

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное учреждение  
высшего профессионального образования  
"Казанский (Приволжский) федеральный университет"  
Институт вычислительной математики и информационных технологий



**УТВЕРЖДАЮ**

Проректор  
по образовательной деятельности КФУ  
Проф. Минзарипов Р.Г.

\_\_\_\_\_ 20\_\_ г.

**Программа дисциплины**

Разработка Интернет приложений Б3.В.3

Направление подготовки: 231000.62 - Программная инженерия

Профиль подготовки: Технологии разработки информационных систем

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

**Автор(ы):**

Хадиева А.И.

**Рецензент(ы):**

Хадиев К.Р. , Туйкин А.М.

**СОГЛАСОВАНО:**

Заведующий(ая) кафедрой: Еникеев А. И.

Протокол заседания кафедры No \_\_\_\_ от " \_\_\_\_ " \_\_\_\_\_ 201\_\_ г

Учебно-методическая комиссия Института вычислительной математики и информационных технологий:

Протокол заседания УМК No \_\_\_\_ от " \_\_\_\_ " \_\_\_\_\_ 201\_\_ г

Регистрационный No

Казань  
2015

## Содержание

1. Цели освоения дисциплины
2. Место дисциплины в структуре основной образовательной программы
3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля
4. Структура и содержание дисциплины/ модуля
5. Образовательные технологии, включая интерактивные формы обучения
6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов
7. Литература
8. Интернет-ресурсы
9. Материально-техническое обеспечение дисциплины/модуля согласно утвержденному учебному плану

Программу дисциплины разработал(а)(и) ассистент, б/с Хадиева А.И. кафедры технологий программирования отделение фундаментальной информатики и информационных технологий, AllHadieva@kpfu.ru

### 1. Цели освоения дисциплины

Цель курса ? познакомить студентов с основами Web-технологий и дизайна Web-систем. Познакомить с принципами архитектуры "Клиент-Сервер". Познакомить с основными технологиями, подходами и принципами написания Web-приложений. Научить писать собственные проекты.

### 2. Место дисциплины в структуре основной образовательной программы высшего профессионального образования

Данная учебная дисциплина включена в раздел " Б3.В.3 Профессиональный" основной образовательной программы 231000.62 Программная инженерия и относится к вариативной части. Осваивается на 3 курсе, 6 семестр.

Данная дисциплина относится к профессиональным дисциплинам.

Читается для студентов 4 курса, обучающихся по направлению "Программная инженерия".

### 3. Компетенции обучающегося, формируемые в результате освоения дисциплины /модуля

В результате освоения дисциплины формируются следующие компетенции:

| Шифр компетенции | Расшифровка приобретаемой компетенции  |
|------------------|--|
| опк-1            | способность разрабатывать концептуальные и теоретические модели решаемых научных проблем и задач проектной и производственно-технологической деятельности  |
| Опк-3            | способность использовать и применять углубленные теоретические и практические знания в области фундаментальной информатики и информационных технологий   |
| Опк-4            | способность самостоятельно приобретать и использовать в практической деятельности новые знания и умения, в том числе, в новых областях знаний, непосредственно не связанных со сферой деятельности, расширять и углублять своё научное мировоззрение |
| пк-10            | способность применять в профессиональной деятельности современные языки программирования и языки баз данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии  |

В результате освоения дисциплины студент:

#### 1. должен знать:

понимать специфику программирования веб систем, принципы подхода веб программирования, особенности в сравнении с другими языками для написания локального ПО, использующиеся технологии, методологии и подходы в современной интернет сети, основные термины. Владеть теоретическими знаниями об основных подходах веб программирования и их использовании при написании веб систем.

#### 2. должен уметь:

Создавать веб-приложения, применения веб технологий в разработке локальных приложений и веб систем

3. должен владеть:

принципами "Клиент-Серверной" архитектуры, принципами создания веб-приложений

к созданию полноценных интернет приложений, участвуя в командной работе.

#### 4. Структура и содержание дисциплины/ модуля

Общая трудоемкость дисциплины составляет 3 зачетных(ые) единиц(ы) 108 часа(ов).

Форма промежуточного контроля дисциплины зачет в 6 семестре.

Суммарно по дисциплине можно получить 100 баллов, из них текущая работа оценивается в 50 баллов, итоговая форма контроля - в 50 баллов. Минимальное количество для допуска к зачету 28 баллов.

86 баллов и более - "отлично" (отл.);

71-85 баллов - "хорошо" (хор.);

55-70 баллов - "удовлетворительно" (удов.);

54 балла и менее - "неудовлетворительно" (неуд.).

