Лебедева Екатерина Александровна
Подписано цифровой подписью: Лебедева Екатерина Александровна
Дата: 2026.06.23 11:49:02 +03'00'

Приложение 4

к образовательной программе

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.01.04 ПРОЕКТИРОВАНИЕ WEB-СЕРВИСОВ ИНФОРМАЦИОННЫХ СИСТЕМ

(индекс, наименование дисциплины в соответствии с учебным планом)

09.04.03 Прикладная информатика

(код, наименование направления подготовки/специальности)

Информационные технологии в государственном и муниципальном управлении

(наименование образовательной программы)

очная, заочная

(форма обучения)

Год набора – 2026

Нижний Новгород, 2026 г.

Автор(ы)-составитель(и) РПД:

Гребенюк Иван Иванович, канд.тех.наук., доцент, заведующий кафедрой информатики и информационных технологий Нижегородского института управления – филиала РАНХиГС

Орлов Михаил Львович, доцент кафедры информатики и информационных технологий, канд.физ.-мат. наук, доцент

Заведующий кафедрой:

Гребенюк Иван Иванович, канд.тех.наук., доцент, заведующий кафедрой информатики и информационных технологий Нижегородского института управления – филиала РАНХиГС

РПД Б1.В.01.04 «Проектирование web-сервисов информационных систем» одобрена на заседании кафедры информатики и информационных технологий.

Протокол №16 от 11марта 2026 г.

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

2. Объем и место дисциплины в структуре образовательной программы

3. Содержание и структура дисциплины

4. Типы оценочных материалов, показатели и критерии их оценивания

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

7. Методические материалы по освоению дисциплины

8. Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет»

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Дисциплина Б1.В.01.04 «Проектирование web-сервисов информационных систем» обеспечивает формирование у обучающихся следующих универсальных, общепрофессиональных и профессиональных компетенций*:

ОТФ/ТФ и реквизиты ПС	**Код компетенции**	**Наименование Компетенции**	**Код индикатора достижения компетенций**	**Наименование индикатора достижения компетенций**	**Образовательный результат**
ОТФ: управление работами по сопровождению и проектами создания (модификации) ИС, автоматизирующих задачи организационного управления и бизнес-процессы (D); ТФ: • разработка инструментов и методов проектирования бизнес-процессов заказчика в рамках управления работами по сопровождению и проектами	ПК-6	Способен выбирать методологию и технологию проектирования ИС с учетом проектных рисков	ИПК-6.1	*Использует современные методологии проектирования информационных систем.*	*ИПК-6.1 З-1* ***Знает*** *теоретические основы проектирования крупного масштаба и сложных систем;* *ИПК-6.1 З-2* ***Знает*** *методы концептуального, функционального и логического проектирования систем;* *ИПК-6.1 З-3* ***Знает*** *способы масштабирования информационных систем для учёта их при логическом проектировании.* *ИПК-6.1 У-1* ***Умеет*** *анализировать и оптимизировать*

ОТФ/ТФ и реквизиты ПС	Код компетенции	Наименование Компетенции	Код индикатора достижения компетенций	Наименование индикатора достижения компетенций	Образовательный результат
создания (модификации) ИС (D/08.7); • разработка инструментов и методов адаптации бизнес-процессов заказчика к возможностям ИС в рамках управления работами по сопровождению и проектами создания (модификации) ИС (D/09.7).					*известные подходы к реализации функционала ИС (подсистем, компонентов);* *ИПК-6.1 У-2* ***Умеет*** *разрабатывать архитектуру, структуру, организацию ИС (подсистем, компонентов);* *ИПК-6.1 У-3* ***Умеет*** *моделировать компоненты и бизнес-процессы ИС в нотациях UML, EPC, BPMN, ARIS, SADT*
ОТФ: управление работами по сопровождению и проектами создания (модификации) ИС, автоматизирующих задачи организационного управления	ПК-11	Способен проектировать архитектуру и сервисы ИС предприятий и организаций в прикладной области	ИПК-11.1	*Применяет способы интеграции сервисов и баз данных информационных систем.*	*ИПК-11.1 З-1* ***Знает*** *возможности ИС* *ИПК-11.1 З-2* ***Знает*** *архитектуру, устройство и функционирование вычислительных систем*

ОТФ/ТФ и реквизиты ПС	Код компетенции	Наименование Компетенции	Код индикатора достижения компетенций	Наименование индикатора достижения компетенций	Образовательный результат
и бизнес-процессы (D); ТФ: • разработка инструментов и методов адаптации бизнес-процессов заказчика к возможностям ИС в рамках управления работами по сопровождению и проектами создания (модификации) ИС (D/09.7); • организационное и технологическое обеспечение разработки баз данных ИС в рамках управления работами по сопровождению и проектами создания (модификации) ИС (D/17.7).					*ИПК-11.1 З-3* ***Знает*** *программные средства и платформы инфраструктуры информационных технологий организаций* *ИПК-11.1 З-4* ***Знает*** *инструменты и методы моделирования бизнес-процессов организации* *ИПК-11.1 У-1* ***Умеет*** *проектировать и реализовывать интеграционные решения.* *ИПК-11.1 У-2* ***Умеет*** *настраивать и оптимизировать процессы управления данными в рамках информационных систем.*

ОТФ/ТФ и реквизиты ПС	**Код компетенции**	**Наименование Компетенции**	**Код индикатора достижения компетенций**	**Наименование индикатора достижения компетенций**	**Образовательный результат**
ОТФ: Управление работами по сопровождению и проектами создания (модификации) ИС, автоматизирующих задачи организационного управления и бизнес-процессы (D); ТФ: • организационное и технологическое обеспечение разработки баз данных ИС (D/17.7).	ПК-13	Способен принимать эффективные проектные решения в условиях неопределенности и риска	ИПК-13.2	*Определяет показатели эффективности и проектных решений.*	*ИПК-13.2 З-1* ***Знает*** *модели управления эффективностью ИС.* *ИПК-13.2 З-2* ***Знает*** *общие положения по созданию автоматизированных информационных систем, сущность технического задания как основного документа в процессе создания ИС.* *ИПК-13.2 У-1* ***Умеет*** *формировать систему экономических показателей эффективности ИС. Учитывать цели и ожидания заказчика, ключевые параметры, отражающие результативность деятельнос*

ОТФ/ТФ и реквизиты ПС	Код компетенции	Наименование Компетенции	Код индикатора достижения компетенций	Наименование индикатора достижения компетенций	Образовательный результат
					ти. *ИПК-13.2 У-2* ***Умеет*** *строить карты причинно-следственных связей влияния внедрения ИС на производство.*
ОТФ: управление работами по сопровождению и проектами создания (модификации) ИС, автоматизирующих задачи организационного управления и бизнес-процессы (D); ТФ: • организационное и технологическое обеспечение разработки баз данных ИС в рамках управления работами по	ПК-17	Способен организовывать и проводить переговоры с представителями заказчика и профессиональные консультации на предприятиях и в организациях	ИПК-17.2	*Создает проекты по разработке и внедрению информационных систем и технологий.*	*ИПК-17.2 З-1* ***Знает*** *методы управления содержанием проекта: документирование требований, анализ продукта, моделируемые совещания* *ИПК-17.2 У-1* ***Умеет*** *создавать проекты по разработке и внедрению информационных систем и технологий.*

ОТФ/ТФ и реквизиты ПС	Код компетенции	Наименование Компетенции	Код индикатора достижения компетенций	Наименование индикатора достижения компетенций	Образовательный результат
сопровождению и проектами создания (модификации) ИС (D/17.7).					

2. Объем и место дисциплины в структуре образовательной программы

Дисциплина Б1.В.01.04. «Проектирование web-сервисов информационных систем» является дисциплиной части, формируемой участниками образовательных отношений.

Общая трудоемкость дисциплины составляет 3 зачетных единицы, 108 академических часов / 81 астрономический час.

Освоение дисциплины опирается на необходимый объем теоретических знаний и практических навыков в области проектирования информационных систем и программирования.

Дисциплины, на которых базируется данная дисциплина: Технологии межведомственного электронного взаимодействия, шифр дисциплины Б1.В.01.05; Тенденции развития Web-пространства, шифр дисциплины Б1.О.07; Управление внедрением IT-проектов, шифр дисциплины Б1.О.06.

Дисциплины, для которых данная дисциплина является предшествующей: Научно-исследовательская работа, шифр Б2.В.01(Н), Преддипломная практика, шифр Б2.В.03(Пд).

При очной форме обучения на контактную работу с преподавателем выделено 44 ч, в том числе на лекционные занятия – 8 ч, на практические (семинарские) занятия – 32 ч, контактная работа на аттестацию в период экзаменационных сессий – 4 ч.; на самостоятельную работу обучающихся выделено 64 ч.

При заочной форме обучения на контактную работу с преподавателем выделено 12 ч, в том числе на лекционные занятия – 4 ч, на практические (семинарские) занятия – 8 ч; на самостоятельную работу обучающихся выделено 92 ч, контроль – 4 ч.

Форма промежуточной аттестации в соответствии с учебным планом – зачет.

3. Содержание и структура дисциплины

3.1. Структура дисциплины

Очная/заочная форма обучения

№ п/п	Наименование тем и (или) разделов	ВСЕГО	Объем дисциплины, ак.час											Форма текущего контроля успеваемости, промежуточной аттестации
			Контактная работа обучающихся с преподавателем по видам учебных занятий									Самостоятельная работа		
			Период теоретического обучения						Период промежуточной аттестации (сессия)					
			Занятия лекционного типа		Занятия семинарского типа		ИК	КСР	КЭ	Каттэк	Контроль	СРкр	СРэк	СР
			Л	ВЛ	ЛР	ПЗ								
Очная форма обучения														
Тема 1	Введение в веб-сервисы	20	2			6								12	Опрос, контрольные задания с развернутым ответом
Тема 2	Основы XML: семантика и синтаксис	20	2			6								12	Опрос, контрольные задания с развернутым ответом
Тема 3	Язык описания веб-сервисов	20	2			6								12	Опрос, контрольные задания с развернутым ответом
Тема 4	Универсальный интерфейс распознавания, описания и интеграции	20	2			6								12	Опрос, контрольные задания с развернутым ответом

№ п/п	Наименование тем и (или) разделов	ВСЕГО	Объем дисциплины, ак.час											Форма текущего контроля успеваемости, промежуточной аттестации
			Контактная работа обучающихся с преподавателем по видам учебных занятий									Самостоятельная работа		
			Период теоретического обучения						Период промежуточной аттестации (сессия)					
			Занятия лекционного типа		Занятия семинарского типа		ИК	КСР	КЭ	Каттэк	Контроль	СРкр	СРэк	СР	
			Л	ВЛ	ЛР	ПЗ									
Тема 5	Реализация веб-сервисов на различных платформах	24				8								16	Опрос, контрольные задания с развернутым ответом
Промежуточная аттестация		4								4					Зачет
Итого		**108**	**8**			**32**				**4**				**64**	
Заочная форма обучения															
Тема 1	Введение в веб-сервисы	20	1			1								18	Опрос, контрольные задания с развернутым ответом
Тема 2	Основы XML: семантика и синтаксис	20	1			1								18	Опрос, контрольные задания с развернутым ответом
Тема 3	Язык описания веб-сервисов	20	1			2								17	Опрос, контрольные задания с развернутым ответом
Тема 4	Универсальный интерфейс	22	1			2								19	Опрос, контрольные задания с развернутым

№ п/п	Наименование тем и (или) разделов	ВСЕГО	Объем дисциплины, ак.час											Форма текущего контроля успеваемости, промежуточной аттестации
			Контактная работа обучающихся с преподавателем по видам учебных занятий									Самостоятельная работа		
			Период теоретического обучения						Период промежуточной аттестации (сессия)					
			Занятия лекционного типа		Занятия семинарского типа		ИК	КСР	КЭ	Каттэк	Контроль	СРкр	СРэк	СР	
			Л	ВЛ	ЛР	ПЗ									
	распознавания, описания и интеграции														ответом
Тема 5	Реализация веб-сервисов на различных платформах	22				2								20	Опрос, контрольные задания с развернутым ответом
Промежуточная аттестация		4								4					Зачет
Итого		**108**	**4**			**8**				**4**				**92**	

Используемые сокращения:

Л – лекции - занятия, предусматривающие преимущественную передачу учебной информации обучающимся педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях,).

