**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Поволжский государственный университет телекоммуникаций и информатики»**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**«УТВЕРЖДАЮ»**

Декан факультета ИСТ

наименование факультета

Салмин А.А.

подпись Фамилия И.О.

« 28 » августа 2015 г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДИСЦИПЛИНЫ**

**Периферийные устройства**

наименование учебной дисциплины (полное, сокращенное)

|  |  |
| --- | --- |
| **Направление (специальность) подготовки** | 02.03.03 - Математическое обеспечение и  администрирование информационных систем |
|  | код и наименование направления (специальности) подготовки |
| **Профиль (специализация) подготовки** | Технология программирования |
|  | указывается при наличии |
| **Квалификация (степень) выпускника** | бакалавр |
|  | бакалавр, магистр, дипломированный специалист |
| **Факультет** | Информационных систем и технологий (ИСТ) |
|  | наименование факультета |
| **Кафедра** | Программное обеспечение и управление в технических системах (ПОУТС) |
|  | наименование кафедры |
| **Форма обучения** | очная |
|  | очная, заочная и т. п. |
|  |  |
| **Курс / семестр** | 3 / 5 |

**ФОС обсужден на заседании кафедры**  ПОУТС

Протокол № 1 от « 28 » 08 2015 г.

Заведующий кафедрой ПОУТС

наименование кафедры

Тарасов В.Н.

подпись Фамилия И.О.

« 28 » 08 2015 г.

**Самара**

**2015**

**Фонд оценочных средств дисциплины**

***«Периферийные устройства»***

***Ответственный исполнитель, уполномоченный по качеству кафедры ПОУТС***

доцент Мезенцева Е.М. \_\_\_\_\_\_\_\_\_

должность подпись фамилия, имя, отчество дата

***Исполнители:***

доцент \_\_\_\_\_\_\_\_\_\_\_ Стефанов М.А. \_\_\_\_\_\_\_\_\_

должность подпись фамилия, имя, отчество дата

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

должность подпись фамилия, имя, отчество дата

***Рецензент***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

должность уч. степень, уч. звание подпись фамилия, имя, отчество

« \_\_\_ » \_\_\_\_\_\_\_\_\_ 2015 г.

1. **Карта компетенций дисциплины**

***«Периферийные устройства»***

**Цель дисциплины:** получение базовых знаний о технологиях Java для реализации распределенных приложений и применение полученных знания для решения практических задач.

**Задачи:** изучение языка разметки xml и инструментов обработки xml-фалов, методов соединения и работы с базой данных в Java-приложениях, получение базовых навыков верстки html-документов, изучение основ обработки сервером http-запросов на основе технологии сервлетов, изучение технологий реализации распределенных Java-приложений – вызов методов удаленных объектов, работа с сессиями, асинхронная обработка сообщений, знакомство с инструментами и фреймворками создания веб-интерфейсов пользователя.

В процессе освоения данной дисциплины студент формирует и демонстрирует следующие компетенции:

Общепрофессиональные компетенции:

ОПК-4: Способность применять в профессиональной деятельности основные методы и средства автоматизации проектирования, производства, испытаний и оценки качества программного обеспечения

ОПК-7: Способность использовать знания основных концептуальных положений функционального, логического, объектно-ориентированного и визуального направлений программирования, методов, способов и средств разработки программ в рамках этих направлений

ОПК-8: Способность использовать знания методов проектирования и производства программного продукта, принципов построения, структуры и приемов работы с инструментальными средствами, поддерживающими создание программного обеспечения

ОПК-9: Способность использовать знания методов организации работы в коллективах разработчиков ПО, направления развития методов и программных средств коллективной разработки ПО

