

Министерство образования и науки Астраханской области
Государственное автономное образовательное учреждение
Астраханской области высшего образования
«Астраханский государственный архитектурно-строительный
университет»
(ГАОУ АО ВО «АГАСУ»)



РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Наименование дисциплины

Перспективные Web-технологии

(указывается наименование в соответствии с учебным планом)

По направлению подготовки

09.04.02 «Информационные системы и технологии»

(указывается наименование направления подготовки в соответствии с ФГОС)

Направленность (профиль)

«Информационные системы и технологии в строительстве и архитектуре»

(указывается наименование профиля в соответствии с ОПОП)

Кафедра

Системы автоматизированного проектирования и моделирования

Квалификация выпускника *магистр*

Астрахань – 2019

Разработчик:

К. П. И. Гоцелет
(занимаемая должность,
учёная степень и учёное звание)

[подпись]
(подпись)

Л. Б. Мещеряков
И.О.Ф.

Рабочая программа рассмотрена и утверждена на заседании кафедры «Системы автоматизированного проектирования и моделирования»

протокол № 10 от 15.05 2019г.

Заведующий кафедрой

[подпись]
(подпись)

Т. В. Хоменко /
И.О.Ф.

Согласовано:

Председатель МКН «Информационные системы и технологии»
направленность (профиль) «Информационные системы и технологии в строительстве и архитектуре»

[подпись] / Т. В. Хоменко
(подпись) И.О.Ф.

Начальник УМУ [подпись] / М. В. Александров
(подпись) И. О. Ф

Специалист УМУ [подпись] / Л. В. Дудирова
(подпись) И. О. Ф

Начальник УИТ [подпись] / С. В. Трунов
(подпись) И. О. Ф

Заведующая научной библиотекой [подпись] / В. С. Кайракирова
(подпись) И.О. Ф

Содержание

| | |
|--|----|
| 1. Цель освоения дисциплины | 4 |
| 2. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы | 4 |
| 3. Место дисциплины в структуре ОПОП магистратуры | 4 |
| 4. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по типам занятий) и на самостоятельную работу обучающихся | 4 |
| 5. Содержание дисциплины, структурированное по разделам с указанием отведенного на них количества академических часов и типов учебных занятий | 6 |
| 5.1. Разделы дисциплины и трудоемкость по типам учебных занятий и работы обучающихся (в академических часах) | 6 |
| 5.1.1. Очная форма обучения | 6 |
| 5.1.2. Заочная форма обучения | 7 |
| 5.2. Содержание дисциплины, структурированное по разделам | 8 |
| 5.2.1. Содержание лекционных занятий | 8 |
| 5.2.2. Содержание лабораторных занятий | 8 |
| 5.2.3. Содержание практических занятий | 9 |
| 5.2.4. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине | 9 |
| 5.2.5. Темы контрольных работ | 10 |
| 5.2.6. Темы курсовых проектов/ курсовых работ | 10 |
| 6. Методические указания для обучающихся по освоению дисциплины | 11 |
| 7. Образовательные технологии | 11 |
| 8. Учебно-методическое и информационное обеспечение дисциплины | 12 |
| 8.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины | 12 |
| 8.2. Перечень необходимого лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, используемого при осуществлении образовательного процесса по дисциплине | 13 |
| 8.3. Перечень современных профессиональных баз данных и информационных справочных систем, доступных обучающимся при освоении дисциплины | 13 |
| 9. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине | 14 |
| 10. Особенности организации обучения по дисциплине для инвалидов и лиц с ограниченными возможностями здоровья | 14 |

1. Цель освоения дисциплины

Целью освоения дисциплины «Перспективные Web-технологии» является углубление уровня освоения компетенций обучающихся в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.04.02 «Информационные системы и технологии»

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

В результате освоения дисциплины обучающийся должен овладеть следующими компетенциями:

ПК-3 – способен распределять задания по выполнению разработки программного обеспечения, осуществлять общее руководство и контроль выполнения заданий;

ПК-9 – способен разбираться в работе системного программного обеспечения, дописывать фрагменты и производить отладку системного программного обеспечения.

В результате освоения дисциплин, формирующих компетенции ПК-3, ПК-9, обучающийся должен овладеть следующими результатами обучения:

знать:

- методологии управления проектами разработки программного обеспечения (ПК-3.1);
- стандарты системной и программной инженерии (ПК-9.1);

уметь:

- применять методологии управления проектами разработки программного обеспечения (ПК-3.2);
- описывать цели проекта и критерии успешности их достижения (ПК-9.2);

иметь практический опыт:

- принятия управленческих решений по выбору средств создания, учета задач, сборки и базы знаний (ПК-3.3);
- подготовка документации по разработке системного программного обеспечения (ПК-9.3).

3. Место дисциплины в структуре ОПОП магистратуры

Дисциплина Б1.В.ДВ.01.01 «Перспективные Web-технологии» реализуется в рамках Блока 1 «Дисциплины, (модули)», части, формируемой участниками образовательных отношений, (элективные дисциплины (по выбору)).

Дисциплина базируется на знаниях, полученных в рамках изучения следующей дисциплины:

- 1) Системы управления базами данных
- 2) Программирование и разработка программного обеспечения

4. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по типам занятий) и на самостоятельную работу обучающихся

| Форма обучения | Очная | Заочная |
|----------------|-------|---------|
| 1 | 2 | 3 |

| | | |
|-----------------------------------|---|---|
| Трудоемкость в зачетных единицах: | 4 семестр – 2 з.е.; всего – 2 з.е. | 4 семестр – 2 з.е.; всего - 2 з.е. |
| Лекции (Л) | 4 семестр – 12 часов; всего – 12 часов | 4 семестр – 4 часа; всего - 4 часа |
| Лабораторные занятия (ЛЗ) | 4 семестр – 24 часа; всего - 24 часа | 4 семестр – 8 часов; всего - 8 часов |
| Практические занятия (ПЗ) | учебным планом не предусмотрены | учебным планом не предусмотрены |
| Самостоятельная работа (СР) | 4 семестр – 36 часов; всего – 36 часов | 4 семестр – 60 часов; всего - 60 часов |
| Форма текущего контроля: | | |
| Контрольная работа | учебным планом не предусмотрена | учебным планом не предусмотрена |
| Форма промежуточной аттестации: | | |
| Зачет | семестр - 4 | семестр - 4 |
| Экзамен | учебным планом не предусмотрен | учебным планом не предусмотрен |
| Зачет с оценкой | учебным планом не предусмотрен | учебным планом не предусмотрен |
| Курсовая работа | учебным планом не предусмотрена | учебным планом не предусмотрена |
| Курсовой проект | учебным планом не предусмотрен | учебным планом не предусмотрен |

5. Содержание дисциплины, структурированное по разделам с указанием отведенного на них количества академических часов и типов учебных занятий

5.1. Разделы дисциплины и трудоемкость по типам учебных занятий и работы обучающихся (в академических часах)

5.1.1. Очная форма обучения

| № п/п | Раздел дисциплины (по семестрам) | Всего часов на раздел | Семестр | Распределение трудоемкости раздела (в часах) по типам учебных занятий и работы обучающихся | | | | Форма текущего контроля и промежуточной аттестации |
|---------------|--|-----------------------|---------|--|-----------|----------|-----------|--|
| | | | | контактная | | | СР | |
| | | | | Л | ЛЗ | ПЗ | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | Раздел 1. Введение в ASP .NET MVC | 11 | 4 | 2 | 4 | - | 5 | Зачет |
| 2 | Раздел 2. Контроллеры. Представления. Модели | 16 | | 2 | 4 | - | 10 | |
| 3 | Раздел 3. Маршрутизация. Метаданные и валидация модели | 11 | | 2 | 4 | - | 5 | |
| 4 | Раздел 4. Фильтры. Привязка модели. | 11 | | 2 | 4 | - | 5 | |
| 5 | Раздел 5. JavaScript. AJAX. Bootstrap. | 12 | | 2 | 4 | - | 6 | |
| 6 | Раздел 6. Авторизация и аутентификация. Тестирование приложения. | 11 | | 2 | 4 | - | 5 | |
| Итого: | | 72 | | 12 | 24 | - | 36 | |