ВЛ – видео лекции.

ЛР – лабораторные работы.

ПЗ – практические занятия (за исключением лабораторных работ).

ИК – индивидуальные консультации.

КСР – контроль самостоятельной работы

КЭ – консультации перед экзаменом

Каттэк – контактная работа на аттестацию в период экзаменационных сессий

Контроль - контактная работа на аттестацию в период экзаменационных сессий для заочной формы обучения

СРкр – самостоятельная работа на подготовку курсовой работы/ курсового проекта.

СРэк – самостоятельная работа на подготовку к экзамену.
СР – самостоятельная работа в семестре на подготовку к учебным занятиям.

3.2. Содержание дисциплины

Тема 1 Введение в веб-сервисы. ИПК-17.2

Архитектура электронного правительства. Реализация «Стратегии развития информационного общества».

Предоставление государственных и муниципальных услуг органами власти средствами веб-сервисов и мобильных приложений. Веб-сервисы как способы вывода информации и данных.

Тема 2 Основы XML: семантика и синтаксис. ИПК-6.1, ИПК-11.1, ИПК-17.2

Язык XML как универсальное средство обеспечения интеграции ИС. Связь XML с другими языками текстовой разметки и его особенности. Создание XML-документов. XML теги и XML элементы.

Тема 3 Язык описания веб-сервисов. ИПК-6.1, ИПК-11.1, ИПК-17.2

Сервисно-ориентированная архитектура приложений. Протоколы и стандарты веб-сервисов: SOAP, WSDL, UDDI. Среда разработки Visual Studio.net, «АСМО-графический редактор».

Тема 4 Универсальный интерфейс распознавания, описания и интеграции. ИПК-6.1, ИПК-11.1, ИПК-13.2, ИПК-17.2

Интеграция данных и массивов элементов в информационных системах. Автоматизация бизнес-схем и бизнес-процессов в различных информационных системах. Проектирование веб-сервиса для связи различных ИС

Тема 5 Реализация веб-сервисов на различных платформах. ИПК-6.1, ИПК-11.1, ИПК-13.2, ИПК-17.2

Анализ порталов предоставления государственных и муниципальных услуг органов власти различных уровней. Проектирование веб-сервиса и разработка мобильного приложения для предоставления государственной (муниципальной) услуги.

4. Типы оценочных материалов, показатели и критерии оценивания

4.1. Оценочные материалы по дисциплине Б1.В.01.04 «Проектирование web-сервисов информационных систем» входят в состав оценочных материалов по образовательной программе. Совокупность оценочных материалов по всем дисциплинам (модулям) образовательной программы составляет фонд оценочных средств (далее – ФОС). ФОС используется при проведении текущего контроля успеваемости и промежуточной аттестации обучающихся с целью оценивания достижения обучающимися планируемых результатов обучения.

4.2. ФОС разработан как комплекс проверочных заданий различного типа и уровня сложности, включает критерии и шкалы оценивания, а также «ключи» правильных ответов. ФОС формируется как отдельный документ и хранится в электронном виде, доступ к ФОС предоставлен ограниченному кругу лиц.

4.3. Для самостоятельной работы обучающихся при подготовке к текущему контролю успеваемости и промежуточной аттестации в рабочих программах дисциплин размещены типовые проверочные задания, которые можно условно разделить на задания закрытого, комбинированного и открытого типов.

Задания закрытого типа — это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных.

Задания комбинированного типа – это тестовые задания, в которых каждый вопрос сопровождается готовыми вариантами ответов, из которых необходимо выбрать один или несколько правильных и обосновать свой выбор.

Задания открытого типа — это задания, в которых на каждый вопрос должен быть предложен развернутый обоснованный ответ.

В зависимости от типа задания рекомендованы определенная последовательность выполнения и система оценивания выполнения заданий.

4.4. Типы заданий, сценарии выполнения, критерии оценивания

ТИП ЗАДАНИЯ	ИНСТРУКЦИЯ	СЦЕНАРИИ ВЫПОЛНЕНИЯ	КРИТЕРИИ ОЦЕНИВАНИЯ
Задание закрытого типа с выбором одного правильного ответа из нескольких вариантов предложенных	Прочитайте текст, выберите правильный ответ	1.Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные вариант-ты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа (например, 3 или В).	Ответ считается верным, если правильно указана цифра или буква
Задание закрытого типа на установление соответствия	Прочитайте текст и установите соответствие	1.Внимательно прочитать текст задания и понять, что в качестве ответа ожидаются пары элементов. 2. Внимательно прочитать оба списка: список 1 – вопросы, утверждения, факты, понятия и т.д.; список 2 – утверждения, свойства объектов и т.д. 3. Сопоставить элементы списка 1 с элементами списка 2, сформировать пары элементов. 4. Записать попарно буквы и цифры (в зависимости от задания) вариантов ответа (например, А1 или Б4).	Ответ считается верным, если правильно указаны цифры или буквы
Задание закрытого типа с выбором нескольких правильных ответов из нескольких	Прочитайте текст, выберите правильные ответы	1.Внимательно прочитать текст задания и понять, что в качестве ответа ожидается несколько правильных ответов из предложенных вариантов. 2. Внимательно прочитать предложенные	Ответ считается верным, если правильно установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями

ТИП ЗАДАНИЯ	ИНСТРУКЦИЯ	СЦЕНАРИИ ВЫПОЛНЕНИЯ	КРИТЕРИИ ОЦЕНИВАНИЯ
вариантов предложенных		вариант-ты ответа. 3. Выбрать несколько правильных ответов. 4. Записать только номера (или буквы) выбранного варианта ответа (например, 1 4 или А Г).	другого)
Задание закрытого типа на установление последовательности	Прочитайте текст и установите последовательность	1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается последовательность элементов. 2. Внимательно прочитать предложенные варианты ответа. 3. Построить верную последовательность из предложенных элементов. 4. Записать буквы/цифры (в зависимости от задания) вариантов ответа в нужной последовательности (например, БВА или 135).	Ответ считается верным, если правильно указана вся последовательность цифр
Задание комбинированного типа с выбором одного правильного ответа из предложенных и обоснованием выбора	Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа	1.Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов. 2. Внимательно прочитать предложенные варианты ответа. 3. Выбрать один верный ответ. 4. Записать только номер (или букву) выбранного варианта ответа. 5. Записать аргументы, обосновывающие выбор	Ответ считается верным, если правильно указана цифра или буква и приведены корректные аргументы, используемые при выборе ответа

ТИП ЗАДАНИЯ	ИНСТРУКЦИЯ	СЦЕНАРИИ ВЫПОЛНЕНИЯ	КРИТЕРИИ ОЦЕНИВАНИЯ
		ответа (например, 4 текст обоснования).	
Задание открытого типа с развернутым ответом	Прочитайте текст и запишите развернутый обоснованный ответ	1. Внимательно прочитать текст задания и понять суть вопроса. 2.Продумать логику и полноту ответа. 3.Записать ответ, используя четкие компактные формулировки. 4.В случае расчетной задачи, записать решение и ответ	Ответ считается верным: 1. Отсутствие фактических ошибок. 2. Раскрытие объема используемых понятий (полнота ответа). 3.Обоснованность ответа (наличие аргументов). 4. Логическая последовательность излагаемого материала.

4.5. Общая шкала оценивания результатов текущего контроля успеваемости и промежуточной аттестации обучающихся с применением БРС

Итоговая балльная оценка	**Традиционная система**	**Бинарная система**	**ECTS**	
			Для традиционной системы	**Для бинарной системы**
95-100	Отлично	Зачтено	A	P/ Passed
85-94			B	P/ Passed
75-84	Хорошо		C	P/ Passed
65-74			D	P/ Passed
55-64	Удовлетворительно		E	P/ Passed
0-54	Неудовлетворительно	Не зачтено	F	F/Failed

Соотношение баллов за текущий контроль успеваемости и промежуточную аттестацию, а также повторную промежуточную аттестацию:

Максимальная сумма баллов за текущий контроль успеваемости	Максимальная сумма баллов за промежуточную аттестацию	Максимальная итоговая балльная оценка	Максимальная сумма баллов за повторную промежуточную аттестацию
60 баллов	40 баллов	100 баллов	100 баллов

5. Формы аттестации, типовые оценочные материалы для текущего контроля успеваемости обучающихся, критерии и шкалы оценивания по контрольным точкам

5.1. В ходе реализации дисциплины Б1.В.01.04 «Проектирование web-сервисов информационных систем» используются следующие формы текущего контроля успеваемости обучающихся (в том числе, задания к контрольным точкам): *опрос, контрольное задание открытого типа с развернутым ответом*.

5.2. Типовые оценочные материалы для текущего контроля успеваемости обучающихся (вне контрольных точек):

Тема 1 Введение в веб-сервисы. ИПК-17.2

Вопросы опроса

№	Вопрос	Ответ
1	Что такое web-сервисы? Приведите примеры.	
2	Назовите основные показатели реализации «Стратегии развития информационного общества» в Нижегородском регионе.	
3	Какие web-сервисы доступны на портале государственных и муниципальных услуг Нижегородской области?	

<u>Контрольные задания с развернутым ответом</u>

Практическая работа №1. Введение в веб-сервисы

Учебные вопросы:

1. Архитектура электронного правительства.
2. Реализация «Стратегии развития информационного общества».
3. Предоставление государственных и муниципальных услуг органами власти средствами веб-сервисов и мобильных приложений.

Задания для самостоятельной выполнения и подготовки к занятиям:

1. Изучить функционал интернет-порталов органов государственной и муниципальной власти. Найти основные услуги, предоставляемые в электронном виде через порталы.
2. Описать структуру портала предоставления государственных услуг www.gosuslugi.ru. Найти и описать функционал веб-сервисов портала.
3. Найти информацию об использовании мобильных приложений, как средства интеграции с базами данных информационных систем органов власти.
4. Создать сравнительную таблицу функционала порталов предоставления услуг с использованием веб-сервисов нескольких муниципальных образований (реестр услуг, административный регламент, используемая платформа и т.д.).

Тема 2 Основы XML: семантика и синтаксис. ИПК-6.1, ИПК-11.1, ИПК-17.2

Вопросы опроса

№	Вопрос	Ответ

№	Вопрос	Ответ
1	Что такое XML?	
2	Для чего используется XML?	
3	Какие основные синтаксические правила XML?	
4	Что такое корневой элемент в XML?	
5	Как записывается декларация XML?	

<u>Контрольные задания с развернутым ответом</u>

Практическая работа №2 Язык XML

Учебные вопросы:

1. Язык XML как универсальное средство обеспечения интеграции ИС.
2. Связь XML с другими языками текстовой разметки и его особенности.
3. Создание XML-документов.
4. XML теги и XML элементы.

Задания для самостоятельной выполнения и подготовки к занятиям:

1. Сделайте обзор и анализ наиболее популярных инструментальных средств построения веб-сервисов информационных систем.
2. Предложите возможные проблемные области для создания веб-сервиса.
3. Проанализируйте с точки зрения эффективности работы любой информационный ресурс, содержащий элементы веб-сервисов. Предложите систему критериев оценки эффективности работы ресурса.
4. Предложите свою модель этого информационного ресурса на основе современных информационных технологий и инструментальных сред.
5. Разработайте структуру информационного ресурса и веб-сервиса.

Тема 3 Язык описания веб-сервисов. ИПК-6.1, ИПК-11.1, ИПК-17.2

Вопросы опроса

№	Вопрос	Ответ
1	Что такое WSDL (Web Services Description Language)?	
2	Какие основные элементы	

№	Вопрос	Ответ
	включает документ WSDL?	
3	Что такое SOAP (Simple Object Access Protocol)?	
4	Что такое UDDI (Universal Description, Discovery and Integration)?	
5	В чём заключается роль XML в контексте веб-сервисов?	
6	Какие преимущества предоставляет WSDL?	
7	Чем отличается абстрактная WSDL от конкретной (concrete) WSDL?	

Контрольные задания с развернутым ответом

Практическая работа №3 Статистическая и динамическая модели XML документ.

