

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кубанский государственный университет»
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

Иванов А.Г.

«01 июля 2016 г.»

2016 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.10 «Web-программирование»

(код и наименование дисциплины в соответствии с учебным планом)

Направление подготовки/специальность 02.03.03 «Математическое
обеспечение и администрирование информационных систем»

(код и наименование направления подготовки/специальности)

Направленность (профиль) / специализация «Технология программирования»

(наименование направленности (профиля) специализации)

Программа подготовки академическая
(академическая /прикладная)

Форма обучения очная
(очная, очно-заочная, заочная)

Квалификация (степень) выпускника бакалавр
(бакалавр, магистр, специалист)

Краснодар 2016

Рабочая программа дисциплины Web-программирование составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 «Математическое обеспечение и администрирование информационных систем» Программу составил(и):

доцент кафедры информационных технологий КубГУ, к.т.н.



Синица С.Г.

Рабочая программа дисциплины Web-программирование утверждена на заседании кафедры информационных технологий протокол №8 от 23.05.2016

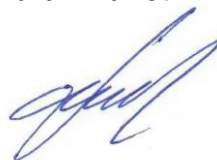
Заведующий кафедрой (разработчика)



Кольцов Ю.В.

Рабочая программа обсуждена на заседании кафедры интеллектуальных информационных систем, протокол №4 от 20.04.2016.

Заведующий кафедрой (выпускающей)



Костенко К.И.

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол №7 от 29 июня 2016 г.

Председатель УМК факультета



Малыхин К.В.

Рецензенты:

1. Рубцов Сергей Евгеньевич, кандидат физико-математических наук, доцент кафедры математического моделирования ФГБГОУ «КубГУ»
2. Бегларян Маргарита Евгеньевна, кандидат физико-математических наук, заведующий кафедрой СГНЕД СКФ ФГБОУ ВО «РГУП»

1. Цели и задачи изучения дисциплины

1.1 Цель освоения дисциплины.

Цель — изучение современных веб-технологий.

1.2 Задачи дисциплины.

- освоение основ веб-архитектуры;
- изучение Hyper Text Transfer Protocol (HTTP) и основ программирования веб-приложений;
- ознакомление с возможностями современных браузеров;
- ознакомление с технологиями веб-сервисов и интернет поиска;
- изучение технологий и подходов обеспечения безопасности в сети интернет.

1.3 Место дисциплины в структуре образовательной программы.

Дисциплина относится к вариативной части Блока 1 "Дисциплины (модули)" учебного плана.

Дисциплина **взаимодействует для формирования компетенций** с большинством дисциплин общенаучного цикла, а также с дисциплинами профессионального цикла:

1. Основы программирования
2. Теория автоматов и формальных языков
3. Параллельное программирование
4. Администрирование информационных систем

Требованием к «входным» знаниям является понимание основ работы сетей и создания сетевых приложений.

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

Перечень результатов образования, формируемых дисциплиной:

1. Владеть основами веб-архитектуры и поиска информации в сети Интернет при разработке программ;
2. Владеть основами HTTP и приобрести практический опыт программирования веб-приложений и поиска информации в сети Интернет с использованием современных систем программных средств, операционных систем и оболочек, сервисных программ;
3. Уметь использовать возможности современных браузеров и поисковых систем на практике;
4. Знать основы технологий веб-сервисов и интернет поиска;
5. Знать технологии и подходы обеспечения безопасности в сети интернет при разработке программ.

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	Знать	Уметь	Владеть
1	ОПК-7	способностью использовать знания основных концептуальных положений функционального, логического, объектно-ориентированного и визуального направлений программирования, методов, способов и средств разработки программ в рамках этих направлений	5)	3)	1)

2 ПК-5 готовностью к использованию современных 4) 3) 2)
систем программных средств; операционных
систем и оболочек, сервисных программ

2. Структура и содержание дисциплины

2.1 Распределение трудоёмкости дисциплины по видам работ.

Общая трудоёмкость дисциплины составляет 5 зач.ед. (180 часов), их распределение по видам работ представлено в таблице (для студентов ОФО).