5.1.2. Заочная форма обучения

| № п/п | Раздел дисциплины (по семестрам) | Всего часов на раздел | Семестр | Распределение трудоемкости раздела (в часах) по типам учебных занятий и ра- боты обучающихся | | | | Форма теку- щего контроля и промежу- точной атте- стации |
|---------------|---|--------------------------|---------|--|----------|----------|-----------|--|
| | | | | контактная | | | СР | |
| | | | | Л | ЛЗ | ПЗ | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | Раздел 1. Введение в ASP .NET MVC | 11 | 4 | | 1 | - | 10 | Зачет |
| 2 | Раздел 2. Контроллеры. Представления. Модели | 13 | | 1 | 2 | - | 10 | |
| 3 | Раздел 3. Маршрутизация. Метаданные и валидация модели | 12 | | 1 | 1 | - | 10 | |
| 4 | Раздел 4. Фильтры. Привязка модели. | 13 | | 1 | 2 | - | 10 | |
| 5 | Раздел 5. JavaScript. AJAX. Bootstrap. | 12 | | 1 | 1 | - | 10 | |
| 6 | Раздел 6. Авторизация и аутентификация. Тестирование приложения. | 11 | | | | 1 | - | |
| Итого: | | 72 | | 4 | 8 | - | 60 | |

5.2. Содержание дисциплины, структурированное по разделам

5.2.1. Содержание лекционных занятий

| № | Наименование раздела дисциплины | Содержание |
|----|--|---|
| 1. | Раздел 1. Введение в ASP .NET MVC | Особенности ASP .NET. Начало работы. Создание и управление проектом по разработке программного обеспечения (веб-приложения) в соответствии с современной методологией. Создание контроллера и представлений. Использование ASP .NET в разработке современного программного обеспечения. |
| 2. | Раздел 2. Контроллеры. Представления. Модели | Основы контроллеров. Методы действий и их параметры. Результаты действий. Переадресация и отправка кодов статуса и ошибок. Отправка файлов. Куки. Сессии. Введения в представления. Мастер-страницы. Частичные представления. HTML-хелперы. Работа с формами. Строго типизированные хелперы. Модели и БД. Шаблонные хелперы. Редактирование, добавление и удаление данных. Работа со сложной структурой и моделями в соответствии со стандартами программной инженерии. |
| 3. | Раздел 3. Маршрутизация. Метаданные и валидация модели | Определение маршрутов. Работа с маршрутами. Создание ограничений для маршрутов. Генерация исходящих адресов. Создание собственного обработчика маршрутов. Аннотации данных для отображения свойств. Основы валидации. Атрибуты валидации. Валидация модели в контроллере. Отображение ошибок валидации. |
| 4. | Раздел 4. Фильтры. Привязка модели. | Фильтры. Фильтры аутентификации. Фильтры авторизации. Фильтры исключений. Примеры фильтров действий. Дополнительные встроенные фильтры. Введение в привязку модели. Явная привязка модели. Поставщики значений. Создание привязчика модели. |
| 5 | Раздел 5. JavaScript. AJAX. Bootstrap. | Использование JavaScript. Введение в AJAX. AJAX-формы. AJAX-ссылки. Форма JSON. Введение в Bootstrap. Позиционирование элементов. Компоненты Bootstrap. Работа с формами в Bootstrap. Работа с таблицами. Модальные окна. |
| 6 | Раздел 6. Авторизация и аутентификация. Тестирование приложения. | ASP .NET Identity. Авторизация через внешние сервисы. Стандарты системной инженерии. Работа с ролями в ASPNet Identity. Настройка валидации пароля. Валидация пользователя. Введение в тестирование. Создание проекта для юнит-тестов. Создание юнит-теста. |

5.2.2. Содержание лабораторных занятий

| № | Наименование раздела дисциплины | Содержание |
|---|---------------------------------|------------|
|---|---------------------------------|------------|

| | | |
|----|--|---|
| 1. | Раздел 1. Введение в ASP .NET MVC | Разработка первого приложения ASP .NET, описание целей проекта, определение критерий успешности его достижения |
| 2. | Раздел 2. Контроллеры. Представления. Модели | Разработка веб-приложения «Сотрудники предприятия» в соответствии с методологией управления проектами по разработке программного обеспечения |
| 3. | Раздел 3. Маршрутизация. Метаданные и валидация модели | Выбор средств создания, сборки, базы знаний. Разработка почтового приложения. |
| 4. | Раздел 4. Фильтры. Привязка модели. | Передача данных HTML-формы на ASPX-страницу методами класса Request. Чтение и запись cookie-файлов |
| 5 | Раздел 5. JavaScript. AJAX. Bootstrap. | Разработка простейшей Web-службы и приложения – потребителя сервиса в соответствии с принятыми управленческими решениями по выбору средств создания, учета задач, сборки и базы знаний. |
| 6 | Раздел 6. Авторизация и аутентификация. Тестирование приложения. | Подготовка документации по разработке. Разработка веб-приложения по индивидуальному заданию |

5.2.3. Содержание практических занятий

Учебным планом не предусмотрены

5.2.4. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Очная форма

| № | Наименование раздела дисциплины | Содержание | Учебно-методическое обеспечение |
|----|--|--|---------------------------------|
| 1 | 2 | 3 | 4 |
| 1. | Раздел 1. Введение в ASP .NET MVC | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1], [2] |
| 2. | Раздел 2. Контроллеры. Представления. Модели | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1]-[3] |
| 3. | Раздел 3. Маршрутизация. Метаданные и валидация модели | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1], [4] |
| 4. | Раздел 4. Фильтры. Привязка модели. | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1], [5] |
| 5. | Раздел 5. JavaScript. AJAX. Bootstrap. | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1] - [4] |

| | | | |
|----|--|--|----------------|
| 6. | Раздел 6. Авторизация и аутентификация. Тестирование приложения. | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1] - [3], [6] |
|----|--|--|----------------|

Заочная форма

| № | Наименование раздела дисциплины | Содержание | Учебно-методическое обеспечение |
|----|--|--|---------------------------------|
| 1 | 2 | 3 | 4 |
| 1. | Раздел 1. Введение в ASP .NET MVC | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1], [2] |
| 2. | Раздел 2. Контроллеры. Представления. Модели | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1]-[3] |
| 3. | Раздел 3. Маршрутизация. Метаданные и валидация модели | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1], [4] |
| 4. | Раздел 4. Фильтры. Привязка модели. | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1], [5] |
| 5. | Раздел 5. JavaScript. AJAX. Bootstrap. | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1] - [4] |
| 6. | Раздел 6. Авторизация и аутентификация. Тестирование приложения. | Подготовка к лабораторной работе Подготовка к тестированию Подготовка к зачету | [1] - [3], [6] |

5.2.5. Темы контрольных работ

Учебным планом не предусмотрены

5.2.6. Темы курсовых проектов/ курсовых работ

Учебным планом не предусмотрены

6. Методические указания для обучающихся по освоению дисциплины

| Организация деятельности студента |
|---|
| <p>Лекция</p> <p>В ходе лекционных занятий необходимо вести конспектирование учебного материала, обращать внимание на категории, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Необходимо задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений, разрешения спорных ситуаций. Целесообразно дорабатывать свой конспект лекции, делая в нем соответствующие записи из литературы, рекомендованной преподавателем и предусмотренной учебной программой</p> |
| <p>Лабораторное занятие</p> <p>Работа в соответствии с методическими указания по выполнению лабораторных работ</p> |
| <p>Самостоятельная работа</p> <p>Самостоятельная работа студента над усвоением учебного материала по учебной дисциплине может выполняться в помещениях для самостоятельной работы, а также в домашних условиях. Содержание самостоятельной работы студента определяется учебной программой дисциплины, методическими материалами, заданиями и указаниями преподавателя.</p> <p>Самостоятельная работа в аудиторное время может включать:</p> <ul style="list-style-type: none">– конспектирование (составление тезисов) лекций;– работу со справочной и методической литературой;– выполнение заданий лабораторной работы;– участие в тестировании. <p>Самостоятельная работа во внеаудиторное время может состоять из:</p> <ul style="list-style-type: none">– повторения лекционного материала;– подготовки к лабораторным занятиям;– изучения учебной и научной литературы;– подготовка к тестированию. |
| <p>Подготовка к зачету</p> <p>Подготовка студентов к зачету включает три стадии:</p> <ul style="list-style-type: none">– самостоятельная работа в течение семестра;– непосредственная подготовка в дни, предшествующие зачету;– подготовка к ответу на вопросы, содержащиеся в перечне вопросов. |

7. Образовательные технологии

Перечень образовательных технологий, используемых при изучении дисциплины «Перспективные Web-технологии».