|  |  |  |  |
| --- | --- | --- | --- |
| Код | Перечень компонентов | Технологии формирования\*\* | Форма оценочного средства \*\*\* |
| *ОПК-4* | Знать методы рефакторинга кода  Уметь разворачивать созданные приложения в рамках серверов приложений  Владеть навыками использования средств генерации xml-файлов, схем, dtd. | Лекции, лабораторные работы, самостоятельная работа | Устный опрос, лабораторная работа, экзамен |
| *ОПК-7* | Знать объектно-ориентированные подходы для работы с SQL базами данных  Уметь использовать объектно-ориентированный язык Java для разработки компонентов распределенных программных систем  Владеть навыками работы с фреймворком Java Data Base Connectivity | Лекции, лабораторные работы, самостоятельная работа | Устный опрос, лабораторная работа, экзамен |
| *ОПК-8* | Знать шаблоны проектирования крупных распределенных систем: mode-view-controller, model-view-presenter  Уметь настраивать среды разработки для сборки программных продуктов  Владеть навыками запуска и настройки, а так же развертывания приложений в рамках сервера приложений GlassFish | Лекции, лабораторные работы, самостоятельная работа | Устный опрос, лабораторная работа, экзамен |
| *ОПК-9* | Знать принципы работы систем контроля версий  Уметь работать с репозиториями исходных кодов и ресурсов разрабатываемого ПО  Владеть навыками парного программирования | Лекции, лабораторные работы, самостоятельная работа | Устный опрос, лабораторная работа, экзамен |
| Уровни освоения компетенции (*ОПК-4*) | | | |
| Пороговый уровень  Знать инструменты автоматической обработки xml-файлов  Уметь настроить параметры сервера приложений в интегрированной среде разработки  Владеть навыками использования инструментов автоматизации создания xml-файлов  Продвинутый уровень  Знать инструменты автоматической обработки html, xml, css -файлов  Уметь разворачивать приложения в рамках запущенного сервера приложений  Владеть навыками использования инструментов автоматизации создания xml-файлов и dtd-файлов.  Высокий (превосходный) уровень  Знать инструменты автоматической обработки jsp и jsf -файлов  Уметь создавать дескрипторы развертывания для управления процессом развертывания приложения или компонентов приложения  Владеть навыками использования инструментов генерации xml-файлов на основе схем и наоборот. | | | |
| Уровни освоения компетенции (*ОПК-7*) | | | |
| Пороговый уровень  Знать библиотеки тэгов JSP (JSTL) для работы с sql-базами данных, а также иметь представление о языке JPQL.  Уметь создавать веб-компоненты с помощью фрейморка JSP.  Владеть основными API фреймворка JDBC для соединения с БД, создания, выполнения и обработки результатов sql-запросов.  Продвинутый уровень  Знать Java Entity Persistence и фреймворк JPA для отображения сущностей sql баз данных на объекты  Уметь создавать компоненты, реализующие бизнес-логику распределенного приложения на основе спецификации EJB 2.0 (спецификация Java EE 1.4)  Владеть навыками работы с Criteria API.  Высокий (превосходный) уровень  Знать фреймворк Hibernate для представления сущностей sql баз данных как объектов  Уметь создавать компоненты, реализующие бизнес-логику распределенного приложения на основе спецификации EJB 3.1 (спецификация Java EE 7)  Владеть навыками выполнения запросов к базам данных с использованием фреймворка JDBC в рамках распределенных компонентов (настройка ресурсов на сервере приложений, инжекция ресурсов) | | | |
| Уровни освоения компетенции (*ОПК-8*) | | | |
| Пороговый уровень  Знать смысл шаблонов проектирования, и основные подходы к их реализации.  Уметь настраивать состав компонента, создаваемого средой разработки для его развертывания в рамках сервера приложения  Владеть навыками запуска сервера приложений GlassFish  Продвинутый уровень  Знать основные принципы и структуру шаблона проектирования model-view-presenter.  Уметь использовать систему автоматизации процесса сборки энтерпрайз-компонентов «Maven»  Владеть навыками создания, удаления и администрирования доменов сервера приложений GlassFish  Высокий (превосходный) уровень  Знать основные принципы и структуру шаблона проектирования model-view-controller.  Уметь использовать систему автоматизации процесса сборки энтерпрайз-компонентов «Gradle»  Владеть навыками развертывания и администрирования приложений в рамках домена сервера приложений GlassFish | | | |
| Уровни освоения компетенции (*ОПК-9*) | | | |
| Пороговый уровень  Знать основные принципы работы систем контроля версий  Уметь настраивать клиентское ПО системы контроля версий GitHub  Владеть навыками декомпозиции задачи на простые «подзадачи»  Продвинутый уровень  Знать пути интеграции клиентского ПО систем управления версиями с интегрированными средами разработки  Уметь создавать репозитории исходных кодов и ресурсов на сервере системы контроля версий GitHub  Владеть навыками «парного программирования»  Высокий (превосходный) уровень  Знать принципы работы с системой контроля версий «GitHub»  Уметь вносить изменения, просматривать историю изменений, восстанавливать исходные коды и ресурсы из репозитория, контролируемого системой контроля версий GitHub  Владеть навыками декомпозиции крупной задачи на «подзадачи» и их распределения по членам команды с последующим контролем выполнения | | | |