Учебные вопросы:

1. Изучить конструкции языка XML, способы описания структуры документа средствами DTD, описания схемы документа на языке XSD. Изучить способы обработки документов XML.
2. Написать по созданной в практической работе № 2 статистической и динамической модели XML документ.
3. Описать структуру документа средствами DTD.
4. Описать схему документа на языке XSD
5. Преобразовать документ средствами XSLT
6. Преобразовать документ с помощью DOM
7. Оформить отчет, включающий постановку задачи, тексты XML, DTD, XSD, XSLT документов, демонстрацию работы программ.
8. Защитить лабораторную работ

Практическая работа №4 Среда разработки Visual Studio.net

Учебные вопросы:

1. Протоколы и стандарты веб-сервисов: SOAP, WSDL, UDDI.
2. Среда разработки Visual Studio.net.

Задания для самостоятельной выполнения и подготовки к занятиям:

Изучить среду разработки Visual Studio.net.

Практическая работа №5 Среда разработки «АСМО-графический редактор»

Учебные вопросы:

1. Протоколы и стандарты веб-сервисов: SOAP, WSDL, UDDI.
2. Среда разработки «АСМО-графический редактор».

Задания для самостоятельной выполнения и подготовки к занятиям:

Изучить среду разработки «АСМО-графический редактор».

Тема 4 Универсальный интерфейс распознавания, описания и интеграции. ИПК-6.1, ИПК-11.1, ИПК-13.2, ИПК-17.2

Вопросы опроса

№	Вопрос	Ответ
1	Что такое универсальный интерфейс распознавания, описания и интеграции (UDDI)? В чём заключается его основная функция?	
2	Какие стандарты и протоколы используются в контексте веб-сервисов для обеспечения интеграции? (Например, SOAP, WSDL, XML)	
3	Как работает WSDL (Web Service Description Language) в контексте описания веб-сервисов? Какие данные включает WSDL-документ и для чего он используется?	
4	Что такое SOAP (Simple Object Access Protocol) и на базе какого стандарта он построен? Какова основная идея этого протокола?	

<u>Контрольные задания с развернутым ответом</u>

Практическая работа №6 Создание Web-сервисов

Содержание работы

1. Изучить основные принципы создания Web-сервисов.
2. Создать web-сервис, который будет использовать XML документ в качестве источника данных. То есть создать Web- сервис, возвращающий обычный набор данных содержащий информацию, взятую из XML-файла.
3. Дополнить созданный web- сервис процедурой, позволяющей сохранять данные в XML файле.
4. Создать клиентское приложение.
5. Оформить отчет, включающий постановку задачи, текст программ, демонстрацию работы программ.
6. Защитить лабораторную работу

Контрольная работа

Вариант 1

1. Дано *DTD*:

```
<!DOCTYPE base_of_students
[
<!ELEMENT base_of_students (university+)>
<!ELEMENT university (student+)>
<!ATTLIST university nameU ID #REQUIRED>
<!ELEMENT student (name,course,group,mark,form,faculty,adress)>
<!ELEMENT name (surname,first,second)>
<!ELEMENT surname (#PCDATA)>
<!ELEMENT first (#PCDATA)>
<!ELEMENT second (#PCDATA)>
<!ELEMENT course (#PCDATA)>
<!ELEMENT group (#PCDATA)>
<!ELEMENT mark (#PCDATA)>
<!ELEMENT form (#PCDATA)>
<!ELEMENT fuculty (#PCDATA)>
<!ELEMENT adress (#PCDATA)>
]
>
```

Написать соответствующий данному *DTD* XML-*файл*.

2. Дан XML-*файл*:

```
<?xml version="1.0" encoding="windows-1251" ?>
<?xml-stylesheet type='text/xsl' href=7apos;1.xsl'?>
<documents>
  <book>
        <title>Языки информационного обмена </title>
        <type> Лекции </type>
```

```
            <file>Лекции ЯИО.doc</file>
        </book>
        <book>
            <title>Языки информационного обмена </title>
            <type>Лабораторные работы</type>
            <file>Лабораторные работы ЯИО.doc</file>
        </book>
        <book>
            <title>Администрирование ИС</title>
            <type>Лекции</type>
            <file>Лекции АИС.doc</file>
        </book>
        <book>
            <title>Администрирование ИС</title>
            <type>Лабораторные работы</type>
            <file>Лабораторные работы АИС.doc</file>
        </book>
    </documents>
```

Написать *файл* 1.xsl для отображения содержимого XML-файла. Содержимое тега <file> должно отображаться как *ссылка*.

Вариант 2

1. Дана часть схемы документа:

```
<xs:element name="adress" minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
<xs:element name="city" type="xs:string" minOccurs="0" />
<xs:element name="street" type="xs:string" minOccurs="0" />
<xs:element name="house" type="xs:string" minOccurs="0" />
<xs:element name="zip" type="xs:string" minOccurs="0" />
<xs:element name="phone-list" minOccurs="1" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
<xs:element name="phone" nillable="true" minOccurs="1" maxOccurs="unbounded">
<xs:complexType>
<xs:simpleContent msdata:ColumnName="phone_Text" msdata:Ordinal="0">
<xs:extension base="xs:string"></xs:extension>
</xs:simpleContent>
</xs:complexType>
</xs:element>
</xs:sequence>
```

```
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
```

Написать фрагмент XML-файла удовлетворяющий данному фрагменту.

2. Дан XML-*файл*:

```
<?xml version="1.0" encoding="windows-1251" ?>
<?xml-stylesheet type='text/xsl' href='1.xsl'?>
<documents>
  <book>
        <title>Языки информационного обмена </title>
        <type> Лекции </type>
        <file>
<id>1</id>
<name>Лекции ЯИО.doc</name>
</file>
  </book>
  <book>
        <title>Языки информационного обмена </title>
        <type>Лабораторные работы</type>
        <file>
<id>3</id>
<name> Лабораторные работы ЯИО.doc </name>
</file>
  </book>
  <book>
        <title>Администрирование ИС</title>
        <type>Лекции</type>
        <file>
<id>2</id>
<name>Лекции АИС.doc</name>
</file>
  </book>
  <book>
        <title>Администрирование ИС</title>
        <type>Лабораторные работы</type>
        <file>
<id>4</id>
<name> Лабораторные работы АИС.doc </name>
</file>

  </book>
```

```
</documents>
```

Написать *файл* 1.xsl для отображения содержимого XML-файла. Содержимое тега <file> должно только в случае если id<3.

Вариант 3

1. Дано *DTD*:

```
<!DOCTYPE notebook
[
<!ELEMENT notebook (person+)>
<!ELEMENT person (name, birthday?, address*, phone-list)>
<!ELEMENT name EMPTY>
<!ATTLIST name
first CDATA #IMPLIED
second CDATA #IMPLIED
surname CDATA #REQUIRED>
<!ELEMENT birthday (#PCDATA)>
<!ELEMENT address (street,city,zip)?>
<!ELEMENT street (#PCDATA)>
<!ELEMENT city (#PCDATA)>
<!ATTLIST city
type (город | поселок | деревня) "город">
<!ELEMENT zip (#PCDATA)>
<!ELEMENT phone-list (work-phone*, home-phone*)>
<!ELEMENT work-phone (#PCDATA)>
<!ELEMENT home-phone (#PCDATA)>
]
>
```

Написать соответствующий данному *DTD* XML-*файл*.

2. Дан XML-*файл*:

```
<?xml version="1.0" encoding="windows-1251" ?>
<?xml-stylesheet type='text/xsl' href='1.xsl'?>
<documents>
  <book>
            <title>Языки информационного обмена </title>
            <type> Лекции </type>
            <file>
<id>1</id>
<name1>Лекции ЯИО.doc</name1>
<name2>Лекции ЯИО.html</name2>
</file>
  </book>
  <book>
            <title>Языки информационного обмена </title>
```

```
            <type>Лабораторные работы</type>
            <file>
        <id>3</id>
        <name1> Лабораторные работы ЯИО.doc </name1>
        <name2> Лабораторные работы ЯИО.html </name2>
        </file>
          </book>
          <book>
            <title>Администрирование ИС</title>
            <type>Лекции</type>
            <file>
        <id>2</id>
        <name1>Лекции АИС.doc</name1>
        <name2>Лекции АИС.html</name2>

        </file>
          </book>
          <book>
            <title>Администрирование ИС</title>
            <type>Лабораторные работы</type>
            <file>
        <id>4</id>
        <name1> Лабораторные работы АИС.doc </name1>
        <name2> Лабораторные работы АИС.html </name2>
        </file>

          </book>
        </documents>
```

Написать *файл* 1.xsl для отображения содержимого XML-файла. Содержимое тега <file> отображается следующим образом: если id>3 отображается только содержимое тега <name1>, если id=3 отображается только содержимое тега <name2>, если id>3 отображается содержимое тегов <name1> и <name2>.

Вариант 4

1. Дана часть схемы документа:

```
<xs:element name="good" minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
<xs:element name="description" minOccurs="0" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
```

```
<xs:element name="model" type="xs:string" minOccurs="0" />
<xs:element name="parametrs" type="xs:string" minOccurs="0" />
<xs:element name="price" type="xs:string" minOccurs="0" />
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="name" form="unqualified" type="xs:string" />
</xs:complexType>
</xs:element>
```

Написать фрагмент XML-файла удовлетворяющий данному фрагменту.

2. Дан XML-*файл*:

```
<?xml version="1.0" encoding="windows-1251" ?>
<documents>
  <book>
        <title>Языки информационного обмена </title>
        <type> Лекции </type>
        <file>Лекции ЯИО.doc</file>
  </book>
  <book>
        <title>Языки информационного обмена </title>
        <type>Лабораторные работы</type>
        <file>Лабораторные работы ЯИО.doc</file>
  </book>
  <book>
        <title>Администрирование ИС</title>
        <type>Лекции</type>
        <file>Лекции АИС.doc</file>
  </book>
  <book>
        <title>Администрирование ИС</title>
        <type>Лабораторные работы</type>
        <file>Лабораторные работы АИС.doc</file>
  </book>
</documents>
```

Отобразить содержимое XML-файла используя интерфейсы *DOM*.

Вариант 5

1. Дана часть схемы документа:

```
<xs:element name="student" maxOccurs="unbounded">
<xs:complexType>
<xs:sequence>
<xs:element name="name">
```

```
<xs:complexType>
<xs:sequence>
<xs:element name="surname" type="xs:string" />
<xs:element name="first" type="xs:string" />
<xs:element name="second" type="xs:string" />
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="faculty" type="xs:string" minOccurs="1" maxOccurs="unbounded"/>
<xs:element name="group" type="xs:string" minOccurs="1" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
```

Написать фрагмент XML-файла удовлетворяющий данному фрагменту.

2. Дан XML-*файл*:

```
<?xml version="1.0" encoding="windows-1251" ?>
<?xml-stylesheet type='text/xsl' href='1.xsl'?>
<documents>
  <book>
        <title>Языки информационного обмена </title>
        <type> Лекции </type>
        <file>Лекции ЯИО.doc</ file >
< file >Лекции ЯИО.pdf</ file >
< file >Лекции ЯИО.html</ file >
  </book>
  <book>
        <title>Языки информационного обмена </title>
        <type>Лабораторные работы</type>
        < file > Лабораторные работы ЯИО.doc </file >
< file > Лабораторные работы ЯИО.html </ file >
  </book>
</documents>
```

Написать *файл* 1.xsl для отображения содержимого XML-файла в следующем виде:

Название: Языки информационного обмена
Описание: Лекции
Электронная версия: Лекции ЯИО.doc, Лекции ЯИО.pdf, Лекции ЯИО.html.

Название: Языки информационного обмена
Описание: Лабораторные работы
Электронная версия: Лабораторные работы ЯИО.doc, Лабораторные работы ЯИО.html.

Тема 5 Реализация веб-сервисов на различных платформах. ИПК-6.1, ИПК-11.1, ИПК-13.2, ИПК-17.2

Вопросы опроса

№	Вопрос	Ответ
1	Что такое сервис-ориентированная архитектура (SOA) и как она связана с веб-сервисами?	
2	Какие протоколы используются в веб-сервисах?	
3	В чём разница между SOAP и RESTful-архитектурой при реализации веб-сервисов?	
4	Какие платформы и фреймворки используются для разработки веб-сервисов?	
5	Какие этапы включает разработка веб-сервиса?	

<u>Контрольные задания с развернутым ответом</u>

Практическая работа №7 Разработка Web-службы в ASPNET

Контрольное задание

Создайте *ASP.NET* веб-службу, поддерживающую три метода, возвращающие соответственно название текущего дня недели, номер текущего дня в месяце, номер текущего дня в году.

