короткой ссылке. Общие файлы и файлы с ограниченным доступом. Для зарегистрированных пользователей – возможность размещения файлов и управление файлами. Управление пользователями и их файлами. Поиск файлов.
- Картотека текстовых материалов (вариант: аудио или видео). Поиск в картотеке по различным критериями. Оценка материалов пользователями.
- Форум. Стандартные операции, присущие любому форуму – добавление тем, сообщений. Модерирование записей. Работа с пользователями форума.

При реализации проекта использовать:

- многоуровневую архитектуру (Business Logic Layer, Data Access Layer, Presentation Layer);
- ADO.NET (подключенный уровень) или Entity Framework для работы с данным;
- пользовательские элементы управления и мастер страницы, темы и скины;
- валидацию данных на стороне клиента и на стороне сервера.

4.

**ПРОТОКОЛ
СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ ПО ИЗУЧАЕМОЙ
УЧЕБНОЙ ДИСЦИПЛИНЕ С ДРУГИМИ ДИСЦИПЛИНАМИ СПЕ-
ЦИАЛЬНОСТИ**

Название дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Web-программирование	Веб-технологий и компьютерного моделирования	Нет	Вносить изменения не требуется (протокол №10 от 14.05.2015г.)
Методы программирования и информатика	Веб-технологий и компьютерного моделирования	Нет	Вносить изменения не требуется (протокол №10 от 14.05.2015г.)
Информационные технологии	Веб-технологий и компьютерного моделирования	Нет	Вносить изменения не требуется (протокол №10 от 14.05.2015г.)
Базы данных	Веб-технологий и компьютерного моделирования	Нет	Вносить изменения не требуется (протокол №10 от 14.05.2015г.)

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ
на 2016/2017 учебный год

№ пп	Дополнения и изменения	Основание
	В пояснительную записку внести требования к компетенциям специалиста: академические – АК-1, АК-2, АК-3, АК-4, АК-5, АК-6, АК-7, АК-8, АК-9; социально-личностные – СЛК-1, СЛК-2, СЛК-3, СЛК-4, СЛК-5, СЛК-6; профессиональные – ПК-1, ПК-2, ПК-3, ПК-4, ПК-5, ПК-6, ПК-7, ПК-8, ПК-9, ПК-10, ПК-11, ПК-12, ПК-13, ПК-14, ПК-15, ПК-16, ПК-17, ПК-18, ПК-19, ПК-20, ПК-21, ПК-22, ПК-23, ПК-24, ПК-25, ПК-26, ПК-27.	

Учебная программа пересмотрена и одобрена на заседании кафедры веб-технологий и компьютерного моделирования (протокол № 9 от 27.05.2016г.)

Заведующий кафедрой

канд. физ.-мат. наук, доцент
(ученая степень, ученое звание)

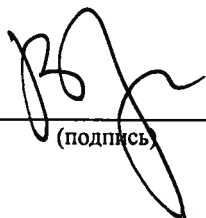

(подпись)

В.С. Романчик
(И.О.Фамилия)

УТВЕРЖДАЮ

 Декан факультета

канд. физ.-мат. наук, доцент
(ученая степень, ученое звание)


(подпись)

Д.Г. Медведев
(И.О.Фамилия)