**2. Паспорт фонда оценочных средств дисциплины**

*«Периферийные устройства»*очнаяформа обучения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Контролируемые разделы (темы) дисциплины** | **Код контролируемой компетенции** | **ФОС\*\*\*** | |
| **Форма оценочного средства** | **Комплект оценочных средств и кол-во вариантов заданий** |
| **1** | **2** | **3** | **4** | **5** |
| **1** | Обработка xml | (ОПК-4) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **2** | Введение в HTML и CSS | (ОПК-4, 7) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **3** | Введение в Java Script | (ОПК-4, 7) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **4** | Удаленный вызов методов | (ОПК-4, 7) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **5** | JDBC | (ОПК-4, 7) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **6** | Servlets. JSP | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **7** | EJB. Session Beans | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **8** | Entity Beans | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **9** | JMS. MDB | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **10** | EJB3 | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **11** | JPA | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **12** | JSF | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
| **13** | AJAX | (ОПК-4, 7, 8, 9) | Опрос. Отчет по лабораторным работам | Тематическая структура банка контрольных вопросов |
|  | **Промежуточная аттестация**  *(вид)* | ОПК- 4, 7, 8, 9 | **Задачи творческого уровня** | **Список задач к экзамену** |

*\*Наименование темы (раздела) или тем (разделов) приводится в соответствии с РП дисциплины (модуля).*

*\*\*Код контролируемой компетенции указывается в соответствии ФГОС.*

*\*\*\*Примерный перечень оценочных средств приведен в Приложении Д.*

**3. Комплект оценочных средств дисциплины\***

*«Периферийные**устройства»*очнаяформа обучения

|  |  |  |
| --- | --- | --- |
| **№** | **Наименование комплекта оценочных средств** | **№ приложения** |
| 1 | Тематическая структура банка контрольных вопросов | А |
| 2 | Список задач к экзамену | Б |

**4. ЛИСТ согласования ФОС дисциплины на 20\_\_/20\_\_ учебный год**

**Направление подготовки:** 02.03.03 Математическое обеспечение и

администрирование информационных систем

шифр и наименование

**Профиль подготовки** Технология программирования

наименование

**Квалификация (степень)**

**выпускника:** Бакалавр

наименование

**Дисциплина:** Периферийные устройства

наименование

**Форма обучения:**  очная

(очная, очно-заочная, заочная)

**Учебный год 2015 / 2016**

**Рекомендована заседанием кафедры** ПОУТС

наименование кафедры

протокол № \_\_\_ от «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_г.

Заведующий кафедрой ПОУТС

наименование кафедры

\_\_\_\_\_\_\_\_\_\_\_ Тарасов В.Н. \_\_\_\_\_\_\_\_\_

подпись расшифровка подписи дата

***Ответственный исполнитель, уполномоченный по качеству кафедры***

доцент Мезенцева Е.М. \_\_\_\_\_\_\_\_\_

должность подпись фамилия, имя, отчество дата

***Исполнители:***

Доцент Стефанов М.А.

должность подпись расшифровка подписи дата

***СОГЛАСОВАНО:***

Заведующий кафедрой ПОУТС

наименование кафедры

\_\_\_\_\_\_\_\_\_\_\_ Тарасов В.Н. \_\_\_\_\_\_\_\_\_

подпись расшифровка подписи дата

Декан ФЗО \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

подпись расшифровка подписи дата

Зам. декана по (направлению) специальности 02.03.03

шифр наименование

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ Чернова С.В. \_\_\_\_\_\_\_\_\_

должность подпись расшифровка подписи дата

**5. ЛИСТ изменений и дополнений в ФОС дисциплины**

**Дополнения и изменения в ФОС дисциплины на 20\_\_/20\_\_ уч.г.**

В ФОС дисциплины вносятся следующие изменения:

1) …………………………………..;

2) …………………………………..;

3) ……………………………………

или делается отметка о нецелесообразности внесения каких-либо изменений на данный учебный год

**ФОС дисциплины пересмотрен и одобрен на заседании кафедры \_\_\_\_\_\_\_\_\_**

Протокол № \_\_\_\_\_ от « \_\_\_ » \_\_\_\_\_\_\_\_\_ 201\_\_ г.