Для получения необходимых данных на стороне сервера можно использовать свойства и методы:

```
Label1.Text = DateTime.Now.DayOfWeek.ToString();
Label1.Text = DateTime.Now.Day.ToString();
Label1.Text = DateTime.Now.DayOfYear.ToString();
```

Практическая работа №8 Разработка веб-службы в ASP.NET. Создание прокси-сборки для веб-службы

Порядок выполнения работы

1. Создание веб-службы

2.Создание прокси-сборки.

Контрольное задание

Создайте *ASP.NET* прокси-сборку на языке C++, Python, JavaScript для веб-службы, поддерживающей три метода, возвращающие соответственно название текущего дня недели, номер текущего дня в месяце, номер текущего дня в году.

Для получения необходимых данных на стороне сервера можно использовать свойства и методы:

```
Label1.Text = DateTime.Now.DayOfWeek.ToString();
Label1.Text = DateTime.Now.Day.ToString();
Label1.Text = DateTime.Now.DayOfYear.ToString();
```

Выбор метода веб-службы должен быть реализован через форму с элементом типа ListBox или RadioButton.

Практическая работа №9 Примеры разработки RSS-источников и RSS-ридеров

Порядок выполнения работы

1. Создания RSS документа с помощью PHP сценария и с использованием интерфейса DOM XML.

Подготовьте файл, содержащий сценарий на языке PHP:

```
<?php
// Создается новый XML документ.
$dfeed = new DOMDocument('1.0', 'utf-8');

// Создание корневого элемента  <rss>
$erss = $dfeed->createElement('rss');
$erss->setAttribute('version', '2.0');

//  И добавление его в дерево документа
$dfeed->appendChild($erss);

// Создание элемента <channel> и добавление его к <rss>
$echannel = $dfeed->createElement('channel');
$erss->appendChild($echannel);

// Создание и добавление в channel
// узлов <title>, <link>, <description>, <language>, <pubDate>
$echannel->appendChild( $dfeed->createElement('title', 'RSS-channel title') );
$echannel->appendChild( $dfeed ->createElement('link', 'http://www.myhp.edu') );
```

```
$echannel->appendChild( $dfeed ->createElement('description', 'my RSS demo') );
$echannel->appendChild( $dfeed ->createElement('language', 'en') );
$echannel->appendChild( $dfeed ->createElement('pubDate', date('r')) );

// Добавление к узлу <channel> 2 узлов <item>
for ( $i = 1; $i <= 2; $i++ )
{
   $eitem = $dfeed->createElement('item');
   $echannel->appendChild($eitem);

   // Создание дочерних элементов для <item>
   $eitem->appendChild( $dfeed->createElement('title', 'Item'.$i) );
   $eitem->appendChild( $dfeed->createElement('link', 'http://www.myhp.edu/rss/'.$i.'.xml') );
   $eitem->appendChild( $dfeed->createElement('description', 'Description for '.$i.' item') );
}

// Сохранение документа в файле demo.rss
$dfeed->save('demo.rss');
?>
```

Созданный файл разместите на веб-сервере, настройте права доступа (право на запись) для веб-сервера к директории, в которой размещается сценарий, либо к специальной директории, в которой будет создан файл demo.rss (это более безопасно).

Выполните сценарий по запросу с веб-браузера.

В результате выполнения сценария получится следующий документ:

В результате выполнения сценария получится следующий документ:

```
<?xml version="1.0" encoding="utf-8" ?>
<rss version="2.0">
 <channel>
  <title>RSS-channel title</title>
  <link>http://www.myhp.edu</link>
  <description>my RSS demo</description>
  <language>en</language>
  <pubDate>Thu, 06 Nov 2008 01:50:53 +0300</pubDate>
<item>
  <title>Item1</title>
  <link>http://www.myhp.edu/rss/1.xml</link>
  <description>Description for 1 item</description>
  </item>
 <item>
  <title>Item2</title>
```

```
<link>http://www.myhp.edu/rss/2.xml</link>
<description>Description for 2 item</description>
</item>
</channel>
</rss>
```

2.Чтение RSS документа с помощью PHP сценария и с использованием интерфейса DOM XML.

Подготовьте файл, содержащий сценарий на языке PHP:

```
<?php
// Загрузка документа
$dfeed = DOMDocument::load('demo.rss');

// Чтение элемента <channel>
$echannel = $dfeed->firstChild->firstChild;

// Чтение первого дочернего элемента узла <channel> т.е.< title>
$cchild = $echannel->firstChild;

// Вывод содержимого дочерних элементов узла <channel> до элемента <item>:

print "<strong><br>";
while ( $cchild->tagName != 'item' )
{
   print $cchild->nodeValue. "<br>";
   $cchild = $cchild->nextSibling;
}
print "</strong><br>";

// Получение списка узлов <item>
$litems = $echannel->getElementsByTagName('item');

print "<br>";
foreach ( $litems as $eitem )
{
  $title = $eitem->firstChild->nodeValue;
  $link = $eitem->firstChild->nextSibling->nodeValue;
  $descr = $eitem->lastChild->nodeValue;

print "<a href=".$link.">".$title."</a> :: ".$descr."<br>";
}
print "<br>";

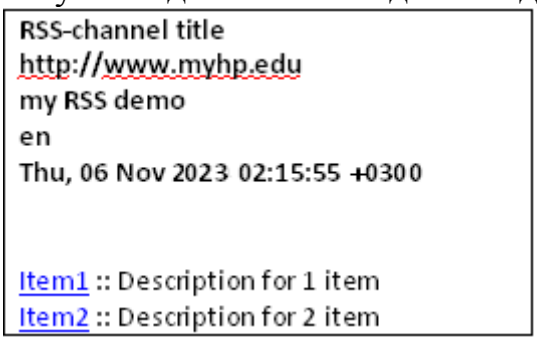
?>
```

В методе load() в качестве параметра укажите путь к RSS-файлу.

Созданный файл разместите на веб-сервере, настройте права доступа (право на запись) для веб-сервера к директории, в которой размещается сценарий, либо к специальной директории, в которой будет создан файл demo.rss (это более безопасно).

Выполните сценарий по запросу с веб-браузера.

Результат должен выглядеть следующим образом:



Контрольное задание

Разработайте PHP-сценарий, генерирующий RSS документ, содержащий описание веб-страниц на вашем локальном веб-сайте. В качестве источника информации для сценария можно использовать либо заранее подготовленный текстовый файл, либо функции PHP из библиотеки для работы с файловой системой (см., например: http://www.softtime.ru/bookphp/help.php)

Практическая работа №10 Реализация асинхронного взаимодействия веб-браузера с веб-сервером с помощью технологии AJAX

Порядок выполнения работы

В данной работе рассматривается пример системы, имитирующей работу сервиса *Google Suggest* на основе *AJAX*.

1. Реализация клиентской части.

Предполагается, что *пользователь* может вводить в текстовое *поле* формы название автомобильной марки, получая при этом динамически *список* вариантов названий, соответствующих уже введенным символам, без перезагрузки страницы.

1. Создайте следующую веб-страницу:

```
<html>
<head>
 <script src="chint.js"></script>
  </head>
   <body>
    <form>
```

```
8.  First Name:
9.  <input type="text" id="txt1"
onkeyup="showHint(this.value)">
10. </form>
11. <p>Suggestions: <span id="txtHint"></span></p>
12. </body>
</html>
```

Как видно из кода, при наступлении события onkeyup (отжатие клавиши) вызывается обработчик showHint().

13. В файле chint.js сохраните следующий код обработчика

Код обработчика загружается из файла chint.js:

```
var xmlHttp;
function showHint(str)
{
if (str.length==0)
  {
  document.getElementById("txtHint").innerHTML="";
  return;
  }
xmlHttp=GetXmlHttpObject();
if (xmlHttp==null)
  {
  alert ("Your browser does not support AJAX!");
  return;
  }
var url = "ghint.php";
url = url + "?q=" + str;
url = url + "&sid=" + Math.random();
xmlHttp.onreadystatechange = stateChanged;
xmlHttp.open("GET", url, true);
xmlHttp.send(null);
}
function stateChanged()
{
if (xmlHttp.readyState==4)
{
document.getElementById("txtHint").innerHTML =
xmlHttp.responseText;
}
}
function GetXmlHttpObject()
{
var xmlHttp=null;
try
```

```
  {
  // Firefox, Opera 8.0+, Safari
  xmlHttp = new XMLHttpRequest();
  }
catch (e)
  {
  // Internet Explorer
  try
    {
    xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");
    }
  catch (e)
    {
    xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
  }
return xmlHttp;
}
```

Из кода видно, что каждый раз, когда вводится символ, вызывается функция-обработчик. Если при этом содержимое текстового поля формы непустое (str.length > 0), функция выполняет следующие действия:

- Формируется *url* для отправки веб-серверу
- Добавляется значение параметра q, равное содержимому текстового поля, к *url*
- Добавляется к *url* случайное число для предотвращения *кеширования*
- Создается объект XMLHTTP, при этом указывается функция (stateChanged) подлежащая исполнению при наступлении события ввода символа
- Открывается объект XMLHTTP с указанным значением *url*
- Отправляется HTTP запрос веб-серверу

Если поле ввода пустое, происходит очистка содержимого раздела txtHint на веб-странице.

Ключевым моментом в данной системе является использование объекта *XMLHttpRequest*.

Данный объект по-разному создается в различных браузерах. Так, *Internet Explorer* для этого использует *ActiveXObject*, в то время как остальные браузеры используют встроенный в JavaScript объект *XMLHttpRequest*.

Для поддержки работы системы в разных браузерах использован оператор " try-catch ".

- Сначала делается попытка создать объект *XMLHttpRequest* для браузеров *Firefox*, *Opera* или *Safari*:

```
xmlHttp = new XMLHttpRequest().
```

- В случае неудачи, делается следующая попытка создания объекта для *Internet Explorer 6.0* + :

```
xmlHttp = new ActiveXObject("Msxml2.XMLHTTP").
```

- Если это также не удается, то делается попытка создания объекта уже для *Internet Explorer 5.5* + :

```
xmlHttp = new ActiveXObject("Microsoft.XMLHTTP").
```

- В случае, если ни одна из этих попыток не принесла успеха, выдается сообщение об отсутствии поддержки AJAX браузером.

2. Реализация серверной части.

Разместите на веб-сервере в файле с именем ghint.php следующий *PHP сценарий*::

```
<?php
header("Cache-Control: no-cache, must-revalidate");
 // Прошедшая дата
header("Expires: Mon, 1 Sep 2008 07:30:00 GMT");
// Инициализация массива названий
$a[]="Audi";
$a[]="BMW";
$a[]="Buick";
$a[]="Chevrolet";
$a[]="Citroen";
$a[]="Dodge";
$a[]="Ferrari";
$a[]="Fiat";
$a[]="Ford";
$a[]="Honda";
$a[]="Hyundai";
$a[]="Cherokee";
$a[]="Cherry";
$a[]="Lada";
$a[]="Lamborghini";
$a[]="Lincoln";
$a[]="Mazda";
$a[]="Mercedes";
$a[]="Mitsubishi";
$a[]="Nissan";
$a[]="Opel";
$a[]="Peugeot";
$a[]="Plymoth";
$a[]="Pontiac";
$a[]="Renault";
```

```
$a[]="Rover";
$a[]="Saab";
$a[]="Subaru";
$a[]="Suzuki";
$a[]="Toyota";
$a[]="Volkswagen";
$a[]="Volvo";
//получение параметра q из URL
$q = $_GET["q"];
//поиск соответствий из массива если длина q > 0
if (strlen($q) > 0)
{
  $hint = "";
  for($i = 0; $i<count($a); $i++)
  {
  if (strtolower($q) == strtolower(substr($a[$i],0,strlen($q))))
    {
    if ($hint == "")
      {
      $hint=$a[$i];
      }
    else
      {
      $hint=$hint." , ".$a[$i];
      }
    }
  }
}
// Возврат строки "нет вариантов" если соответствий не найдено
// либо найденное соответствие
if ($hint == "")
{
$response = "no suggestion";
}
else
{
$response = $hint;
}
//вывод результата
echo $response;
?>
```

3. Проверка работоспособности системы.

Проверьте с помощью подходящего веб-браузера работу системы.

На скриншоте показана работа в браузере с веб-страницей ghint.html, использующей *AJAX*:

Контрольное задание

С использованием PHP-сценария №7 из практической работы 9 разработайте модифицированный вариант калькулятора для четырех арифметических операций, в котором происходит асинхронная передачи данных между браузером и сервером (с помощью AJAX).