Традиционные образовательные технологии

Дисциплина «Перспективные Web-технологии» проводится с использованием традиционных образовательных технологий, ориентирующихся на организацию образовательного процесса, предполагающую прямую трансляцию знаний от преподавателя к студенту (преимущественно на основе объяснительно-иллюстративных методов обучения), учебная деятельность студента носит в таких условиях, как правило, репродуктивный характер. Формы учебных занятий с использованием традиционных технологий:

Лекция – последовательное изложение материала в дисциплинарной логике,

осуществляемое преимущественно вербальными средствами (монолог преподавателя).

Лабораторное занятие – организация учебной работы с реальными материальными и информационными объектами, экспериментальная работа с моделями реальных объектов.

Интерактивные технологии

По дисциплине «Перспективные Web-технологии» лекционные занятия проводятся с использованием следующих интерактивных технологий:

Лекция-визуализация – представляет собой визуальную форму подачи лекционного материала средствами ТСО или аудиовидеотехники (видео-лекция). Чтение такой лекции сводится к развернутому или краткому комментированию просматриваемых визуальных материалов (в виде схем, таблиц, графов, графиков, моделей). Лекция-визуализация помогает студентам преобразовывать лекционный материал в визуальную форму, что способствует формированию у них профессионального мышления за счет систематизации и выделения наиболее значимых, существенных элементов.

По дисциплине «Перспективные Web-технологии» лабораторные занятия проводятся с использованием следующих интерактивных технологий:

Работа в малых группах – это одна из самых популярных стратегий, так как она дает всем обучающимся (в том числе и стеснительным) возможность участвовать в работе, практиковать навыки сотрудничества, межличностного общения (в частности, умение активно слушать, вырабатывать общее мнение, разрешать возникающие разногласия). Все это часто бывает невозможно в большом коллективе.

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

а) основная учебная литература:

1. Колисниченко, Д.Н. PHP и MySQL. Разработка Web-приложений / Д.Н. Колисниченко. – Санкт-Петербург: «БХВ-Петербург». – 2017. – 640с. – ISBN 978-5-9775-3835-0.

2. Тукеев, У.А. Программирование Web-приложений информационных систем / У.А. Тукеев, Ж.М. Жуманов. – Алматы: Издательство «Казахский национальный университет им. аль-Фараби». – 2012. – 102с. – ISBN 978-601-247-399-5. – [Электронный ресурс] Режим доступа: <http://www.iprbookshop.ru/61206.html>

3. Маркин, А.В. Основы Web-программирования на PHP: учебное пособие / А.В. Маркин, С.С. Шкарин. – Москва: «Диалог-МИФИ». – 2012. – 252с. – ISBN 978-5-86404-241-0. – [Электронный ресурс] Режим доступа: <http://biblioclub.ru/index.php?page=book&id=229742>

б) дополнительная учебная литература:

4. Торопова, О.А. Основы Web-программирования. Технологии HTML, DHTML: учебное пособие / О.А. Торопова, И.Ф. Сытник. – Саратов: Издательство «Саратовский государственный технический университет имени Ю.А. Гагарина». – 2012. – 106с. – ISBN 978-5-7433-2606-8. – [Электронный ресурс] Режим доступа: <http://www.iprbookshop.ru/76493.html>

5. Громов, Ю.Ю. Основы Web-инжиниринга: разработка клиентских приложений: учебное пособие / Ю.Ю. Громов, О.Г. Иванова, С.В. Данилкин. – Тамбов: Издательство ФГБОУ ВПО «Тамбовский государственный технический университет». – 2012. – 240с. – [Электронный ресурс] Режим доступа: <http://biblioclub.ru/index.php?page=book&id=277648>

6. Столбовский, Д.Н. Основы разработки Web-приложений на ASP.NET: учебное пособие / Д.Н. Столбовский. – Москва: Издательство Национальный Открытый Универ-

ситет «Интернет-Университет Информационных Технологий (ИНТУИТ)». – 2009. – 304с. – ISBN 978-5-94774-991-5. – [Электронный ресурс] Режим доступа: <http://biblioclub.ru/index.php?page=book&id=233488>

в) перечень учебно-методического обеспечения:

1. Аминул, Л.Б. Методические указания к выполнению лабораторных работ по дисциплине «Перспективные Web-технологии» / Л.Б. Аминул. – Астрахань: АГАСУ. – 2019 г. – 16с.

<http://moodle.aucu.ru>

2. Аминул, Л.Б. Методические указания по выполнению самостоятельной работы по дисциплине «Перспективные Web-технологии» / Л.Б. Аминул. – Астрахань: АГАСУ. – 2019 г. – 16с.

<http://moodle.aucu.ru>

г) перечень онлайн курсов:

1. Руководство по ASP .NET MVC 5. Электронный ресурс. URL: <https://metanit.com/sharp/mvc5/>

2. Руководство по ASP .NET Core 3. Электронный ресурс. URL: <https://metanit.com/sharp/aspnet5/>

8.2. Перечень необходимого лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, используемого при осуществлении образовательного процесса по дисциплине

1. 7-Zip
2. Office 365 A1
3. Adobe Acrobat Reader DC
4. Google Chrome
5. VLC media player
6. Apache Open Office
7. Office Pro Plus Russian OLPNL Academic Edition
8. Kaspersky Endpoint Security
9. Internet Explorer
10. Microsoft SQL Server 2016 Express
11. Visual Studio
12. Microsoft Azure Dev Tools for Teaching
13. Blender
14. Mamp pro

8.3. Перечень современных профессиональных баз данных и информационных справочных систем, доступных обучающимся при освоении дисциплины

1. Электронная информационно-образовательная среда Университета: образовательный портал (<http://moodle.aucu.ru>)
2. Электронно-библиотечная система «Университетская библиотека» (<https://biblioclub.ru/>)
3. Электронно-библиотечная система «IPRbooks» (www.iprbookshop.ru)

4. Научная электронная библиотека (<http://www.elibrary.ru/>)
5. Консультант + (<http://www.consultant-urist.ru/>)
6. Федеральный институт промышленной собственности (<https://www1.fips.ru/>)
7. Патентная база USPTO (<https://www.uspto.gov/patents-application-process/search-patents>)

9. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

| № п\п | Наименование специальных помещений и помещений для самостоятельной работы | Оснащенность специальных помещений и помещений для самостоятельной работы |
|-------|--|---|
| 1 | Учебная аудитория для проведения учебных занятий 414056, г. Астрахань, ул. Татищева, 18, аудитория №207 | аудитория №207 Комплект учебной мебели Компьютеры – 15 шт. Стационарный мультимедийный комплект Доступ к информационно – телекоммуникационной сети «Интернет» |
| | 414056, г. Астрахань, ул. Татищева, 18, аудитория №209 | аудитория №209 Комплект учебной мебели Компьютеры – 15 шт. Стационарный мультимедийный комплект Доступ к информационно – телекоммуникационной сети «Интернет» |
| | 414056, г. Астрахань, ул. Татищева, 18, аудитория №211 | аудитория №211 Комплект учебной мебели Компьютеры – 15 шт. Стационарный мультимедийный комплект Доступ к информационно – телекоммуникационной сети «Интернет» |
| 2 | Помещение для самостоятельной работы 414056, г. Астрахань, ул. Татищева, 18, аудитория №201 | аудитория №201 Комплект учебной мебели Компьютеры – 4 шт. Доступ к информационно – телекоммуникационной сети «Интернет» |
| | 414056, г. Астрахань, ул. Татищева, 186, аудитория №308 | аудитория №308 Комплект учебной мебели Компьютеры – 11 шт. Доступ к информационно – телекоммуникационной сети «Интернет» |

10. Особенности организации обучения по дисциплине для инвалидов и лиц с ограниченными возможностями здоровья

Для обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья на основании письменного заявления дисциплина «Перспективные Web-технологии» реализуется с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья (далее – индивидуальных особенностей)

**Лист внесения дополнений и изменений
в рабочую программу учебной дисциплины**

«Перспективные Web-технологии»

(наименование дисциплины)

на 20__ - 20__ учебный год

Рабочая программа пересмотрена на заседании кафедры *«Системы автоматизированного проектирования и моделирования»*

протокол № ____ от _____ 20__ г.

Зав. кафедрой

ученая степень, ученое звание

подпись

/ _____ /

И.О. Фамилия

В рабочую программу вносятся следующие изменения:

1. _____
2. _____
3. _____
4. _____
5. _____

Составители изменений и дополнений:

ученая степень, ученое звание

подпись

/ _____ /

И.О. Фамилия

ученая степень, ученое звание

подпись

/ _____ /

И.О. Фамилия

Председатель МКН направления подготовки «Информационные системы и технологии»
профиля подготовки «Информационные системы и технологии в строительстве и архитектуре»

ученая степень, ученое звание

подпись

/ _____ /

И.О. Фамилия

« ____ » _____ 20__ г.

Министерство образования и науки Астраханской области
Государственное автономное образовательное учреждение
Астраханской области высшего образования
«Астраханский государственный архитектурно-строительный
университет»
(ГАОУ АО ВО «АГАСУ»)



ОЦЕНОЧНЫЕ И МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Наименование дисциплины

Перспективные Web-технологии

(указывается наименование в соответствии с учебным планом)

По направлению подготовки

09.04.02 «Информационные системы и технологии»

(указывается наименование направления подготовки в соответствии с ФГОС)

Направленность (профиль)

«Информационные системы и технологии в строительстве и архитектуре»

(указывается наименование профиля в соответствии с ОПОП)

Кафедра Системы автоматизированного проектирования и моделирования

Квалификация выпускника *магистр*

Астрахань – 2019

Разработчик:

Е.И.И. доцент
(занимаемая должность,
учёная степень и учёное звание)

[подпись]
(подпись)

А.Б. Акимов
И.О.Ф.

Оценочные и методические материалы рассмотрены и утверждены на заседании кафедры
«Системы автоматизированного проектирования и моделирования»

протокол №10 от 25.05.2019г.

Заведующий кафедрой

[подпись]
(подпись)

/ Т.В. Хоменко /
И.О.Ф.

Согласовано:

Председатель МКН «Информационные системы и технологии»
направленность (профиль) подготовки «Информационные системы и технологии в
строительстве и архитектуре»

[подпись]
(подпись)

/ Т.В. Хоменко /

И.О.Ф.

Начальник УМУ

[подпись] / Н.В. Акимов
(подпись) И. О. Ф.

Специалист УМУ

[подпись] / Д.А. Бурдасова
(подпись) И. О. Ф.

Содержание

| | |
|--|----|
| 1. Оценочные и методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине | 4 |
| 1.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы | 4 |
| 1.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания | 5 |
| 1.2.1. Перечень оценочных средств текущего контроля успеваемости | 5 |
| 1.2.2. Описание показателей и критериев оценивания компетенций по дисциплине на различных этапах их формирования, описание шкал оценивания..... | 6 |
| 1.2.3. Шкала оценивания..... | 7 |
| 2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков, характеризующих этапы формирования компетенций в процессе освоения образовательной программы..... | 9 |
| 3. Перечень и характеристики процедуры оценивания знаний, умений, навыков, характеризующих этапы формирования компетенций..... | 11 |
| Приложение 1..... | 12 |
| Приложение 2..... | 13 |

1. Оценочные и методические материалы для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине

Оценочные и методические материалы являются неотъемлемой частью рабочей программы дисциплины (далее РПД) и представлены в виде отдельного документа

1.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

| Индекс и формулировка компетенции N | Индикаторы достижений компетенций, установленные ОПОП | Номер раздела дисциплины (в соответствии с п.5.1РПД) | | | | | | Формы контроля с конкретизацией задания |
|--|---|--|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| ПК-3 – способен распределять задания по выполнению разработки программного обеспечения, осуществлять общее руководство и контроль выполнения заданий | Знать: | | | | | | | Зачет, вопросы 1-10 Защита лабораторной работы № 2,3,5 тестирование вопросы 1-7 |
| | методологии управления проектами разработки программного обеспечения | | | X | | X | X | |
| | Уметь: | | | | | | | |
| | применять методологии управления проектами разработки программного обеспечения | | X | X | | X | X | |
| ПК-9 – способен разбираться в работе системного программного обеспечения, дописывать фрагменты и производить отладку системного программного обеспечения | Иметь практический опыт: | | | | | | | Зачет, вопросы 11-25 Защита лабораторной работы № 1,4,6 тестирование вопросы 8-20 |
| | принятия управленческих решений по выбору средств создания, учета задач, сборки и базы знаний | X | X | X | X | | | |
| | Знать: | | | | | | | |
| | стандарты системной и программной инженерии | X | | | | X | | |
| | Уметь: | | | | | | | |
| | описывать цели проекта и критерии успешности их достижения | X | X | X | X | X | | |
| | Иметь практический опыт: | | | | | | | |
| | подготовка документации по разработке системного программного обеспечения | | X | X | X | | X | |

1.2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

1.2.1. Перечень оценочных средств текущего контроля успеваемости

| Наименование оценочного средства | Краткая характеристика оценочного средства | Представление оценочного средства в фонде |
|----------------------------------|---|--|
| Защита лабораторной работы | Средство, позволяющее оценить умение и владение обучающегося излагать суть поставленной задачи, самостоятельно применять стандартные методы решения поставленной задачи с использованием имеющейся лабораторной базы, проводить анализ полученного результата работы. Рекомендуется для оценки умений и владений студентов. | Темы лабораторных работ и требования к их защите |
| Тест | Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося | Фонд тестовых заданий |