Вид учебной работы		Всего часов	Семестры (часы)			
			6	—		
Контактная работа, в том числе:						
Аудиторные занятия (всего):						
Занятия лекционного типа		34	34	-	-	-
Лабораторные занятия		50	50	-	-	-
Занятия семинарского типа (семинары, практические занятия)		-	-	-	-	-
		-	-	-	-	-
Иная контактная работа:						
Контроль самостоятельной работы (КСР)		13	13			
Промежуточная аттестация (ИКР)		0,5	0,5			
Самостоятельная работа, в том числе:						
Курсовая работа		-	-	-	-	-
Проработка учебного (теоретического) материала		20	20	-	-	-
Выполнение индивидуальных заданий (подготовка сообщений, презентаций)		20	20	-	-	-
Реферат		-	-	-	-	-
Подготовка к текущему контролю		6,8	6,8	-	-	-
Контроль:						
Подготовка к экзамену		35,7	35,7			
Общая трудоемкость	час.	180	180	-	-	-
	в том числе контактная работа	97,5	97,5			
	зач. ед	5	5			

Процедура промежуточной аттестации проходит в форме зачета и экзамена.

2.2 Структура дисциплины:

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.
Разделы дисциплины, изучаемые в 6 семестре (очная форма).
Вид промежуточной аттестации: зачет, экзамен.

№	Наименование разделов	Количество часов		
		Всего	Аудиторная работа	Внеаудиторная работа

			Л	ЛР	СРС	контроль
1.	Основы веб-архитектуры	26	4	12	6	4
2.	Протокол HTTP	32	8	14	6	4
3.	Возможности браузеров	34	10	14	6	4
4.	Интернет-поиск	12	2		6	4
5.	Безопасность в Сети	16	4	2	6	4
6.	Веб-сервисы	16	4	2	6	4
7.	Новейшие веб-технологии	18	2	4	8	4
8.	Обзор изученного материала и прием зачета	12,5		2	2,8	7,7
9.	Контроль самостоятельной работы (КСР)	13				
10.	Промежуточная аттестация (ИКР)	0,5				
	Итого по дисциплине:	180	34	50	46,8	35,7

Примечание: Л – лекции, ПЗ – практические занятия / семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

2.3 Содержание разделов дисциплины:

№ разд ела	Наименование раздела	Содержание раздела	Форма текущего контроля	Разработано с участием представителя работодателя (указать организацию)
1	Основы веб-архитектуры	DNS, домены, хостинг, URI и URL, гипертекст и гиперссылки	Задачи	
2	Протокол HTTP	Схема работы HTTP, версии протокола, формат запросов и ответов, основные заголовки, механизмы договаривания, авторизации, условных запросов	Задачи	
3	Возможности браузеров	Формы в браузерах, HTML, DOM, CSS, JavaScript, XMLHttpRequest, Cookies и сессии	Задачи	
4	Веб-сервисы	Технологии веб-сервисов	Задачи	Рассматриваются примеры информационных систем, разработанных специалистами ООО «Инитлаб»
5	Интернет-поиск	Введение в архитектуру поисковых систем.	Задачи	Материал подготовлен с привлечением веб-мастеров ООО

		Индексация сайтов поисковиками, введение в поисковую оптимизацию и алгоритмы поисковых систем	«Инитлаб», занимающихся поисковой оптимизацией и продвижением сайтов на практике
6	Безопасность в Сети	SSL и сертификаты безопасности, уязвимости веб-приложений, спам сайтов, безопасность клиентов и серверов	Задачи
7	Новейшие веб-технологии	CMS, обзор возможностей HTML5, SVG, WebGL	Задачи

2.3.1 Занятия лекционного типа.

№ лекции	Раздел	Тема лекции	Содержание лекции	Форма текущего контроля
1	Основы веб-архитектуры	DNS, домены, хостинг	Домены и DNS, хостинг, схема взаимодействия браузера и серверов при запросе веб-страницы.	Задачи
2	Основы веб-архитектуры	URI и URL, гипертекст и гиперссылки, формы	URI и URL. Формы в браузерах.	Задачи
3	Протокол HTTP	Схема работы HTTP, версии протокола, формат запросов и ответов	HTTP, схема работы и основные понятия, версии протокола. Общий вид запроса и ответа в HTTP. Методы запросов.	Задачи
4	Протокол HTTP	Основные заголовки, механизмы договаривания	Основные заголовки запроса и ответа HTTP. Коды статуса ответа HTTP.	Задачи
5	Протокол HTTP	Авторизация	HTTP Authentication. Разбор примеров.	Задачи
6	Протокол HTTP	Условный запрос	Условный GET-запрос. Схема работы кеширующего прокси-сервера. Разбор примеров на PHP.	Задачи
7	Возможности браузеров	Формы в браузерах, HTML	Возможности современного браузера.	Задачи

HTML.