На скриншоте выше показана работа в браузере с веб-страницей ghint.html, использующей AJAX:

Практическая работа №11 Создание мобильных веб-приложений при помощи ASP.NET

Порядок выполнения работы

Для работы с примерами, приводимыми в данной работе помимо среды разработки *Visual Studio .NET* потребуется установка компонентов *.NET Mobile* и *WAP* эмулятора.

В качестве *WAP* эмулятора был использован Series 40 5th Edition SDK (на основе *Nokia 6267*). Для установки эмулятора необходимо скачать с сайта http://www.*forum*.nokia.com/ (требуется предварительная *регистрация*) и установить программу Nokia *Mobile Internet Toolkit* и сам эмулятор Nokia *Series* 40 5th *Edition SDK*.

Общая последовательность разработки веб-приложения с использованием *.NET Mobile* выглядит следующим образом:

- Создание проекта веб-приложения ASP.NET.
- Подключение интерфейса System.Mobile.UI.
- Вставка в веб-страницу управляющих элементов .NET Mobile.

Рассмотрим пример создания простейшей веб-странице, отображающей на экране мобильного устройства сообщение " Hello world!".

Создайте новый проект типа ASP.NET Web Application.

Активируйте с помощью меню окно "Solution Explorer". Из списка файлов проекта удалите файлы Default.aspx.cs и Default.aspx.designer.cs.

Страница Default.aspx должна содержать следующий код:

```
<%@ Page Inherits= "System.Web.UI.MobileControls.MobilePage " Language= "C# "%>
<%@ Register TagPrefix= "mobile" Namespace= "System.Web.UI.MobileControls "
Assembly= "System.Web.Mobile " %>

<mobile:Form runat="server">
  <mobile:Label runat="server">Hello world!</mobile:Label>
```

```
</mobile:Form>
```

Директива в первой строке сообщает обработчику ASP о необходимости обработки данной страницы как предназначенной для мобильных устройств.

Директива во второй строке регистрирует префикс "mobile" для обращения к элементами управления для мобильных устройств.

Тэг <mobile:Form> указывает серверу на необходимость вставки формы.

Тэг <mobile: Label> указывает серверу на необходимость вставки метки с текстом " Hello world!".

Откомпилируйте проект.

Создайте новый виртуальный каталог, связанный с директорией текущего проекта. Добавьте в список документов по умолчанию страницу Default.aspx.

Запустите эмулятор Nokia *Series* 40 5th Edition SDK и программу Nokia *Mobile Internet* Toolkit. В программе Nokia *Mobile Internet* Toolkit укажите полный URL созданной с помощью ASP.NET страницы Default.aspx.

После обработки данной страницы для мобильного устройства, поддерживающего протокол *WAP*, будет сгенерирована страница, вид которой в эмуляторе приведен ниже:

В эмуляторе выберите раздел меню "Tools > Diagnostics". В появившемся окне диагностики можно будет увидеть следующий код страницы, загруженной в эмуляторе:

```
<?xml version='1.0'?>
 <!DOCTYPE wml PUBLIC '-//WAPFORUM//DTD WML 1.1//EN'
 'http://www.wapforum.org/DTD/wml_1.1.xml'>
 <wml>
  <head>
   <meta http-equiv="Cache-Control" content="max-age=0" />
  </head>
  <card>
   <p>Hello world!</p>
  </card>
 </wml>
```

Следующий пример демонстрирует использование простого элемента формы для ввода текста (в режиме пароля):

```
<%@ Page Inherits= "System.Web.UI.MobileControls.MobilePage "
Language= "C# "%>
<%@ Register TagPrefix= "mobile " Namespace=
"System.Web.UI.MobileControls "
Assembly= "System.Web.Mobile " %>

<script runat="server" language="C#">
```

```
string uname;

protected void AgeClick(Object sender, EventArgs e)
{
   uname=text1.Text;
   ActiveForm=form2;
}

protected void  Form2_Activate(Object sender, EventArgs e)
{
 message.Text="Wellcome " + uname + "!";
}
</script>

<Mobile:Form id="form1" runat="server">
<Mobile:Label runat="server">Your name, please</Mobile:Label>
<Mobile:TextBox runat="server" id="text1" password="true"/>
<Mobile:Command  runat="server"  OnClick="AgeClick"  Text="Submit"
/>
</Mobile:Form>

<Mobile:Form                id="form2"                runat="server"
OnActivate="Form2_Activate">
<Mobile:Label runat="server" id="message" />
</Mobile:Form>
```

Если выполнить как в предыдущем примере все шаги с 1 по 8, то после загрузки получившейся страницы в эмуляторе на экране отобразится первая форма:



После ввода строки и нажатия кнопки " *Submit* " активируется вторая форма:



Код страницы для первой формы, загруженный в мобильное устройство, выглядит следующим образом:

```
<?xml version='1.0'?>
<!DOCTYPE wml PUBLIC '-//WAPFORUM//DTD WML 1.1//EN'
'http://www.wapforum.org/DTD/wml_1.1.xml'>
<wml>
 <head>
  <meta http-equiv="Cache-Control" content="max-age=0" />
 </head>
 <card>
  <do type="prev" label="PќP°P•P°Pr"><prev /></do>
  <p>Your name, please<br/>
  <input name="mcsvprtws0" type="password" value="" />
  <anchor title="P'PїPµCЂPµPr">Submit
  <go href="Default.aspx?__ufps=058985" method="post">
  <postfield name="__EVENTTARGET" value="ctl01" />
  <postfield name="text1" value="$(mcsvprtws0)" />
  </go>
  </anchor>
  </p>
 </card>
</wml>
```

Контрольное задание

Разработайте с помощью MS Visual Studio .NET серверное веб-приложение, использующее управляющий ASP.NET Mobile

элемент: <Mobile:Calendar> Добавьте для этого элемента обработчик события OnSelectionChanged, который должен отображать выбранный пользователем мобильного устройства день календаря с помощью элемента <Mobile:Label>.

Код обработчика события может выглядеть следующим образом:

```
<script runat="server">

protected void CalChanged(Object sender, EventArgs e)
{
   label1.Text = "You selected: " + calendar1.SelectedDate;
}
</script>
```

Создайте необходимый виртуальный каталог на веб-сервере и проверьте работу приложения с помощью эмулятора мобильного устройства.

Практическая работа №12 Инструменты создания Web-ресурсов

Цели: Ознакомиться с технологией создания веб-сайтов, создать сайт по теме «Компьютерные вирусы. Антивирусные программы».

Задачи:

- ознакомиться с основными терминами электронного обучения
- на основе карты сайта из практического занятия 6, создайте иерархическую структуру сайта (разделы и страницы)
- поберите и разместите материалы на страницах сайта

Содержание сайта

В рамках данной работы вам предстоит разработать электронный образовательный ресурс по теме «Компьютерные вирусы. Антивирусные программы». По карте сайта из работы №6 вам необходимо создать набор связанных друг с другом страниц (сайт) при помощи инструментов Google. Материалы для сайта можно найти в Интернете (презентации, видеоролики, изображения, текст...), однако ваша главная задача – правильно структурировать и отформатировать материал для сайта.

! Абсолютно не допускается бездумное копирование текста с различных сайтов без обработки.

Примерный план работы над сайтом может быть таким:

1. Продумайте содержание и структуру сайта (см. карту сайта)
2. Создайте новый сайт в среде Google.
3. Используя интернет-ресурсы, подберите материал и разместите его на страницах.
4. Опубликуйте видеоролики и презентацию в блоге.

План работы:

1) Создание сайта

2) Редактирование страниц

3) Добавление страниц, создание иерархической структуры

4) Настройка

Требования к содержанию сайта

1. На главной странице сайта разместить ссылки на страницы с презентацией и видео-роликом и краткую информацию об авторе и назначении сайта.
2. На сайте обязательно должны быть встроены **рисунки**, **видео** и **презентация**.
3. Встроить в одну из страниц форму из работы №8. На главную страницу сайта добавить ссылку, ведущую к странице с формой.
4. На странице "Об авторе" разместить логотип.

Практическая работа №13 Интеграция ASP.NET Web Forms, MVC и веб-API

В этой практической работе вы узнаете, как выполнять следующие задачи:

- Создание веб-сайта на основе одного типа проекта **ASP.NET.**
- Использование различных платформ **ASP.NET** , таких как **MVC** и **веб-API** , в одном проекте.
- Поиск основных компонентов приложения **ASP.NET.**
- Воспользуйтесь преимуществами платформы **формирования шаблонов ASP.NET** , чтобы автоматически создавать контроллеры и представления для выполнения операций CRUD на основе классов модели.
- Предоставление одинакового набора сведений в форматах машинного и удобного для чтения данных с помощью правильного инструмента для каждого задания.

Эта практическая работа включает в себя следующие упражнения:

1. Создание нового проекта веб-форм
2. Создание контроллера MVC с помощью формирования шаблонов
3. Создание контроллера веб-API с помощью формирования шаблонов

Упражнение 1. Создание нового проекта веб-форм

В этом упражнении вы создадите новый сайт Web Forms в Visual Studio 2013 с помощью единого унифицированного проекта **ASP.NET** , который позволит легко интегрировать веб-формы, MVC и компоненты веб-API в одно приложение. Затем будет рассмотрено созданное решение и его части, и, наконец, вы увидите, что веб-сайт в действии.

Задача 1. Создание нового сайта с помощью одного интерфейса ASP.NET

В этой задаче вы приступите к созданию нового веб-сайта в Visual Studio на основе одного типа проекта **ASP.NET** . **Один ASP.NET** объединяет все технологии ASP.NET и дает возможность смешивать и сопоставлять их по своему усмотрению. Затем вы узнаете о различных компонентах веб-форм, MVC и веб-API, которые параллельно находятся в приложении.

Упражнение 2. Создание контроллера MVC с помощью формирования шаблонов

В этом упражнении вы будете использовать преимущества платформы формирования шаблонов ASP.NET, предоставляемой Visual Studio для создания контроллера ASP.NET MVC 5 с действиями и представлениями Razor для выполнения операций CRUD, не написав ни одной строки кода. Процесс формирования шаблонов будет использовать Entity Framework Code First для создания контекста данных и схемы базы данных в базе данных SQL.

Задача 1. Создание новой модели

Задача 2. Создание контроллера MVC

Задача 3. Запуск решения

Упражнение 3. Создание контроллера веб-API с помощью формирования шаблонов

Платформа веб-API является частью стека ASP.NET и предназначена для упрощения реализации служб HTTP, которая обычно отправляет и получает данные в формате JSON или XML через API RESTFUL.

В этом упражнении для создания контроллера веб-API будет использоваться формирование шаблонов ASP.NET. Вы будете использовать те же классы **Person** и **персонконтекст** из предыдущего упражнения, чтобы предоставить одни и те же данные пользователя в формате JSON. Вы увидите, как можно предоставить одни и те же ресурсы различными способами в рамках одного приложения ASP.NET.

Задача 1. Создание контроллера веб-API

Задача 2. Запуск решения

Задача 3. Добавление страниц справки по веб-API

5.3. Один или несколько тематических блоков дисциплины завершаются контрольной точкой (далее – КТ). Текущий контроль успеваемости по дисциплине предусматривает не менее 2 (двух) и не более 10 (десяти) КТ в течение периода освоения дисциплины.

Максимальное количество баллов за любой тип работ в рамках КТ составляет 100 (сто) баллов.

Распределение весовых коэффициентов по КТ в рамках текущего контроля успеваемости по дисциплине и формулы расчета:

Наименование контрольной точки	Максимальное количество баллов за работу в рамках КТ, которое может набрать студент	Коэффициент веса контрольной точки	Результат контрольной точки, участвующий в формировании итоговой балльной оценки по дисциплине (отражается в журнале БРС в СДО)
КТ 1	100	0,2	20
КТ 2	100	0,2	20
КТ 3	100	0,2	20
Итого:	х	0,6	60

Формула расчета результата контрольной точки:
Результат контрольной точки = Количество баллов за работу в рамках КТ х Коэффициент веса контрольной точки.

5.4. Формы текущего контроля успеваемости обучающихся в рамках КТ и типовые оценочные материалы:

КТ – 1.