1.2.2. Описание показателей и критериев оценивания компетенций по дисциплине на различных этапах их формирования, описание шкал оценивания

| Компетенция, этапы освоения компетенции | Планируемые результаты обучения | Показатели и критерии оценивания результатов обучения | | | |
|--|--|---|---|---|---|
| | | Ниже порогового уровня (незачтено) | Пороговый уровень (зачтено) | Продвинутый уровень (зачтено) | Высокий уровень (зачтено) |
| 1 | 2 | 3 | 4 | 5 | 6 |
| ПК-3 – способен распределять задания по выполнению разработки программного обеспечения, осуществлять общее руководство и контроль выполнения заданий | Знает: методологии управления проектами разработки программного обеспечения | Обучающийся не знает методы принятия управленческих решений | Обучающийся имеет знания только основного материала, но не усвоил его деталей, допускает неточности | Обучающийся знает методы принятия решений, основные принципы и методы управления персоналом | Обучающийся знает современные методы принятия управленческих решений и методы управления персоналом при управлении проектами |
| | Умеет: применять методологии управления проектами разработки программного обеспечения | Обучающийся не умеет применять методы принятия управленческих решений и нормативно-технические документы (стандарты и регламенты) | Обучающийся умеет применять методы принятия управленческих решений только под руководством опытных специалистов | Обучающийся умеет применять методы принятия управленческих решений самостоятельно, получая консультации специалистов в способен контролировать выполнение заданий | Обучающийся умеет самостоятельно применять методы принятия управленческих решений, способен адекватно распределять задания по выполнению разработки программного обеспечения. |
| | Иметь практический опыт: принятия управленческих решений по выбору средств создания, учета задач, сборки и базы знаний | Обучающийся не имеет опыта принятия управленческих решений по результатам проектирования программного обеспечения | Обучающийся имеет опыт работы с опытом принятия управленческих решений по результатам проектирования программного обеспечения под руководством опытных специалистов | Обучающийся имеет опыт принятия управленческих решений по результатам проектирования программного обеспечения в команде | Обучающийся имеет опыт принятия эффективных управленческих решений по результатам проектирования программного обеспечения. |

| | | | | | |
|--|--|--|--|--|---|
| ПК-9 – способен разбираться в работе системного программного обеспечения, дописывать фрагменты и производить отладку системного программного обеспечения | Знает: стандарты системной и программной инженерии | Обучающийся не знает и не понимает особенности программирования и стандартные библиотеки выбранного языка | Обучающийся не твердо знает и не вполне понимает особенности программирования и стандартные библиотеки выбранного языка | Обучающийся знает и понимает особенности программирования и стандартные библиотеки выбранного языка, методы и алгоритмы оптимизации исполняемого кода. | Обучающийся знает и детально понимает особенности программирования и стандартные библиотеки выбранного языка, методы и алгоритмы оптимизации исполняемого кода. |
| | Умеет: описывать цели проекта и критерии успешности их достижения | Обучающийся не умеет описывать цели проекта и критерии успешности их достижения | Обучающийся умеет описывать цели проекта только под руководством специалистов | Обучающийся умеет решать стандартные профессиональные задачи, в том числе самостоятельно описывать цели проекта | Обучающийся умеет решать стандартные профессиональные, в том числе самостоятельно описывать цели проекта и критерии успешности его достижения |
| | Иметь практический опыт: подготовка документации по разработке системного программного обеспечения | Обучающийся не имеет навыки сопровождения программного обеспечения инструментальных средств программирования | Обучающийся имеет навыки сопровождения программного обеспечения инструментальных средств программирования только под руководством специалистов | Обучающийся имеет навыки самостоятельного сопровождения программного обеспечения инструментальных средств программирования | Обучающийся имеет навыки самостоятельного и профессионального сопровождения программного обеспечения современных инструментальных средств программирования |

1.2.3. Шкала оценивания

| Уровень достижений | Отметка в 5-бальной шкале | Зачтено/ не зачтено |
|--------------------|---------------------------|---------------------|
| высокий | «5»(отлично) | зачтено |
| продвинутый | «4»(хорошо) | зачтено |
| пороговый | «3»(удовлетворительно) | зачтено |

| | | |
|-----------------|--------------------------|------------|
| ниже порогового | «2»(неудовлетворительно) | не зачтено |
|-----------------|--------------------------|------------|

2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

ТИПОВЫЕ ЗАДАНИЯ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

2.1. Зачет

- а) типовые вопросы к зачету (Приложение 1)
- б) критерии оценки

При оценке знаний на зачете учитывается:

1. Уровень сформированности компетенций.
2. Уровень усвоения теоретических положений дисциплины, правильность формулировки основных понятий и закономерностей.
3. Уровень знания фактического материала в объеме программы.
4. Логика, структура и грамотность изложения вопроса.
5. Умение связать теорию с практикой.
6. Умение делать обобщения, выводы.

| № п/п | Оценка | Критерии оценки |
|-------|---------------------|--|
| 1 | Отлично | Ответы на поставленные вопросы излагаются логично, последовательно и не требуют дополнительных пояснений. Полно раскрываются причинно-следственные связи между явлениями и событиями. Делаются обоснованные выводы. Демонстрируются глубокие знания базовых нормативно-правовых актов. Соблюдаются нормы литературной речи |
| 2 | Хорошо | Ответы на поставленные вопросы излагаются систематизировано и последовательно. Базовые нормативно-правовые акты используются, но в недостаточном объеме. Материал излагается уверенно. Раскрыты причинно-следственные связи между явлениями и событиями. Демонстрируется умение анализировать материал, однако не все выводы носят аргументированный и доказательный характер. Соблюдаются нормы литературной речи |
| 3 | Удовлетворительно | Допускаются нарушения в последовательности изложения. Имеются упоминания об отдельных базовых нормативно-правовых актах. Неполно раскрываются причинно-следственные связи между явлениями и событиями. Демонстрируются поверхностные знания вопроса, с трудом решаются конкретные задачи. Имеются затруднения с выводами. Допускаются нарушения норм литературной речи |
| 4 | Неудовлетворительно | Материал излагается непоследовательно, сбивчиво, не представляет определенной системы знаний по дисциплине. Не раскрываются причинно-следственные связи между явлениями и событиями. Не проводится анализ. Выводы отсутствуют. Ответы на дополнительные вопросы отсутствуют. Имеются заметные нарушения норм литературной речи |
| 5 | Зачтено | Выставляется при соответствии параметрам экзаменационной шкалы на уровнях «отлично», «хорошо», «удовлетворительно». |

| | | |
|---|------------|---|
| 6 | Не зачтено | Выставляется при соответствии параметрам экзаменационной шкалы на уровнях «отлично», «хорошо», «неудовлетворительно». |
|---|------------|---|

ТИПОВЫЕ ЗАДАНИЯ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ:

2.2. Защита лабораторной работы

- а) типовые задания для лабораторных работ (Приложение 2)
- б) критерии оценивания

При оценке знаний на защите лабораторной работы учитывается:

1. Уровень сформированности компетенций.
2. Уровень усвоения теоретических положений дисциплины, правильность формулировки основных понятий и закономерностей.
3. Уровень знания фактического материала в объеме программы.
4. Логика, структура и грамотность изложения вопроса.
5. Умение связать теорию с практикой.
6. Умение делать обобщения, выводы.

| № п/п | Шкала оценивания | Критерии оценивания |
|-------|---------------------|--|
| 1 | Отлично | Студент выполнил все задания в полном объеме, умеет обоснованно излагать свои мысли и делать необходимые выводы. |
| 2 | Хорошо | Студент выполнил работу полностью, но допустил в ней не более одной негрубой ошибки и одного недочета, или не более двух недочетов. Умеет обоснованно излагать свои мысли и делать самостоятельно необходимые выводы. |
| 3 | Удовлетворительно | Студент выполнил более 2/3 работы и допустил в ней не более двух негрубых ошибок и двух недочетов. Умеет обоснованно излагать свои мысли и делать необходимые выводы, исправляемые после замечания преподавателя. |
| 4 | Неудовлетворительно | Студент демонстрирует отдельные, несистематизированные навыки, допускает ошибки при выполнении заданий, выполняет задание при подсказке преподавателя. Студент не способен правильно выполнить самостоятельно задание или выполнил менее 60% от общего объема заданий. |

2.3. Тест

- а) типовые вопросы к проведению тестирования (Приложение 2)
- б) критерии оценки

При оценке знаний по результатам тестов учитывается:

1. Уровень сформированности компетенций.
2. Уровень усвоения теоретических положений дисциплины, правильность формулировки основных понятий и закономерностей.
3. Уровень знания фактического материала в объеме программы.
4. Логика, структура и грамотность изложения вопроса.
5. Умение связать теорию с практикой.
6. Умение делать обобщения, выводы.