#### 4.1 Структура и содержание аудиторной работы по дисциплине/ модулю

##### Тематический план дисциплины/модуля

| N  | Раздел<br>Дисциплины/<br>Модуля   | Семестр | Неделя<br>семестра | Виды и часы<br>аудиторной работы,<br>их трудоемкость<br>(в часах) |                         |                        | Текущие формы<br>контроля |
|----|---|---------|--------------------|---|-------------------------|------------------------|---------------------------|
|    |   |         |                    | Лекции  | Практические<br>занятия | Лабораторные<br>работы |                           |
| 1. | Тема 1. Концепция клиент-серверной архитектуры                                  | 6       | 1                  | 2   | 0                       | 2                      |                           |
| 2. | Тема 2. Облачные вычисления   | 6       | 2                  | 2   | 0                       | 2                      |                           |
| 3. | Тема 3. Протоколы HTTP и HTTPS  | 6       | 3                  | 2   | 0                       | 2                      |                           |
| 4. | Тема 4. Безопасности web-приложений   | 6       | 4                  | 2   | 0                       | 2                      |                           |
| 5. | Тема 5. Разделение логики между клиентом и сервером. Архитектура Web-приложений | 6       | 5                  | 2   | 0                       | 2                      |                           |
| 6. | Тема 6. Клиентские задачи   | 6       | 6-12               | 4   | 0                       | 14                     |                           |
| 7. | Тема 7. Серверные задачи  | 6       | 13-18              | 4   | 0                       | 12                     |                           |
|    | Тема . Итоговая форма контроля  | 6       |                    | 0   | 0                       | 0                      | зачет                     |
|    | Итого   |         |                    | 18  | 0                       | 36                     |                           |

#### 4.2 Содержание дисциплины

##### Тема 1. Концепция клиент-серверной архитектуры

**лекционное занятие (2 часа(ов)):**

1. Клиент серверная архитектура разделение труда 2. Виды клиентов от толстого до тонкого на жизненном примере. Плюсы, минусы. 3. Как клиент-серверная архитектура применяется к Web-технологиям

**лабораторная работа (2 часа(ов)):**

Схематичная разработка архитектуры приложения.

**Тема 2. Облачные вычисления**

**лекционное занятие (2 часа(ов)):**

1. Что такое облачные вычисления. 2. Виды облачных платформ.

**лабораторная работа (2 часа(ов)):**

Примеры и решения клиентских задач. Базовые примеры.

**Тема 3. Протоколы HTTP и HTTPS**

**лекционное занятие (2 часа(ов)):**

1.Протокол HTTP 2. Протокол HTTPS

**лабораторная работа (2 часа(ов)):**

1. Введение в HTML 2. Введение в Java Script

**Тема 4. Безопасности web-приложений**

**лекционное занятие (2 часа(ов)):**

1.Безопасность Сервера 2. Безопасность Клиента

**лабораторная работа (2 часа(ов)):**

3. Работа с текстом. 4. Создание раскрывающегося теста

**Тема 5. Разделение логики между клиентом и сервером. Архитектура Web-приложений**

**лекционное занятие (2 часа(ов)):**

1 Раздельное хранение состояния. 2 Разделение задач на клиентские и серверные

**лабораторная работа (2 часа(ов)):**

1. Динамическая смена картинок 2. Плавное появление и скрытие 3. Убегающий элемент

**Тема 6. Клиентские задачи**

**лекционное занятие (4 часа(ов)):**

Технологии клиентской части. 1 Общие принципы 2 Краткие сведения об HTML5, CSS, JavaScript, AJAX.

**лабораторная работа (14 часа(ов)):**

1. Показ текущей даты и времени в реальном времени. 2. Стрелочные часы 3. Создать таймер отсчета времени до ближайшего нового года.

**Тема 7. Серверные задачи**

**лекционное занятие (4 часа(ов)):**

Технологии серверной части. 1 История развития 2 MVC как пример Клиент-серверной архитектуры модулей приложения

**лабораторная работа (12 часа(ов)):**

Написать игру "крестики нолики".