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

наименование кафедры

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

подпись расшифровка подписи дата

***СОГЛАСОВАНО:***

Декан ФЗО \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

подпись расшифровка подписи дата

Зам. декана по (направлению) специальности \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

шифр наименование

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_

должность подпись расшифровка подписи дата

**Внесенные изменения на 201\_\_/201\_\_ учебный год *УТВЕРЖДАЮ***

Декан факультета \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

наименование факультета

\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись расшифровка подписи

« \_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 201\_\_ г.

Приложение А

Федеральное агентство связи

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Поволжский государственный университет телекоммуникаций и информатики»

Кафедра ПОУТС

Тематическая структура банка контрольных вопросов по дисциплине

«*Периферийные устройства»*

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование темы  (раздела) | Контрольные вопросы по разделам дисциплины | Коды контролируемых  компетенций |
| **1** | Обработка xml | 1. Теги и атрибуты 2. Подходы к обработке XML 3. Назначение XSLT 4. Принципы JAXB 5. DTD 6. XML – схемы | (ОПК-4) |
| **2** | Введение в HTML и CSS | 1. отличия от XML 2. Виды DTD для html-документов 3. Виды тэгов 4. Селекторы CSS 5. Подключение таблиц ситлей к документу | (ОПК-4, 7) |
| **3** | Введение в Java Script | 1. Объекты в JS 2. Функции 3. Формат JSON 4. События | (ОПК-4, 7) |
| **4** | Удаленный вызов методов | 1. Архитектура системы RMI 2. Реестр RMI 3. Заглушки 4. Требования к удаленным объектам | (ОПК-4, 7) |
| **5** | JDBC | 1. Виды драйверов 2. Подключение к БД 3. Формирование запросов к БД 4. Обработка результатов запросов | (ОПК-4, 7) |
| **6** | Servlets. JSP | 1. Обработка Http-запросов сервером 2. Формирование отклика на запрос 3. Структура JSP-документов 4. Назначение элементов JSP-документа 5. Назначение скриплетов 6. Библиотеки тэгов JSP | (ОПК-4, 7, 8, 9) |
| **7** | EJB. Session Beans | 1. Назначение EJB 2. Виды сессионных бинов 3. Компонентный интерфейс 4. Home-интерфейс 5. Основные параметры дискрипторов развертывания | (ОПК-4, 7, 8, 9) |
| **8** | Entity Beans | 1. Назначение entity beans и их виды 2. Особенности BMP 3. Особенности CMP 4. Интерфейсы, реализуемые компонентами 5. Основные принципы JPQL 6. Организация один к одному, один ко многим, многие ко многим | (ОПК-4, 7, 8, 9) |
| **9** | JMS. MDB | 1. Назначение и основные объекты системы сообщений JMS 2. Виды сообщений 3. Интерфейсы, реализуемые компонентами MDB 4. Очереди 5. Топики | (ОПК-4, 7, 8, 9) |
| **10** | EJB3 | 1. Основные отличия EJB3 от EJB2 2. Аннотации для настройки сессионных бинов 3. Аннотации для настройки MDB 4. Инжекция зависимостей | (ОПК-4, 7, 8, 9) |
| **11** | JPA | 1. Назначение менеджера персистенции 2. Принципы объектного представления сущностей БД 3. Организация отношении 1 к 1, 1 ко многим, многие ко многим средствами JPA. 4. Criteria API 5. Модуль персистентности | (ОПК-4, 7, 8, 9) |
| **12** | JSF | 1. Основные элементы JSF 2. Принципы навигации 3. Настройка навигации 4. Язык выражений 5. Managed Bean | (ОПК-4, 7, 8, 9) |
| **13** | AJAX | 1. Принципы AJAX веб-приложений 2. Статусы AJAX запросов 3. Работа с XML-данными 4. Работа с HTML-данными 5. Работа с JSON-данными | (ОПК-4, 7, 8, 9) |

Коды контролируемых компетенций ОПК- 4, 7, 8, 9.

**Критерии оценки:**

- оценка «***зачтено***» выставляется студенту, если он ответил на свыше 49% от количества вопросов;

- оценка «***не*** ***зачтено***» выставляется студенту, если он ответил на менее 49% от количества вопросов.

Составитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.А. Стефанов

(подпись)

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2015 г.