| № п/п | Оценка | Критерии оценки |
|-------|--------|-----------------|
|-------|--------|-----------------|

| 1 | 2 | 3 |
|---|---------------------|---|
| 1 | Отлично | если выполнены следующие условия: – даны правильные ответы не менее чем на 90% вопросов теста, исключая вопросы, на которые студент должен дать свободный ответ; – на все вопросы, предполагающие свободный ответ, студент дал правильный и полный ответ |
| 2 | Хорошо | если выполнены следующие условия: – даны правильные ответы не менее чем на 75% вопросов теста, исключая вопросы, на которые студент должен дать свободный ответ; – на все вопросы, предполагающие свободный ответ, студент дал правильный ответ, но допустил незначительные ошибки и не показал необходимой полноты |
| 3 | Удовлетворительно | если выполнены следующие условия: – даны правильные ответы не менее чем на 50% вопросов теста, исключая вопросы, на которые студент должен дать свободный ответ; – на все вопросы, предполагающие свободный ответ, студент дал непротиворечивый ответ, или при ответе допустил значительные неточности и не показал полноты |
| 4 | Неудовлетворительно | если студентом не выполнены условия, предполагающие оценку «удовлетворительно». |
| 5 | Зачтено | выставляется при соответствии параметрам экзаменационной шкалы на уровнях «отлично», «хорошо», «удовлетворительно» |
| 6 | Не зачтено | выставляется при соответствии параметрам экзаменационной шкалы на уровне «неудовлетворительно» |

3. Перечень и характеристики процедуры оценивания знаний, умений, навыков, характеризующих этапы формирования компетенций

Процедура проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине регламентируется локальным нормативным актом.

Перечень и характеристика процедур текущего контроля и промежуточной аттестации по дисциплине

| № | Наименование оценочного средства | Периодичность и способ проведения процедуры оценивания | Виды вставляемых оценок | Форма учета |
|----|----------------------------------|--|--|---------------------------------------|
| 1. | Зачет | Раз в семестр, по окончании изучения дисциплины | зачтено/не зачтено | Ведомость, зачетная книжка, портфолио |
| 2. | Защита лабораторной работы | Систематически на занятиях | По пятибалльной шкале | Журнал успеваемости преподавателя |
| 3. | Тест | Систематически на занятиях | По пятибалльной шкале (зачтено/не зачтено) | Журнал успеваемости преподавателя |

Типовые вопросы к зачету

ПК-3

1. Принятия управленческих решений при создании проекта для юнит-тестов
2. Управление проектами. Стандарты создания юнит-тестов
3. Управление проектами. Регламенты создания юнит-тестов
4. Принятия управленческих решений при управлении иерархическими данными
5. Переопределение шаблонов формирования при проектировании программного обеспечения
6. Применение методологии управления проектами при проектировании программных интерфейсов веб-приложения
7. Выбор средств сборки и базы знаний. Разработка модели данных на основе структуры выбранной базы данных.
8. Работа с формами в Bootstrap. Программные интерфейсы.
9. Bootstrap. Модальные окна.
10. Создание привязки модели по результатам спроектированной структуры базы данных. Принятие управленческого решения по выбору средств создания спроектированной базы данных.

ПК-9

11. Формирование технического задания на разработку приложения на платформе NET в соответствии со стандартами программной инженерии.
12. Описание целей проекта. Синтаксис, особенности программирования. Контроллеры. Методы действий и их параметры.
13. Использование языка программирования C# для написания программного кода генерации представлений и оценка успешности достижения результатов в соответствии с критериями.
14. Стандартные библиотеки. Строго типизированные представления.
15. Стандартные библиотеки. Частичные представления.
16. Работа с формами. Алгоритмы оптимизации исполняемого кода.
17. Строго типизированные хелперы. Синтаксис. Особенности.
18. Модели и БД. Методы оптимизации программного кода системного ПО. Подготовка документации по разработке системного ПО.
19. Работа со сложными моделями. Особенности программирования.
20. Шаблоны формирования. Сопровождение программного обеспечения.
21. Сопровождение программного обеспечения при работе с моделями со связью многие-ко-многим. Подготовка документации по сопровождению.
22. Создание пагинации. Синтаксис. Стандартные библиотеки.
23. Оптимизация исполняемого кода при реализации миграции баз данных
24. Определение маршрутов. Создание ограничений для маршрутов.
25. Создание собственного обработчика маршрутов.

Типовые задания для лабораторных работ

ПК-9

Лабораторная работа №1

1. Создайте две Web-страницы. На одной странице вводится два числа, при нажатии на гиперссылку числа передаются второй странице, где выводится результат их суммы.
2. Создайте две страницы. На одной странице вводится фамилия сотрудника, при нажатии на гиперссылку фамилия передается второй странице, где выводится информация о сотруднике. Данные о сотруднике хранятся в коллекции.
3. Создайте две страницы. На одной странице вводится название группы, при нажатии на кнопку открывается вторая страница, где выводится список студентов группы в GridView
4. Создайте две страницы. На одной странице вводится название города-прибытия, при нажатии на кнопку открывается вторая страница, где выводится список поездов, следующих до указанной станции в GridView
5. Создать хэш-таблицу, содержащую сведения о студентах вашей группы

ПК-3

Лабораторная работа №2

1. Создайте две основные таблицы: Сотрудники (Шифр, ФИО, дата рождения, пол) и Должности (Название, Базовый оклад). Сотрудник может занимать несколько должностей одновременно. По каждой должности сотрудника должен быть указан размер ставки. Также необходимо вести учет (по признаку) на каких должностях сотрудник еще числится, а с каких уже уволен с указанием причины увольнения. База данных должна хранить дату принятия на работу и дату увольнения по каждой должности сотрудника, сведения о начислениях заработной платы каждому сотруднику по каждой должности с учетом года и месяца.
2. Разработайте Web-приложения для возможности добавления и просмотра данных. При добавлении сотрудника необходимо проверять сколько ставок (по всем должностям) на текущий момент он занимает. Если данный показатель больше 2, выводить соответствующее предупреждение.
3. Добавьте в разработанное Web-приложения возможность выборки данных (для вывода списка используйте объект GridView):

- 3.1. Выбрать сотрудников определенной должности.
- 3.2. Выбрать все занимаемые должности сотрудника на текущий момент с окладом (Базовый оклад * Размер ставки) и датой принятия на эту должность.
- 3.3. Вывести список сотрудников, принятых в выбранный интервал времени.
- 3.4. Вывести название всех имеющихся должностей с указанием максимального и минимального оклада

Если необходимо в LINQ запросе использовать выборку из нескольких таблиц, то запрос будет иметь следующий вид:

```
var Запрос = from Таб1 in Таблица1
              join Таб2 in Таблица2 on Таб1.Поле equals Таб2.Поле
              where условие отбора
              select new { Таб1.Поле1, Таб1.Поле2, Таб2.Поле1, . . . };
```

4. Предусмотреть в системе следующие функции:

1) Формирование ведомости начисления заработной платы за определенной месяц года

Месяц _____ Год _____

| № | ФИО | Суммарный Оклад | Премия | НДФЛ | К выдаче |
|---|-----|--------------------|--------|------|----------|
|---|-----|--------------------|--------|------|----------|

$$\text{Суммарный Оклад} = V_1 * S_1 + V_1 * S_1 + \dots + V_n * S_n$$

V – базовый оклад по должности 1, S – занимаемая ставка по должности 1, n – количество занимаемых должностей

Премия: начисляется в размере 15% от оклада, если сотруднику в этом месяце день рождения.

НДФЛ: 13% от оклада

К выдаче: Оклад + Премия – НДФЛ

2) Формирование сведений о движении сотрудника в хронологическом порядке

ФИО сотрудника _____

| № | Должность | Дата принятия | Дата увольнения | Причина увольнения |
|---|-----------|------------------|--------------------|-----------------------|
|---|-----------|------------------|--------------------|-----------------------|

При создании Web-форм используйте различные стили путем применения каскадных таблиц стилей (Cascading Style Sheets - CSS).