**4.3 Структура и содержание самостоятельной работы дисциплины (модуля)**

| N  | Раздел Дисциплины   | Семестр | Неделя семестра | Виды самостоятельной работы студентов  | Трудоемкость (в часах) | Формы контроля самостоятельной работы |
|----|---|---------|-----------------|--|------------------------|---------------------------------------|
| 1. | Тема 1. Концепция клиент-серверной архитектуры                                  | 6       | 1               | 1. Создание проектов с использованием клиент-серверной архитектуры. Разработка концепции проекта и т | 2                      | контрольная работа                    |
| 2. | Тема 2. Облачные вычисления   | 6       | 2               | 2. Проектирование различных видов облачные вычисления, таких как: SaaS, IaaS, PaaS.                  | 6                      | контрольная работа                    |
| 3. | Тема 3. Протоколы HTTP и HTTPS  | 6       | 3               | 3. Проектирование сообщений в соответствие с протоколы HTTP и HTTPS.                                 | 6                      | контрольная работа                    |
| 4. | Тема 4. Безопасности web-приложений   | 6       | 4               | 4. Создание сценариев атаки и защиты web-приложения.   | 6                      | контрольная работа                    |
| 5. | Тема 5. Разделение логики между клиентом и сервером. Архитектура Web-приложений | 6       | 5               | 5. Решение задач на разделение логики между Клиентом и сервером. Проектирование архитектура Web-при  | 6                      | контрольная работа                    |
| 6. | Тема 6. Клиентские задачи   | 6       | 6-12            | 6. Решение типовых задач на в клиентской части Web-приложений.                                       | 7                      | контрольная работа                    |
|    |   |         |                 | 7. Решение задач клиентской части используя Javascript.  | 7                      | контрольная работа                    |

| N     | Раздел Дисциплины        | Семестр | Неделя семестра | Виды самостоятельной работы студентов  | Трудоемкость (в часах) | Формы контроля самостоятельной работы |
|-------|--------------------------|---------|-----------------|--|------------------------|---------------------------------------|
| 7.    | Тема 7. Серверные задачи | 6       | 13-18           | 10. Решение задач клиентской части используя язык Java и платформу Spring Framework.         | 3                      | контрольная работа                    |
|       |                          |         |                 | 11. Решение задач клиентской части используя язык Java и платформу Spring Framework часть 2. | 4                      | контрольная работа                    |
|       |                          |         |                 | 8. Проектирование серверной части Web-приложений.  | 4                      | контрольная работа                    |
|       |                          |         |                 | 9. Проектирование серверной части Web-приложений часть 2.                                    | 3                      | контрольная работа                    |
| Итого |                          |         |                 |  | 54                     |                                       |

### 5. Образовательные технологии, включая интерактивные формы обучения

Освоение дисциплины "Разработка интернет приложений" предполагает использование как традиционных ( лекции и самостоятельная работа с использованием компьютеров, методических материалов ) , так и инновационных образовательных технологий с использованием в учебном процессе мультимедийных программ.

### 6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

#### Тема 1. Концепция клиент-серверной архитектуры

контрольная работа , примерные вопросы:

1. Создание проектов с использованием клиент-серверной архитектуры. Разработка концепции проекта и технического задания

#### Тема 2. Облачные вычисления

контрольная работа , примерные вопросы:

2. Проектирование различных видов облачные вычисления, таких как: SaaS, IaaS, PaaS.

#### Тема 3. Протоколы HTTP и HTTPS

контрольная работа , примерные вопросы:

3. Проектирование сообщений в соответствии с протоколы HTTP и HTTPS.

#### **Тема 4. Безопасности web-приложений**

контрольная работа , примерные вопросы:

4. Создание сценариев атаки и защиты web-приложения.

#### **Тема 5. Разделение логики между клиентом и сервером. Архитектура Web-приложений**

контрольная работа , примерные вопросы:

5. Решение задач на разделение логики между Клиентом и сервером. Проектирование архитектура Web-приложений

#### **Тема 6. Клиентские задачи**

контрольная работа , примерные вопросы:

контрольная работа , примерные вопросы:

#### **Тема 7. Серверные задачи**

контрольная работа , примерные вопросы:

Проектирование серверной части Web-приложений. часть 1

контрольная работа , примерные вопросы:

контрольная работа , примерные вопросы:

контрольная работа , примерные вопросы:

#### **Тема . Итоговая форма контроля**

Примерные вопросы к зачету:

1 Что такое Клиент-Серверная архитектура.

2 Виды клиентов. В чем их отличия. В чем преимущества и недостатки.

3 По каким параметрам нужно выбирать толщину клиента.

4 Как Клиент серверная архитектура проявляется в Web-приложениях?

5 Приведите аналогии Клиент-Сервера с Коммуникационными вычислениями.

6 Что такое облачные вычисления? В чем преимущества и недостатки?

7 Какие есть основные виды облачных платформ. Какие преимущества недостатки? Как по задаче определить какой вид использовать?