Приложение Б

Федеральное агентство связи

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Поволжский государственный университет телекоммуникаций и информатики»

Кафедра ПОУТС

Список задач к зачету по дисциплине

«*Периферийные устройства»*

**Задача №1 Календарь**

Реализовать небольшой on-line сервис – календарь. Сервис определяется несколькими страницами:

1. Страница входа, отображается по-умолчанию и содержит 2 текстовых поля ввода логина и пароля, а так же 2 кнопки: «войти» и «зарегистрироваться». При нажатии на первую кнопку происходит переход на страницу календаря (или страницу ошибки с сообщением о не корректности логина или пароля), на вторую – форму регистрации.
2. Форма регистрации содержит 3 текстовых поля: логин, пароль, mail и кнопку «зарегистрироваться» при нажатии на которую происходит либо переход на страницу ошибки, с сообщением о том, что введенный пользователем логин уже зарегистрирован, либо на страницу календаря.
3. Календарь по-умолчанию отображает события на **текущей** неделе. Каким именно образом отображается эта информация каждый студент решает сам (можно не заморачиваться и сделать в виде списка, а дни разделить линией (аля <hr> или <br> - всегда путал какой из них делает перевод, а какой добавляет черту =)))).

Предусмотреть 2 кнопки – «следующая неделя» и «предыдущая неделя», при нажатии на которых формируется страница календаря, отображающего, соответственно, следующую или предыдущую недели.

Так же, на этой странице предусмотреть форму добавления и удаления события – поля ввода даты и времени начала события (можно использовать поля выбора дат и времени: date, time, datetime-local), названия события, длительности (часы и минуты), и кнопки “добавить” и “удалить”. Событие добавляется, если введены все 3 поля (можно использовать значения по-умолчанию – подумайте сами какие именно). При добавлении события формируется страница, отображающая неделю, на которой было добавлено событие. Удаление события работает только если введена информацию о названии, даты и времени начала события.

*Доп\* Предусмотрите возможность заполнения полей формы добавления\удаления события средствами JS – при клике на событие, значения полей формы заполняется данными этого события.*

Таблицы БД:

1. Пользователи (ID – primary key, логин – unique key, пароль, mail)
2. События (день, время, название, длительность, userID – foreign key на ID таблицы «пользователи»)

Допущения:

1. пароль можно передавать и хранить в БД в не шифрованном виде.
2. пароль можно передавать по не защищенному *http*, но методом *post*.
3. параметры сессии можно в куках не хранить

**Задача №2 Сервис обмена простым текстом**

Реализовать сервис обмена текстом. Вводимый пользователем текст сохраняется на сервере и предоставляется ссылка, по которой любой пользователь может ознакомиться с текстом (либо так-же отредактировать, в зависимости от опций сохранения). Сервис определяется несколькими страницами:

1. Страница входа, отображаемая по-умолчанию, содержит 2 текстовых поля ввода логина и пароля, а так же 3 кнопки: «войти», «зарегистрироваться», «войти без регистрации». При нажатии на первую кнопку происходит переход на страницу редактирования текста (или страницу ошибки с сообщением о не корректности логина или пароля), на вторую – форму регистрации, на третью – страницу просмотра текста.
2. Форма регистрации содержит 3 текстовых поля: логин, пароль, mail и кнопку «зарегистрироваться» при нажатии на которую происходит либо переход на страницу ошибки, с сообщением о том, что введенный пользователем логин уже зарегистрирован, либо на страницу редактирования.
3. Страница редактирования содержит многострочное текстовое поле, в которое пользователь может вводить текст, кнопку «сохранить». По-умолчанию, выделяется переключатель «доступ на чтение». При нажатии на кнопку сохранить формируется страница с текстовым полем, в котором отображается URL. Страница по этому URL должна отобразить форму входа. Если пользователь осуществил вход, то можно отредактировать этот текст (то есть в текстовом поле будет введен текст), если же пользователь кнопку “войти без регистрации” – отображается страница, содержащая только текст (без элементов управления).