Тема 1 и2

Опрос:

Вопросы для опроса:

№ п.п.	Содержание вопроса
1	Что такое предоставление государственных и муниципальных услуг в электронной форме согласно Федеральному закону от 27.07.2010 №210-ФЗ?
2	Какие информационные системы используются для предоставления услуг в электронной форме?
3	Какие данные содержатся в информационной карточке услуги на портале?
4	Какие требования предъявляются к инфраструктуре, обеспечивающей информационно-технологическое взаимодействие информационных систем?
5	Какие документы необходимы для регистрации на Едином портале государственных услуг (ЕПГУ)?
6	Какие правила применяются к именованию элементов в XML? (
7	Как записываются значения атрибутов в XML?
8	Какой синтаксис используется для комментариев в XML?

№ п.п.	Содержание вопроса
9	Чем отличается корректно сформированный (well-formed) документ от действительного (valid)?
10	Что такое пространства имён в XML и для чего они нужны?

Критерии оценивания опроса:

Диапазон баллов	Описание критерия
85-100	Обучающийся полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; технически излагает материал.
65-84	Обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого.
55-64	Обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.
0-54	Обучающийся обнаруживает незнание вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

КТ – 2.

Тема 3 и 4

Опрос:

Вопросы для опроса:

№ п.п.	Содержание вопроса
1.	Какова основная цель веб-сервисов?
2.	Какой протокол обычно используется для веб-сервисов?
3.	Что означает WSDL?
4.	SOAP — это аббревиатура от какого из следующих понятий?

№ п.п.	Содержание вопроса
5.	Сервисы RESTful в первую очередь основаны на каком архитектурном стиле?
6.	Как XML используется для интеграции информационных систем? Приведите пример реализации такой интеграции.
7.	Какие задачи решает интеграция данных и массивов элементов в информационных системах? Как это связано с универсальным интерфейсом распознавания и описания?
8.	В чём заключается роль UDDI в контексте веб-сервисов? Какие данные он хранит и как используется для поиска веб-сервисов?
9.	Какие проблемы могут возникать при интеграции различных информационных систем и как UDDI помогает их решать?
10.	Как универсальный интерфейс распознавания и описания способствует автоматизации бизнес-процессов в информационных системах?

Критерии оценивания опроса:

Диапазон баллов	Описание критерия
85-100	Обучающийся полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; технически правильно излагает материал.
65-84	Обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого.
55-64	Обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.
0-54	Обучающийся обнаруживает незнание вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

КТ – 3.

Тема 5

Опрос:

Вопросы для опроса:

№ п.п.	Содержание вопроса
1.	Какие платформы управления контентом (CMS) могут использоваться для реализации веб-сервисов?
2.	Какие принципы важны при разработке веб-сервисов?
3.	Какие инструменты помогают тестировать веб-сервисы?
4.	В чём преимущества использования определённых платформ или фреймворков при разработке веб-сервисов?
5.	Какие проблемы могут возникнуть при реализации веб-сервисов и как их можно решить?

Критерии оценивания опроса:

Диапазон баллов	Описание критерия
85-100	Обучающийся полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; технически правильно излагает материал.
65-84	Обучающийся дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого.
55-64	Обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.
0-54	Обучающийся обнаруживает незнание вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

5.5. Описание дополнительных материалов и оборудования, необходимых для выполнения проверочных заданий.

Обучающемуся разрешается использование компьютера с установленными программными продуктами Web-проектирования.

.

6. Формы промежуточной аттестации, критерии и шкала оценивания, типовые оценочные материалы по дисциплине

6.1. Промежуточная аттестация проводится в форме зачета.

6.2. Типовые оценочные материалы промежуточной аттестации.

Тема 1 Введение в веб-сервисы. ИПК-17.2

1. Задания открытого типа.

1.1. Вопросы открытого типа.

№ п.п.	Вопрос	Ответ
1.	Архитектура электронного правительства.	
2.	Реализация «Стратегии развития информационного общества».	
3	Предоставление государственных и муниципальных услуг органами власти средствами веб-сервисов и мобильных приложений.	
4	Веб-сервисы как способы вывода информации и данных.	

1.2. Контрольные задания с ключами правильных ответов.

Задание 1

Улучшение существующего сервиса. Задание на анализ конкретного государственного или муниципального сервиса (портала, приложения) и подготовку предложений по его улучшению (например, добавление новых функций, оптимизация интерфейса, интеграция с другими системами).

Задание 2

Создание прототипа веб-сервиса. Используя инструменты для прототипирования (например, Figma или Adobe XD), требуется разработать макет веб-страницы или мобильного приложения для предоставления определённой услуги (например, для подачи заявления на субсидию или получения информации о местных мероприятиях).

2. Задания закрытого типа.

2.1. Тестовые задания.

Тест 1.

Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов.

Внимательно прочитать предложенные варианты ответа.

Выбрать один верный ответ.

Записать только букву выбранного варианта ответа.

1. ***Как называется единый портал государственных и муниципальных услуг РФ?***

а) «Госуслуги»;

б) «Мой налог»;

в) «Правосудие онлайн»;

г) «Электронный бюджет».

2. ***Какой закон регулирует организацию предоставления государственных и муниципальных услуг в РФ?***

а) ФЗ № 152-ФЗ «О персональных данных»;

б) ФЗ № 210-ФЗ «Об организации предоставления государственных и муниципальных услуг»;

в) ФЗ № 59-ФЗ «О порядке рассмотрения обращений граждан»;

г) ФЗ № 323-ФЗ «Об основах охраны здоровья граждан».

3. ***Что из перечисленного НЕ является преимуществом электронного предоставления госуслуг?***

а) сокращение времени оказания услуги;

б) необходимость личного посещения нескольких инстанций;

в) реализация принципа «одного окна»;

г) возможность оплаты госпошлин онлайн.

4. ***Какая система обеспечивает информационно-технологическое взаимодействие при предоставлении госуслуг в электронной форме?***

а) ЕСИА (Единая система идентификации и аутентификации);

б) ГИС ГМП (Государственная информационная система о государственных и муниципальных платежах);

в) ФГИС ПГС (Федеральная государственная информационная система предоставления госуслуг);

г) всё вышеперечисленное.

5. ***Какой процент скидки на оплату госпошлин доступен при оплате через портал госуслуг?***

а) 10 %;

б) 20 %;

в) 30 %;

г) 50 %.

Тест 2.

Внимательно прочитать предложенные варианты ответа.

Выбрать верные ответы.

Записать буквы выбранного варианта ответа.

6. ***Какие функции доступны на портале госуслуг? Выберите все верные варианты:***

а) подача заявлений на получение услуг;

б) оплата госпошлин и штрафов;

в) получение информации о начислениях;

г) запись на приём к врачу;

д) всё вышеперечисленное.

7. ***Какие категории пользователей могут использовать портал госуслуг?***

а) физические лица;

б) юридические лица;

в) индивидуальные предприниматели;

г) только государственные служащие.

8. ***Какие данные обычно требуются для регистрации на портале госуслуг?***

а) ФИО;

б) номер телефона;

в) адрес электронной почты;

г) СНИЛС;

д) все вышеперечисленные.

Тема 2 Основы XML: семантика и синтаксис. ИПК-6.1, ИПК-11.1, ИПК-17.2

1. Задания открытого типа.

1.1. Вопросы открытого типа.

№ п.п.	Вопрос	Ответ
1.	Какие языки описания схем используются для валидации XML-документов?	
2.	Как в XML задаётся структура документа с помощью схемы?	
3	Какие символы являются зарезервированными в XML?	
4	Чем отличаются элементы от атрибутов в XML?	

№ п.п.	Вопрос	Ответ
5	Как в XML обрабатываются пустые элементы?	

1.2. Контрольные задания с ключами правильных ответов.

Задание 1

Найдите ошибки в следующем XML-фрагменте и исправьте их:

xml

<book>

<title>Война и мир

<author>Лев Толстой</author>

<year>1869</year>

</book>

Задание 2

Добавьте атрибут id со значением 123 к элементу <book> в приведён ном фрагменте:

xml

<library>

<book>

<title>Мастер и Маргарита</title>

<author>Михаил Булгаков</author>

</book>

</library>

Задание 3

Создайте корректный XML-документ с корневым элементом <catalog >. Внутри него разместите два элемента <product>, каждый из которых сод ержит:

- элемент <name> с названием товара;
- элемент <price> с ценой;
- атрибут category с категорией товара.

Задание 4

Экранируйте специальные символы в следующем фрагменте:

xml

<description>

Используйте символ < для начала тега и & для ссылок.

</description>

Задание 5

Добавьте декларацию XML версии 1.0 и кодировки UTF-8 в начало с ледующего документа:

xml

<message>Привет, мир!</message>

2. Задания закрытого типа.

2.1. Тестовые задания.

Тест 1.

Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов.

Внимательно прочитать предложенные варианты ответа.

Выбрать один верный ответ.

Записать только букву выбранного варианта ответа.

1. **Что означает аббревиатура XML?**
   а) eXtensible Markup Language
   б) Extended Markup Language
   в) External Markup Language

2. **Основное назначение XML:**
   а) отображение данных в веб-браузерах
   б) хранение и транспортировка данных
   в) создание веб-страниц с интерактивными элементами

3. **Какой элемент обязателен в любом XML-документе?**
   а) декларация XML
   б) корневой элемент
   в) хотя бы один атрибут

4. **Какие символы используются для обозначения открывающего т ега?**
   а) <!-- *...* *-->*
   б) < *...* *>*
   в) { ... }

5. **Должен ли каждый открывающий тег иметь соответствующий з акрывающий тег?**
   а) да, всегда

б) нет, только для некоторых тегов
в) закрывающие теги не обязательны

6. **Как записываются атрибуты в XML?**
а) внутри открывающего тега в формате имя=значение
б) внутри закрывающего тега
в) отдельным элементом

7. **Какая кодировка чаще всего используется в XML-документах?**
а) ASCII
б) UTF-8
в) Windows-1251

8. **Как добавить комментарий в XML-документ?**
а) # комментарий
б) // комментарий
в) <!-- комментарий -->

9. **Что такое пространства имён (namespaces) в XML?**
а) способ организации файлов на диске
б) механизм для устранения конфликтов имён тегов
в) специальные теги для комментариев

10. **Какие специальные символы нужно экранировать в XML?**
а) <, >, &
б) $, %, @
*в) #, *, +*

Тема 3 Язык описания веб-сервисов. ИПК-6.1, ИПК-11.1, ИПК-17.2

1. Задания открытого типа.

1.1. Вопросы открытого типа.

№ п.п.	Вопрос	Ответ
1.	Что такое SOAP?	
2.	Какие протоколы используются для защиты пользовательских данных при их передаче через Интернет?	
3	Что такое CORS в контексте веб-разработки?	
4	Какой протокол обеспечивает передачу гипертекстовых документов в сети?	

№ п.п.	Вопрос	Ответ
5	Что такое WebSocket и для чего он используется в веб-разработке?	
6	Какие стандарты регламентируют взаимодействие в рамках сервис-ориентированной архитектуры (SOA)?	
7	Что такое микросервисы и как они связаны с протоколами и стандартами веб-сервисов?	
8	Какие технологии используются для асинхронной обработки запросов в веб-приложениях?	

1.2. Контрольные задания с ключами правильных ответов.

Задание 1. Создать простое WSDL-описание для веб-сервиса

Задача: описать веб-сервис CurrencyConverter для конвертации валют . Сервис должен предоставлять один метод convert, который принимает:

- fromCurrency (строка, код исходной валюты);
- toCurrency (строка, код целевой валюты);
- amount (число, сумма для конвертации).

Метод возвращает число — сумму в целевой валюте.

Что нужно сделать:

1. Определить типы данных для входных и выходных параметров.
2. Описать сообщение для запроса (convertRequest) и ответа (convertResponse).
3. Описать операцию convert в порте (интерфейсе).
4. Указать привязку (binding) с использованием SOAP.
5. Указать адрес сервиса (endpoint).

Результат: WSDL-файл с корректной структурой, включающий разделы:

- <types>;
- <message>;
- <portType>;
- <binding>;
- <service>.

Задание 2. Проанализировать существующий WSDL-файл

Задача: взять публичный WSDL-файл (например, от веб-сервиса погоды или конвертера валют) и выполнить анализ.