8 Что такое протокол HTTP? Какие есть обязательные блоки для запроса и ответа? Какие есть основные методы передачи параметров, в чем отличие?

9 Что такое HTTPS и как он работает?

10 Расскажите о безопасности Клиента

11 Расскажите о безопасности Сервера

12 Расскажите о безопасности от прослушивания.

13 Как любое приложение представляется в виде автомата? Какие особенности состояний системы у многопользовательских приложений, в частности Web-приложений?

14 Какие способы хранения состояния на сервере? В чем особенности, когда применяются, приведите примеры.

15 Какие способы хранения состояния на клиенте? В чем особенности, когда применяются, приведите примеры.

16 Как распределяется информация о состоянии и функциональность между клиентом и сервером?

17 Назревшая необходимость технологии HTML5? В чем отличия HTML5 от предыдущей версии?

18 Кратко расскажите о HTML. Для чего его используют в рамках подхода HTML5?

19 Кратко расскажите о CSS. Для чего его используют в рамках подхода HTML5?

20 Кратко расскажите о JavaScript. Для чего его используют в рамках подхода HTML5?



- 21 Что такое Servlet и где применяется аналогичный подход к серверной части приложения? Когда этот подход необходим?
- 22 Что такое JSP и где применяется аналогичный подход к серверной части приложения? Когда этот подход необходим. Что такое JSTL, его плюсы и минусы?
- 23 Что такое модульная организация приложения, когда нужна, какие плюсы и минусы?
- 24 Что такое архитектура MVC, когда нужна, какие плюсы и минусы?

### 7.1. Основная литература:

### 7.2. Дополнительная литература:

### 7.3. Интернет-ресурсы:

Видеоуроки и статьи по изучению верстки - <http://www.sdelaysite.com/>  
Множество примеров готовых решений в web-разработки - <http://ruseller.com/>  
Различные примеры и статьи по JavaScript - <http://javascript.ru/>  
Справочник по HTML тегам и CSS свойствам - <http://htmlbook.ru/>  
Что почитать, чтобы повысить свой уровень JavaScript ( - <http://habrahabr.ru/post/117838/>

## 8. Материально-техническое обеспечение дисциплины(модуля)

Освоение дисциплины "Разработка Интернет приложений" предполагает использование следующего материально-технического обеспечения:

Мультимедийная аудитория, вместимостью более 60 человек. Мультимедийная аудитория состоит из интегрированных инженерных систем с единой системой управления, оснащенная современными средствами воспроизведения и визуализации любой видео и аудио информации, получения и передачи электронных документов. Типовая комплектация мультимедийной аудитории состоит из: мультимедийного проектора, автоматизированного проекционного экрана, акустической системы, а также интерактивной трибуны преподавателя, включающей тач-скрин монитор с диагональю не менее 22 дюймов, персональный компьютер (с техническими характеристиками не ниже Intel Core i3-2100, DDR3 4096Mb, 500Gb), конференц-микрофон, беспроводной микрофон, блок управления оборудованием, интерфейсы подключения: USB, audio, HDMI. Интерактивная трибуна преподавателя является ключевым элементом управления, объединяющим все устройства в единую систему, и служит полноценным рабочим местом преподавателя. Преподаватель имеет возможность легко управлять всей системой, не отходя от трибуны, что позволяет проводить лекции, практические занятия, презентации, вебинары, конференции и другие виды аудиторной нагрузки обучающихся в удобной и доступной для них форме с применением современных интерактивных средств обучения, в том числе с использованием в процессе обучения всех корпоративных ресурсов. Мультимедийная аудитория также оснащена широкополосным доступом в сеть интернет. Компьютерное оборудование имеет соответствующее лицензионное программное обеспечение.

Компьютерный класс, представляющий собой рабочее место преподавателя и не менее 15 рабочих мест студентов, включающих компьютерный стол, стул, персональный компьютер, лицензионное программное обеспечение. Каждый компьютер имеет широкополосный доступ в сеть Интернет. Все компьютеры подключены к корпоративной компьютерной сети КФУ и находятся в едином домене.

Лекционная аудитория с мультимедийным проектором, ноутбуком и экраном на штативе и дисплейные классы.

Программа составлена в соответствии с требованиями ФГОС ВПО и учебным планом по направлению 231000.62 "Программная инженерия" и профилю подготовки Технологии разработки информационных систем .

Автор(ы):

Хадиева А.И. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.

Рецензент(ы):

Хадиев К.Р. \_\_\_\_\_

Туйкин А.М. \_\_\_\_\_

"\_\_" \_\_\_\_\_ 201\_\_ г.