Лабораторная работа №3

1. Создайте таблицы: Пользователи (id,логин,пароль,ФИО), Сообщения (id, от_кого, кому, заголовок_сообщения, текст, дата_отправки).

2. Создать главную Web-форму с кнопками «Регистрация», «Вход»

3. При нажатии на кнопку «Регистрация» должна открываться вторая Web-форма где пользователь может зарегистрироваться с текстовыми полями: логин, пароль, ФИО и кнопкой «Сохранить». Введенные данные должны сохраняться в таблице «Пользователи».

Для перехода на другую страницу используйте код:

```
Response.Redirect("имя Web-формы");
```

4. При нажатии на кнопку «Вход» на главной форме должна открываться третья Web-форма с текстовыми полями: логин, пароль и кнопкой «Вход». Пользователь вводит свои данные. Если они являются верными (сравнение с данными, которые хранятся в таблице), то приложение перенаправляет пользователя на четвертую Web-форму или выдается сообщение, что логин и пароль неверны.

5. На четвертой Web-форме при правильном вводе идентификационных данных, должна появиться фамилия, имя и отчество пользователя, а также заголовки сообщений, которые были ему адресованы и от кого (ник). Заголовки сообщения являются гиперссылками, при щелчке на который будет открывать пятая Web-форма с текстом сообщения. Для реализации данной функции необходимо создать объект класса `HyperLink`. Свойству `Text` объекта присвоить отображаемое на форме имя гиперссылки – заголовок сообщения, а свойству `NavigateUrl` - имя пятой страницы с параметром (n1):

```
объект.NavigateUrl = "имя web-страницы?n1=" + id;
```

В качестве параметра будет передаваться id сообщения. Количество гиперссылок должно равняться количеству сообщений для пользователя. Для этого объект класса `HyperLink` будет создаваться в цикле `foreach`, реализованный для перебора всех сообщений пользователя в таблице «Сообщения».

Для добавления гиперссылки на форму используйте код:

```
this.Controls.Add(объект_класса HyperLink);
```

Пятая Web-форма будет принимать переданное ей значение, используя код:

```
id = Request.QueryString.Get("n1");
```

Извлекаем переданный параметр `n1` (в данном случае прописные и строчные буквы равнозначны) методом `QueryString.Get` объекта `Request`.

По полученному `id` будет нетрудно получить текст сообщения. Для вывода текста сообщения на Web-страницу используйте метод `Write` объекта `Response`.

6. Пользователь может отправлять сообщения. Для этого необходимо поместить 2 текстовых поля и кнопку «Отправить». В первое текстовое поле вводится ник получателя, а во второе – сообщение. Все отправленные сообщения сохраняются в таблице «Сообщения».

ПК-9

Лабораторная работа №4

1. Создайте таблицы Поезда (Номер, Ст_отправления, Ст_назначения) и Расписание (Номер_поезда, Дата, Время). Создайте HTML-форму, на которой пользователь вводит станцию отправления и назначения, дату. На второй странице получает список всех поездов, которые отправляются в указанную дату по необходимому маршруту.

2. Добавьте в проект, созданный в лабораторной работе №3 возможность хранить в cookie-файлах введённые логин и пароль на третьей Web-форме.

3. Разместите таймер на четвертой Web-форме, созданной в лабораторной работе №3.

4. На форме регистрации пользователя, созданной в лабораторной работе №3, добавьте объект `Image`. Сгенерируйте число от 0 до 5 и выведите его на изображение (картинку)

5. Добавьте еще одно текстовое поле, куда пользователь будет вводит проверочный код. Если сгенерированный код совпадет с введенным, регистрация успешно завершится, иначе будет объявлено об ошибке.

ПК-3

Лабораторная работа №5

1. Создать потребителя сервиса, который обращается к серверу, передавая ему введенную фамилию сотрудника. Сервер должен вернуть полную информацию о сотруднике. Сведения необходимо хранить в коллекции.

2. Создать потребителя сервиса, который обращается к серверу, передавая ему номер рейса и количество покупаемых мест. Сервер должен вернуть информацию о сумме покупки билетов или сообщение «Столько свободных мест нет». Сведения хранить в коллекции

Лабораторная работа №6

Изучить предметную область, разработать базу данных и Web-приложение.

Вариант 1. СОБСТВЕННИКИ ЖИЛЬЯ

Предметная область. Имеется реестр собственников квартир, о которых известна следующая информация: адрес квартиры (район, улица, номер дома, номер корпуса, номер квартиры), ФИО собственника, его дата рождения, пол, тип документа владельца квартиры (паспорт, свидетельство о рождении), серия и номер документа. В реестре фиксируется дата начала владения, дата окончания владения (не обязательное значение, которое заполняется только в случае, если собственность передана другому человеку). Собственники квартир могут быть владельцами единолично либо иметь определенную долю квартиры (например, 1, 1/3, 3/4 и т.д.). В числе собственников могут быть несовершеннолетние дети. Квартиры могут не иметь ни одного собственника.

СЕРВЕРНАЯ ЧАСТЬ

1. Создать базу данных в соответствии с предметной областью. Определить необходимые отношения, ограничения для полей. Установить связи для поддержки ссылочной целостности.

2. Создать триггеры:

А) для формирования значений первичных ключей.

Б) для проверки того, что дата окончания владения или не определена, или больше даты начала владения.

В) для проверки правильности формирования "Серия документа" и "Номер документа".

Маска серии:

- для свидетельства о рождении – X...X-КК, где X...X – латинское число, КК – две русские буквы.

- для паспорта – 4 цифры.

Маска номера: шесть цифр (обратите внимание на то, что в номере могут быть ведущие нули).

3. Создать пакет, содержащий следующие функции и процедуры:

А) Функция, принимающая в качестве параметров две даты и возвращающая строку "несовершеннолетний", если между этими датами прошло менее 18-и лет и одного дня. Если вторая дата не определена, считать до текущей даты.

Б) Функция, проверяющая правильность поля "Пол". Параметры: отчество и пол. Возвращаемое значение – пустая строка или строка "не соответствует", если отчество не соответствует установленному полу. Основные правила: мужской пол – отчество оканчивается на 'ИЧ'; женский пол – окончание отчества 'НА'.

В) Функция, преобразующая значение ФИО в фамилию с инициалами (например, "Иванов Иван Сергеевич" в "Иванов И.С.").

Г) Процедура, возвращающая по номеру дома список квартир, по которым нет информации о собственниках.

Д) Процедура, возвращающая по номеру дома список квартир, у которых в настоящее более 5 собственников или квартир, у которых с начала первого владения более 10 собственников, в том числе, бывших.

КЛИЕНТСКАЯ ЧАСТЬ

1. Создать Web – приложение, которое будет взаимодействовать с серверной частью. Web-приложение должно содержать форму авторизации для ввода логина и пароля. Все комбинации логин и пароль необходимо хранить в отдельной таблице БД. При вводе логина и пароля осуществлять процедуру аутентификации и в случае успеха, открывать главную интерфейсную форму с меню для перехода на другие формы приложения. В приложении реализовать две роли: администратор, оператор. Самостоятельно определить набор функций для каждой роли. Администратор может добавлять новых пользователей - операторов.

2. Создать интерфейсные формы для возможности добавления, просмотра и редактирования данных таблиц.

3. Создать интерфейсные формы для возможности выполнения и просмотра результатов функций и процедур серверной части (при необходимости).

4. Реализовать необходимые расчеты, согласно выданному заданию.

5. Создать интерфейсные формы для выполнения запросов:

5.1. единоличных владельцев помещений (доля которых равна 1) с указанием адреса квартиры, которой он владеет;

5.2. квартир, в числе собственников которых в настоящее время есть несовершеннолетние дети;

6. Спроектировать и создать выходной документ, который содержит сгруппированные по районам квартиры, в числе собственников которых в настоящее время есть несовершеннолетние дети с долей не менее 1/3 со списком (ФИО, дата рождения) собственников. По каждому району вывести общее количество таких квартир.