Что нужно сделать:

1. Найти и выписать все операции (методы), предоставляемые сервисом.
2. Для каждой операции определить:
   - входные параметры и их типы;
   - выходной параметр и его тип.
3. Определить протокол и стиль взаимодействия (например, SOAP 1.1/1.2, RPC или Document/Literal).
4. Найти URL endpoint сервиса.
5. Кратко описать назначение сервиса на основе его WSDL-описания.

2. Задания закрытого типа.

2.1. Тестовые задания.

Тест 1.

Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов.

Внимательно прочитать предложенные варианты ответа.

Выбрать один верный ответ.

Записать только букву выбранного варианта ответа.

Вопрос 1.

Что такое WSDL?

а) Протокол передачи данных между веб-сервисами.

б) Язык описания веб-сервисов на основе XML.

в) Технология шифрования данных в веб-приложениях.

г) Фреймворк для разработки веб-интерфейсов.

Вопрос 2.

Какой элемент WSDL описывает абстрактный набор операций, которые может выполнять веб-сервис?

а) <types>

б) <message>

в) <portType>

г) <service>

Вопрос 3.

В каком элементе WSDL указывается URL веб-сервиса?

а) <binding>

б) <portType>

в) <service>

г) <message>

Вопрос 4.

Какая технология чаще всего используется совместно с WSDL дл я определения структуры данных?

а) JSON

б) XSD

в) YAML

г) CSV

Вопрос 5.

Верно ли, что WSDL-документ может содержать встроенное X SD-описание?

а) Да

б) Нет

Вопрос 6.

Выберите компонент, который не входит в стандартную струк туру WSDL 1.1.

а) <message>

б) <operation>

в) <binding>

г) <definitions>

Вопрос 7.

Для какого типа веб-сервисов преимущественно используется WS DL?

а) RESTful API

б) GraphQL API

в) SOAP-сервисы

г) gRPC-сервисы

Тема 4 Универсальный интерфейс распознавания, описания и интеграции. ИПК-6.1, ИПК-11.1, ИПК-13.2, ИПК-17.2

1. Задания открытого типа.

1.1. Вопросы открытого типа.

№ п.п.	Вопрос	Ответ
1.	Что означает аббревиатура UDDI? Дайте полное определение.	
2.	Какие три типа данных хранятся в реестре UDDI? Кратко опишите каждый.	
3	Как UDDI взаимодействует с SOAP и WSDL? Приведите пример цепочки действий.	
4	Перечислите основные функции UDDI-реестра.	
5	В чём преимущества использования UDDI для интеграции веб-сервисов?	
6	Почему UDDI не стал массово популярным стандартом? Укажите 2–3 причины.	
7	Опишите пошаговый сценарий поиска и вызова веб-сервиса через UDDI.	
8	Какие протоколы и стандарты лежат в основе работы UDDI?	
9	Где может применяться UDDI: только в публичных сетях или в корпоративных системах тоже? Аргументируйте ответ.	
10	Сравните UDDI и современные API-маркетплейсы (например, AWS API Gateway). В чём ключевые различия?	

1.2. Контрольные задания с ключами правильных ответов.

Задание 1. Регистрация веб-сервиса в UDDI-реестре.

- Задание: реализовать программу, которая регистрирует веб-сервис в UDDI-реестре. Необходимо создать объект Provider (бизнес-сущность), добавить к нему веб-сервис (Service), указать точку доступа (Binding) и, при наличии, файл WSDL как объект tModel.

- Инструменты: можно использовать UDDI SDK (например, для .NET или Java), SOAP-клиенты или веб-приложения для взаимодействия с UDDI-сервером.
- Требования: программа должна запрашивать у пользователя имя провайдера, название сервиса, URL и другие необходимые данные, а затем выполнять регистрацию в реестре.

Задание 2. Сравнение UDDI с современными технологиями интеграции.

- Задание: подготовить сравнительный анализ UDDI и современных подходов к интеграции систем (например, REST API, GraphQL, gRPC).
- Требования: в анализе должны быть рассмотрены такие аспекты, как простота использования, гибкость, поддержка современных стандартов, масштабируемость и область применения.

2. Задания закрытого типа.

2.1. Тестовые задания.

Тест 1.

Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов.

Внимательно прочитать предложенные варианты ответа.

Выбрать один верный ответ.

Записать только букву выбранного варианта ответа.

Вопрос 1.

Что означает аббревиатура UDDI?

а) Unified Data Delivery Interface

б) Universal Description, Discovery, and Integration

в) Universal Data Distribution Initiative

г) Unified Digital Development Infrastructure

Вопрос 2.

Какую роль выполняет UDDI в архитектуре веб-сервисов?

а) Протокол обмена сообщениями

б) Язык описания интерфейсов

в) Каталог/реестр для поиска и публикации сервисов

г) Технология шифрования данных

Вопрос 3.

Какой стандарт используется для передачи запросов к UDDI-реестру?

а) HTTP GET/POST

б) FTP

в) SOAP

г) SMTP

Вопрос 4.

Что такое tModel в контексте UDDI?

а) Шаблон для стандартизации описаний сервисов

б) Тип данных в XML-схеме

в) Протокол аутентификации

г) Инструмент тестирования API

Вопрос 5.

Какие данные хранятся в BindingTemplate?

а) Контактные данные компании

б) URL и технические детали подключения к сервису

в) Отзывы пользователей о сервисе

г) История обновлений веб-сервиса

Вопрос 6.

Верно ли, что UDDI может автоматически обнаруживать веб-сервисы без их ручной регистрации?

а) Да

б) Нет

Вопрос 7.

Какая технология дополняет UDDI, предоставляя детальное описание интерфейса сервиса?

а) JSON

б) WSDL

в) HTML

г) CSV

Вопрос 8.

Назовите один недостаток использования UDDI.

а) Высокая скорость работы

б) Большой объём сетевого трафика из-за XML

в) Полная совместимость с любыми протоколами

г) Отсутствие необходимости в стандартах

Тема 5 Реализация веб-сервисов на различных платформах. ИПК-6.1, ИПК-11.1, ИПК-13.2, ИПК-17.2

1. Задания открытого типа.

1.1. Вопросы открытого типа.

№ п.п.	Вопрос	Ответ
1.	Какие протоколы используются для взаимодействия между клиентом и сервером в веб-сервисах? Кратко охарактеризуйте кажды	

№ п.п.	Вопрос	Ответ
	й.	
2.	Что такое REST? Опишите его о сновные принципы и HTTP-мето ды (GET, POST, PUT, DELETE).	
3	Что такое SOAP? В чём его отли чия от REST? Когда предпочтит ельнее использовать SOAP?	
4	Что такое WSDL и UDDI? Для ч его они применяются?	
5	Что такое XML-RPC? Как он свя зан с SOAP? Почему XML-RPC менее популярен сегодня?	
6	Какие форматы данных использ уются для обмена информацией между веб-сервисами? Сравните их.	
7	В чём преимущества JSON пере д XML? Приведите примеры сит уаций, где предпочтительнее X ML.	
8	Как происходит сериализация и десериализация данных при обм ене между клиентом и сервером ?	
9	Как реализовать веб-сервис на Ja va с использованием JAX-WS? О пишите основные шаги и аннота ции (@WebService, @WebMetho d).	
10	Как создать REST-сервис на PH P? Приведите пример простого API.	
11	Какие инструменты и библиотек и помогают автоматизировать со здание WSDL-описаний и клиен тских прокси?	

1.2. Контрольные задания с ключами правильных ответов.

Задание 1

Создайте в Postman запрос к публичному API (например, JSONPlaceh older) и проверьте статус ответа 200 OK. Опишите шаги.

Задание 2

Кейс-задания

Моделируют реальные ситуации.

Пример:

Веб-сервис медленно загружается при одновременном доступе 1000 пользователей. Предложите 3 инструмента для диагностики проблемы и кр атко обоснуйте выбор.

Задание 3

Опишите 2 преимущества использования Docker при тестировании в еб-сервисов.

Задание 4

Напишите curl-команду для GET-запроса к API .

Задание 5

Кейс: После обновления фронтенда пользователи жалуются на ошиб ки 500. Предложите план диагностики (3–4 шага).

2. Задания закрытого типа.

2.1. Тестовые задания.

Тест 1.

Внимательно прочитать текст задания и понять, что в качестве ответа ожидается только один из предложенных вариантов.

Внимательно прочитать предложенные варианты ответа.

Выбрать один верный ответ.

1. ***Что такое API?***

а) Графический интерфейс пользователя
б) Интерфейс программирования приложений
в) Протокол передачи файлов

2. Выберите инструменты для UI-автоматизации:

а) [ ] Selenium
б) [ ] Cypress
в) [ ] OWASP ZAP
г) [ ] Playwright

3. Соотнесите:

1. REST API
2. SOAP
3. ***GraphQL***

а) XML-ориентированный протокол
б) Запросы для получения конкретных данных
в) Архитектурный стиль с HTTP-методами

4. ***Соответствие стандартам HTML-кода – это…***

а) соответствие стандарту

б) кроссбраузерность

в) гибкость вёрстки

5. ***Какой вид сервера подразумевает размещение на одном физическом сервере файлов сайтов сразу нескольких владельцев, ресурсы распределены между всеми пользователями на одном сервере?***

а) виртуальный

б) выделенный

в) виртуальный выделенный

6. Платформы-конструкторы для создания сайтов

а) https://www.wix.com;

б) https://www.vk.com;

в) https://www.gsuite.google.ru

г) https://www.site123.com;

д) https://www.ok.ru;

е) https://www.jimdo.com.

6.3. Критерии и шкала оценивания на основе БРС.

Критерии и балльная шкала определяются преподавателем

КРИТЕРИИ ОЦЕНИВАНИЯ	РЕЗУЛЬТАТ В БАЛЛАХ
Дан полный, в логической последовательности развернутый ответ на поставленный вопрос, где он продемонстрировал знания предмета в полном объеме учебной программы, достаточно глубоко осмысливает дисциплину, самостоятельно, и исчерпывающе отвечает на дополнительные вопросы, приводит собственные примеры по проблематике поставленного вопроса, решил предложенные практические задания без ошибок	40
Дан развернутый ответ на поставленный вопрос, где студент демонстрирует знания, приобретенные на лекционных и семинарских занятиях, а также полученные посредством изучения обязательных учебных материалов по курсу, дает аргументированные ответы, приводит примеры, в ответе присутствует свободное владение монологической речью, логичность и последовательность ответа. Однако допускается неточность в ответе. Решил предложенные практические задания с небольшими неточностями.	30-39
Дан ответ, свидетельствующий в основном о знании процессов изучаемой дисциплины, отличающийся недостаточной глубиной и полнотой раскрытия темы, знанием основных вопросов теории, слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры, недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается	20-29

КРИТЕРИИ ОЦЕНИВАНИЯ	РЕЗУЛЬТАТ В БАЛЛАХ
несколько ошибок в содержании ответа и решении практических заданий.	
Дан ответ, который содержит ряд серьезных неточностей, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы, незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов, неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Выводы поверхностны. Решение практических заданий не выполнено, т.е. студент не способен ответить на вопросы даже при дополнительных наводящих вопросах преподавателя.	0-19

6.4. Описание дополнительных материалов и оборудования, необходимых для выполнения проверочных заданий.

Обучающемуся разрешается использование компьютера с установленными программными продуктами Web-проектирования.

7. Методические материалы по освоению дисциплины (модуля)

Обучение по дисциплине «Проектирование web-сервисов информационных систем» предполагает изучение курса на аудиторных занятиях (лекции, практические) и самостоятельной работы студентов. Практические занятия дисциплины «Проектирование web-сервисов информационных систем» предполагают их проведение в различных формах с целью выявления полученных знаний, умений, навыков и компетенций.

Подготовка к лекции студентами заключается в следующем:

– повторить материал предыдущей лекции, прочитав его повторно;

– ознакомиться с темой предстоящей лекции (в рабочей программе учебной дисциплины);

– ознакомиться с учебными материалами по данной теме в соответствии с предложенным списком литературы в рабочей программе учебной дисциплины или с электронными материалами, предложенными лектором;

– записать возможные вопросы, которые можно будет задать лектору.