Таблицы БД:

1. Пользователи (ID – primary key, логин – unique key, пароль, mail)
2. Тексты (ID – primary key, текст, userID – foreign key на ID таблицы «пользователи»

Допущения:

1. пароль можно передавать и хранить в БД в не шифрованном виде.
2. пароль можно передавать по не защищенному *http*, но методом *post*.
3. параметры сессии можно в куках не хранить

**Задача №3 Сервис заметок**

Реализовать небольшой on-line сервис заметок. Сервис определяется несколькими страницами:

1. Страница входа, отображается по-умолчанию и содержит 2 текстовых поля ввода логина и пароля, а так же 2 кнопки: «войти» и «зарегистрироваться». При нажатии на первую кнопку происходит переход на страницу заметок (или страницу ошибки с сообщением о не корректности логина или пароля), на вторую – форму регистрации.
2. Форма регистрации содержит 3 текстовых поля: логин, пароль, mail и кнопку «зарегистрироваться» при нажатии на которую происходит либо переход на страницу ошибки, с сообщением о том, что введенный пользователем логин уже зарегистрирован, либо на страницу заметок.
3. Страница заметок. По-умолчанию отображается последняя редактированная заметка Состоит из нескольких областей:
   1. Слева отображается список заметок (название и дата последнего изменения заметки). Каждая заметка – ссылка, при нажатии на которую происходит переход к соответствующей заметке.
   2. Справа (б*о*льшая часть экрана) отображается текстовое поле ввода имени заметки и многостраничное поле с содержимым выбранной заметки. Текст заметки можно редактировать.
   3. Снизу расположены кнопки:
      1. новая заметка. При нажатии на нее происходит переход на эту же страницу, но с пустыми текстовыми полями.
      2. сохранить. При нажатии на нее происходит создание новой заметки. В поле слева добавляется ссылка на эту запись, страница по-прежнему должна отображать сохраненную заметку.
      3. удалить. При нажатии на нее происходит удаление выбранной заметки. В поле слева ссылка на заметку более не отображается и страница отображает содержимое следующей (по списку заметок) заметки.
      4. выход. При нажатии на нее происходит переход на страницу входа

Примерные таблицы БД:

1. Пользователи (ID – primary key, логин – unique key, пароль, mail)
2. Заметки (ID – primary key, название, дата изменения, userID – foreign key на ID таблицы «пользователи»)

Допущения:

1. пароль можно передавать и хранить в БД в не шифрованном виде.
2. пароль можно передавать по не защищенному *http*, но методом *post*.
3. параметры сессии можно в куках не хранить

**Задача №4 Недвижимость**

Реализовать on-line сервис объявлений о покупке и продаже недвижимости (офисов и квартир). Сервис определяется несколькими страницами:

1. Страница поиска объявлений (открывается по-умолчанию). Содержит поля выбора параметров поиска: тип-недвижимости (офис, квартира), город, улица, номер дома, номер офиса(квартиры), общая площадь, площадь балкона, этаж, количество комнат, стоимость, контактные данные. *Опционально: при поиске офиса, количество комнат и площадь балкона становятся недоступны.* Также на странице есть две кнопки:
   1. поиск – при нажатии на эту кнопку происходит переход на страницу результатов поиска
   2. добавить – при нажатии на эту кнопку информация о недвижимости добавляется в БД и происходит переход на страницу, содержащую ссылку на страницу объявления..
2. Результаты поиска. Страница отображает список найденных квартир. По каждой квартире отображается только адрес, стоимость и ссылка на страницу объявления. Результаты поиска сортируются по дате добавления (начиная с самых последних). Также определить элемент управления, позволяющий выбрать поле сортировки результатов (стоимость, площадь, число комнат)
3. На странице объявления отображается вся введенная при добавлении информация (в любой форме).

Примерные таблицы БД:

1. Тип (ID – primary key, тип – офис\квартира)
2. Недвижимость (ID – primary key, город, улица, typeID - foreign key на ID таблицы «Тип», общая площадь, площадь балкона, этаж, количество комнат, стоимость, контактные данные)

Коды контролируемых компетенций ОПК- 4, 7, 8, 9.

**Критерии оценки знаний, умений и навыков**

Промежуточная аттестация осуществляется в форме экзамена, который проводится по билетам, включающим 1 практическую задачу. Оценка знаний студентов производится по следующим критериям:

оценка «*отлично*» выставляется студенту, если он реализовал более 90% основных функций в соответствии с заданием;

оценка «*хорошо*» выставляется студенту, если он реализовал более 70% основных функций в соответствии с заданием;

оценка «*удовлетворительно*» выставляется студенту, если он реализовал более 50% основных функций в соответствии с заданием;

оценка «*не удовлетворительно*» выставляется студенту, если он реализовал менее 50% основных функций в соответствии с заданием.

Составитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.А. Стефанов

(подпись)

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2015