ПК-3

1. Какие утверждения являются неверными с точки зрения методологии управления проектами разработки программного обеспечения?

- а) Web-страница имеет доступ к компонентам пользовательского элемента, если класс пользовательского элемента содержит дополнительные свойства и функции доступа к нужным членам индивидуально для каждого компонента
- б) Web-страница всегда имеет доступ к компонентам интегрированного пользовательского элемента программного интерфейса
- в) Web-страница не может иметь доступ к компонентам интегрированного пользовательского элемента программного интерфейса

2. При проектировании программного обеспечения в соответствии с методологией управления проектами было принято решение использовать класс DataReader. Какие возможности предоставляет класс DataReader?

- а) запись данных
- б) чтение данных
- в) сортировка данных
- г) отображение данных

3. При проектировании программного обеспечения было принято решение использовать аутентификацию. Каким образом аутентификация может быть реализована в ASP.NET при принятии управленческого решения по выбору средств создания и сборки?

- а) с помощью паспортов
- б) с помощью форм
- в) с помощью «самодельных» классов
- г) с использованием Windows

4. По какому принципу работают каскадные таблицы стилей согласно нормативно-техническим документам?

- а) стиль внутреннего каскада перекрывает действие стилей всех внешних каскадов
- б) стиль внешнего каскада перекрывает действие стилей всех внутренних каскадов
- в) стили не могут быть вложенными

5. При формировании программного интерфейса, какой класс, представляющий списковые Web-элементы, в HTML трансформируется в дескриптор <select> с атрибутом size= «1»?

- а) RadioButtonList
- б) DropDownList
- в) CheckBoxList
- г) ListBox
- д) BulletedList

6. Какое событие является первым среди перечисленных событий жизненного цикла страницы пользовательского программного интерфейса?

- а) Page_PreLoad
- б) Page_PreInit
- в) Page_Load
- г) Page_Init

7. Какие управленческие решения могут быть приняты, когда при передаче пакета потери данных не обнаружено?

- а) принимающая сторона отправляет об этом подтверждение
- б) принимающая сторона отправляет требование повторить передачу пакета
- в) принимающая сторона прекращает получение пакетов

ПК-9

8. В каком случае представлен синтаксис для отображения черным цветом в текстовом поле TextBox2 текста «Поле2»?

а) `TextBox1.Style["color"] = "red";`
`TextBox1.Style.Add("background-color", "yellow");`
`TextBox2.Style.Add("color", "black");`
`TextBox1.Text = "Поле1";`

б) `TextBox1.Style["color"] = "red";`
`TextBox1.Style.Add("background-color", "yellow");`
`TextBox2.Style.Add("color", "black");`
`TextBox1.Text = "Поле1";`
`TextBox2.Text = "Поле2";`

в) `TextBox1.Style["color"] = "red";`
`TextBox2.Style["background-color"] = "blue";`
`TextBox1.Style.Add("background-color", "yellow");`
`TextBox1.Text = "Поле1";`
`TextBox2.Text = "Поле2";`

9. В каком случае форма отправляется обратно на сервер при изменении пользователем текущего выбора? Выберите верный синтаксис.

а) `<asp:TextBox ID="TextBox1" runat="server" AutoPostBack="False" OnTextChanged="CtrlChanged">Текстовое поле</asp:TextBox>`

б) `<asp:RadioButton ID="RadioButton2" runat="server" GroupName="radioGroup" OnCheckedChanged="CtrlChanged" />`

в) `<asp:RadioButton ID="RadioButton1" runat="server" AutoPostBack="True" Checked="True" GroupName="radioGroup" OnCheckedChanged="CtrlChanged" />`

10. В каком случае описан синтаксис селектор HTML-тега?

а) `.btnResult`
{
 `background-image:url(BackgroundImage.bmp);`
 `color:white;`
}

б) `h1`
{
 `color: red;`
 `text-align: center;`
}

в) `#TextBox1`

```
{  
  font-family:"Courier New", Courier, mono;  
  font-size:24px;  
  font:bold italic;  
  background-color:#0000FF;  
  color:yellow;  
  width:400;  
}
```

11. В документации по разработке системного программного обеспечения для валидации используется стандартная библиотека языка программирования. Какой валидатор следит, чтобы поле TextBox2, не было пустым?

- а) `<asp:CompareValidator ID="Val2" runat="server" ControlToValidate="TextBox2" Display="Dynamic" Text="Заполните правильно поле заказа" Type="Integer" ControlToCompare="TextBox1" Operator="LessThanEqual" />`
- б) `<asp:CompareValidator ID="Val3" runat="server" ControlToValidate="TextBox2" Display="Dynamic" Text="Заполните правильно поле заказа" ValueToCompare="0" Operator="GreaterThan" />`
- в) `<asp:RequiredFieldValidator ID="Val1" runat="server" ControlToValidate="TextBox2" Display="Dynamic" Text="Укажите количество товара" />`

12. Учитывая особенности программирования ASP .NET, укажите какой из перечисленных файлов является необязательным для приложения и не учитывается при оценке успешности проекта?

- а) Global.asax
- б) Web.config
- в) Default.aspx

13. В целях проекта указано, что для оптимизации кода необходимо сгенерировать несколько HTML-дескрипторов. Элементы какой группы способны генерировать несколько HTML-дескрипторов?

- а) Standard
- б) Validation
- в) HTML

14. Какие файлы не используются при развертывании сайта на рабочем сервере?

- а) файлы решения
- б) файлы содержимого сайта
- в) рабочие файлы сайта для ASP-машины

15. Какой элемент языка программирования используется, чтобы HTML трансформировать в дескриптор `<select>` с атрибутом `size= «N»`, где N — количество видимых элементов?

- а) CheckBoxList
- б) ListBox
- в) BulletedList
- г) DropDownList

16. Какие данные содержит ASP.NET-папка Bin?

- а) файлы .cs исходного кода классов

- б) стилевые файлы для страниц и элементов управления
- в) файлы данных приложения .mdf и .xml
- г) скомпилированные сборки .dll, на которые ссылается приложение

17. Какие утверждения являются неверными с точки зрения программной инженерии?

- а) события изменения могут генерироваться ASP.NET в произвольном порядке
- б) события изменения следуют в порядке следования дескрипторов на форме
- в) наступление события щелчка генерируется первым независимо от расположения кнопки
- г) Web-страница имеет доступ к компонентам пользовательского элемента, если класс пользовательского элемента содержит для каждого компонента по одному дополнительному свойству, возвращающему ссылку на соответствующий компонент

18. Какие утверждения являются неверными для файла приложения Global.asax?

- а) файл является необязательным для приложения
- б) файл должен находиться в любом подкаталоге виртуального каталога приложения
- в) при наличии файла он должен быть единственным

19. Что такое состояние приложения?

- а) область памяти сервера, выделяемая при запуске приложения для хранения локальных данных, доступных с текущей страницы приложения
- б) область памяти сервера, выделяемая при запуске приложения для хранения локальных данных, доступных с любой страницы приложения
- в) область памяти сервера, выделяемая при запуске приложения для хранения глобальных данных, доступных с любой страницы приложения

20. Выберите верный синтаксис использования выражений привязки данных?

- а) `<form method="post" runat="server">`
 `<asp:TextBox ID="TextBox1" runat="server" Text=< %# GetFilePath() %> />
`
`</form>`
- б) `<form method="post" runat="server">`
 `<asp:Image ID="Image1" runat="server" ImageUrl='< %# FilePath %>' />
`
`</form>`
- в) `<form method="post" runat="server">`
 `<asp:Label ID="Label1" runat="server" Text="< %# FilePath %>" />
`
`</form>`