Подготовка к практическим (семинарским) занятиям:

– внимательно прочитать материал лекций относящихся к данному занятию, ознакомиться с учебными материалами, включая электронные в

соответствии с предложенным списком литературы в рабочей программе учебной дисциплины;

– подготовить развернутые ответы на вопросы, предложенные в рабочей программе дисциплины для обсуждения;

– выполнить задания, если они предусмотрены в письменной форме;

– понять, что для вас осталось неясными и постараться получить на них ответ заранее;

– готовиться к практическим/семинарским занятиям можно как индивидуально, так и в составе малой группы;

– рабочую программу учебной дисциплины необходимо использовать в качестве основного ориентира в организации обучения;

– электронная версия рабочей программы по дисциплине размещена на сайте Нижегородского института управления, к ней предоставлен авторизованный доступ.

Самостоятельная работа обучающегося:

Самостоятельная работа магистрантов призвана закрепить теоретические знания и практические навыки, полученные магистрантами на лекциях, практических и семинарских занятиях.

Самостоятельная работа магистрантов проводится с целью:

– систематизации и закрепления полученных теоретических знаний и практических умений обучающихся;

– углубления и расширения теоретических знаний;

– формирования умений использовать справочную документацию и специальную литературу;

– развития познавательных способностей и активности магистрантов: творческой инициативы, самостоятельности, ответственности и организованности;

– формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;

– развития исследовательских умений.

Эффективность лекционных, семинарских и практических занятий по дисциплине во многом зависит от качества самостоятельной работы магистрантов, от их самоподготовки.

Часть времени, отведенного на самостоятельную работу должна использоваться на подготовку к аудиторным занятиям, другая часть на выполнение домашней работы, осмысление и оформление результатов практических занятий.

При подготовке к занятиям магистранту полезно:

– изучить теоретический материал по данной теме (конспект занятия);

– ознакомиться с литературой, рекомендованной преподавателем;

– выполнить задания, предложенные преподавателем, к занятию;

– составить перечень вопросов, вызывающих затруднения, неясности или сомнения, обсудить их с преподавателем или на занятии;

– заниматься самостоятельным поиском дополнительной литературы по изучаемой теме.

Подготовка к зачету. К зачету необходимо готовиться целенаправленно, регулярно, систематически и с первых дней обучения по данной дисциплине. В самом начале учебного курса познакомьтесь со следующей учебно-методической документацией:

– программой дисциплины;

– перечнем знаний и умений, которыми студент должен владеть;

– тематическими планами лекций, семинарских занятий;

– учебником, учебными пособиями по дисциплине, а также электронными ресурсами;

– перечнем и тематикой письменных работ, а также методическими рекомендациями по их выполнению;

– перечнем вопросов.

Систематическое выполнение всех видов заданий на лекциях, практических занятиях, а также самостоятельная работа позволит успешно освоить дисциплину и создать хорошую базу для сдачи зачета**.**

Вопросы для самостоятельной подготовки к занятиям

1. Что такое JAXB? Как аннотации JAXB можно применять в контексте SOAP-сервисов?

2. В чем ценность паттерна DTO применительно к веб-сервисам? Можно ли использовать этот паттерн в разработанном в соответствии с заданием веб-сервисе?

3. Предположим, что разработчик веб-сервиса не предоставил WSDL для разработки клиента к веб-сервису, а предоставил только XSD-схему. Достаточно ли XSD-схемы для разработки клиента к такому веб-сервису?

4. В чем заключается роль утилиты wsimport, и можно ли обойтись без нее?

5. В чем заключается роль утилиты wsgen, и можно ли обойтись без нее?

6. Модифицируйте REST и SOAP сервисы так, чтобы для CREATE, DELETE, UPDATE операций требовалась Basic-аутентификация. Внесите соответствующие изменения в клиентские приложения.

7. Предположим, что при реализации SOAP веб-сервиса необходимо передавать двоичные данные (аватары пользователей, архивы и др.). Какие стратегии можно предложить для работы с binary attachments? Реализуйте одну из них.

8. Обычно взаимодействие с внешним сервисом является дорогостоящей операцией и может привести к значительным задержкам. В таких случаях имеет смысл взаимодействовать с веб-сервисом асинхронно. Модифицируйте клиентское приложение таким образом, чтобы взаимодействие с веб-сервисом было асинхронным.

9. Предположим, что одним из нефункциональных требований к вашему сервису является поддержка throttling – умышленное ограничение количества одновременно выполняемых запросов, то есть если поступает новый запрос, а в это время уже выполняется максимально разрешенное количество, то необходимо прервать выполнение запроса, например, путем выброса ThrottlingException (класс исключения нужно создать самостоятельно). Реализуйте это в вашей работе.

10. Для чего нужны аннотации @Produces и @Consumes при реализации RESTсервиса?

11. В чем принципиальное различие при реализации обработки ошибок в REST и SOAP-сервисах с точки зрения разработки сервиса, а также с точки зрения реализации клиента?

12. Реализуйте endpoint, позволяющий пользователю загружать файлы (для простоты не нужно делать никаких проверок корректности имен файлов, наличия дубликатов файлов, и других подобных операций). Реализуйте клиентскую часть, взаимодействующую с сервисом.

13. Какие методы являются идемпотентными в отношении REST? В чем заключается свойство идемпотентности?

14. Для чего используются такие компоненты сервис-ориентированной архитектуры, как ESB? Можно ли обойтись без них?

15. В чем заключается смысл использования UDDI-реестров в промышленных SOA-системах? Можно ли обойтись без реестров?

16. В каком случае вы предпочтете сервис-ориентированную архитектуру вместо монолитной? Приведите ваши рассуждения по данному вопросу.

17. Предположим, что нужно спроектировать public API для предоставления программного доступа сторонним клиентам к ресурсам корпоративной информационной системы. Что вы выберете, REST или SOAP? Какие будут основные критерии выбора? Приведите ваши рассуждения по данному вопросу.

8. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет

8.1. Основная литература

1. Гвоздева, Т. В. Проектирование информационных систем: технология автоматизированного проектирования. Лабораторный практикум : учебное пособие / Т. В. Гвоздева, Б. А. Баллод. — 2-е изд., стер. — Санкт-Петербург : Лань, 2020. — 156 с. — ISBN 978-5-8114-5147-0. — Текст : электронный // Лань : электронно-библиотечная система. —

URL: https://e.lanbook.com/book/133477 (дата обращения: 12.04.2026). — Режим доступа: для авториз. пользователей.

2. Коваленко, В. В. Проектирование информационных систем : учебное пособие / В.В. Коваленко. — 2-е изд., перераб. и доп. — Москва : ИНФРА-М, 2023. — 357 с. — (Высшее образование: Бакалавриат). — DOI 10.12737/987869. - ISBN 978-5-00091-783-1. - Текст : электронный. - URL: https://znanium.com/catalog/product/1894610 (дата обращения: 12.04.2026). – Режим доступа: по подписке.

3. Козловский, П. Разработка веб-приложений с использованием AngularJS : практическое руководство / П. Козловский, П. Б. Дарвин ; пер. с англ. А. Н. Киселёва. - 2-е изд. - Москва : ДМК Пресс, 2023. - 395 с. - ISBN 978-5-89818-539-8. - Текст : электронный. - URL: https://znanium.com/catalog/product/2107216 (дата обращения: 07.06.2026). – Режим доступа: по подписке.

8.2. Дополнительная литература

1. Гумерова, Л. З. Основы web-программирования : учебное пособие / Л. З. Гумерова. — Красноярск : Научно-инновационный центр, 2019. — 104 c. — ISBN 978-5-6042232-6-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/97112.html (дата обращения: 02.03.2026). - Режим доступа: по подписке.

2. Меркулова А.Ш. Формирование баз данных [Электронный ресурс] : учебно-методическое пособие. — Электрон. дан. — Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=49651 Кемерово : КемГУКИ (Кемеровский государственный университет культуры и искусств), 2013. — 104 с.

3. Окулич, В.И. Разработка WEB-приложений : учебно-методическое пособие . Часть 1 : Логика функционирования Web приложений. Конструирование Web страниц с использованием языка HTML и каскадных таблиц стилей (CSS) / В.И. Окулич, Н.М. Трубилов. - Нижний Новгород : Издательство ВВАГС, 2011. - 184 с. - 125-71; 125-72. - Текст : непосредственный.

4. Окулич, В.И. Разработка WEB-приложений : учебно-методическое пособие . Часть 2 : Основы PHP - основного языка создания Web-приложений (введение в процедурное программирование на PHP) /

В.И. Окулич, Н.М. Трубилов. - Нижний Новгород : Издательство ВВАГС, 2011. - 298 с. - 680-00. - Текст : непосредственный.

5. Сергеенко, С. В. Разработка Web-приложений в Oracle Forms : учебное пособие / С. В. Сергеенко. — 2-е изд. — Москва : ИНТУИТ, 2016. — 221 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/100475 (дата обращения: 02.03.2026). — Режим доступа: для авториз. пользователей.

8.3. Нормативные правовые документы и иная правовая информация

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 №273-ФЗ. // Справочная правовая система КонсультантПлюс: [сайт]. URL: https://www.consultant.ru/document/cons_doc_LAW_140174/ (дата обращения: 14.04.2025).

2. Приказ Министерства науки и высшего образования Российской Федерации от 06.04.2021 №245 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры» // Справочная правовая система КонсультантПлюс: [сайт]. URL: https://www.consultant.ru/document/cons_doc_LAW_393023/ (дата обращения: 14.04.2025).

3. Приказ РАНХиГС Об утверждении положения о единой балльно-рейтинговой системе оценивания успеваемости студентов Академии и ее использовании при проведении текущей и промежуточной аттестации от 12.12.2024 №02-2531.

4. Федеральный государственный образовательный стандарт высшего образования - магистратура по направлению подготовки 09.04.03 Прикладная информатика, утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 №916 (зарегистрировано в Министерстве юстиции Российской Федерации 10.10.2017, регистрационный номер № 48495)

8.4 Интернет-ресурсы

1. Основы работы в OpenOffice. Web: http://www.intuit.ru/department/office/OpenOfficebasics.

2. Основы работы с Яндекс. Web: http://www.intuit.ru/department/internet/bwyandex/.

3. Современные офисные приложения. Web: http://www.intuit.ru/department/office/odpr/.

9. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

9.1. Материально-техническая база

Перечень материально-технического обеспечения:

1. Учебные аудитории, оборудованные для проведения учебных занятий лекционного и семинарского типа, групповых и индивидуальных консультаций, коллоквиумов, мероприятий текущего контроля и промежуточной аттестации, в том числе мультимедийным оборудованием для демонстрации электронных презентаций и аудио- и видеоматериалов.
2. Компьютерные классы для выполнения групповых тестовых и иных заданий, а также для самостоятельной работы обучающихся оснащенные компьютерной техникой и обеспечением доступа к сети «Интернет» и доступа в электронную информационно-образовательную среду организации.
3. Специализированные аудитории и лаборатории.
4. Библиотека с обеспечением печатными изданиями или электронно-библиотечная система обеспечивающая доступ к электронным изданиям (электронная библиотека).
5. Читальный зал.
6. Технические средства обучения: персональные компьютеры; компьютерные проекторы; звуковые динамики; программные средства, обеспечивающие просмотр видеофайлов в форматах AVI, MPEG-4, DivX, RMVB, WMV и др.

9.2. Информационные технологии, программное обеспечение:

При осуществлении образовательного процесса по дисциплине используется информационные технологии и программное обеспечение:

1. Современная операционная система.
2. Kaspersky Endpoint Security (или аналог).
3. Средство просмотра файлов формата pdf.
4. Современные офисные средства (текстовые и табличные редакторы, средства работы с презентационными материалами и т.д.).
5. Архиватор 7-Zip.
6. Система дистанционного обучения.
7. Автоматизированная библиотечная система.

9.3. Информационные справочные системы:

1. https://www.iprbookshop.ru – Электронно-библиотечная система [ЭБС]

«IPRSMART» (ранее – IPRBooks)

2. https://e.lanbook.com - Электронно-библиотечная система [ЭБС] «Лань».
3. Электронно-библиотечная система [ЭБС] «Znanium.com».
4. https://www.book.ru - Электронно-библиотечная система [ЭБС] «Book.ru».
5. https://ibooks.ru - Электронно-библиотечная система [ЭБС] «ibooks.ru».
6. https://ranepalib.miflib.ru - Электронная библиотека издательства «МИФ».
7. https://eivis.ru – Ивис. Полные тексты российских научных и практических журналов, а также газет центральной прессы России. Доступ с ip-адресов локальной сети Института.
8. http://www.consultant.ru/ - Справочно-правовая система «Консультант».
9. http://www.garant.ru/ Справочно-правовая система «Гарант».