8	Возможности браузеров	DOM, CSS	Алгоритм отображения документа в браузере. CSS.	Задачи
9	Возможности браузеров	JavaScript, XMLHttpRequest	JavaScript, манипуляции DOM, XMLHttpRequest.	Задачи
10	Возможности браузеров	Cookies	Cookies. Примеры на PHP	Задачи
11	Возможности браузеров	Сессии	Сессии. Примеры на PHP	Задачи
12	Веб-сервисы	Технологии веб-сервисов	Веб сервисы. Архитектурные стили веб-сервисов. Примеры.	Задачи
13	Веб-сервисы	REST	Архитектурный стиль REST. Пример.	Задачи
14	Интернет-поиск	Введение в интернет-поиск	Архитектура поисковых механизмов в сети. Индексация сайта поисковиками. Внутренние, внешние и поведенческие факторы ранжирования страниц в поисковиках. Понятие релевантности. Google PageRank и Yandex ТИЦ.	Задачи
15	Безопасность в Сети	Спам и защита от спама, SSL и сертификаты безопасности	Методы спама сайтов. Методы защиты POST-форм от спама. Понятия приватности и безопасности в Сети. SSL шифрование и сертификаты безопасности. Электронная цифровая подпись и электронный документооборот. Законодательство и ГОСТ РФ в сфере инфокоммуникационных технологий.	Задачи
16	Безопасность в Сети	Уязвимости веб-приложений, безопасность клиентов и	Уязвимости веб-приложений XSS, SQL-injection, Include, Upload, CSRF. Защита клиента и	Задачи

		серверов	сервера веб-приложений.	
17	Новейшие веб-технологии	Новейшие веб-технологии	Обзор возможностей HTML5, SVG, WebGL	Задачи

2.3.2 Занятия семинарского типа.

Не предусмотрены

2.3.3 Лабораторные занятия.

№ занятия	Раздел	Тема лабораторного занятия	Содержание лабораторного занятия	Форма текущего контроля
1	Основы веб-архитектуры	Гиперссылки	Задача 1.	Задачи
2	Основы веб-архитектуры	Настройка веб-сервера, работа с PHP	Настройка веб-сервера, работа с PHP.	Задачи
3	Основы веб-архитектуры	Настройка веб-сервера, работа с PHP	Настройка веб-сервера, работа с PHP.	Задачи
4	Основы веб-архитектуры	Настройка веб-сервера, работа с PHP	Работа с IDE на примере Eclipse, удаленная работа с проектами PHP по сети.	Задачи
5	Протокол HTTP	HTTP-Запросы	Задача 2. Putty и telnet	Задачи
6	Протокол HTTP	HTTP-Запросы	Задача 2. Знакомство со спецификацией HTTP	Задачи
7	Протокол HTTP	Введение в PHP	Введение в PHP	Задачи
8	Протокол HTTP	Фреймворк	Написание фреймворка на PHP. Единая точка входа и параметры. Методы запросов.	Задачи
9	Протокол HTTP	Фреймворк	Написание фреймворка. Роутинг и модули. . Шаблоны.	Задачи
10	Возможности браузеров	HTML-форма	Задача 7.	Задачи
11	Возможности браузеров	Работа в PHP с формой и XML	Задача 8.	Задачи
12	Протокол HTTP	HTTP-авторизация	Задача 10.	Задачи
13	Протокол HTTP	HTTP-авторизация	Задача 11.	Задачи
14	Возможности браузеров	XMLHTTPREQUEST	XMLHTTPREQUEST, задача 3.	Задачи
15	Возможности браузеров	jQuery	jQuery, задача 4.	Задачи
16	Основы веб-	Сокеты	Работа с сокетами, задача	Задачи

	архитектуры		5.	
17	Основы веб-архитектуры	Сокеты	Работа с сокетами, задача 6.	Задачи
18	Возможности браузеров	Cookies	Задача 12.	Задачи
19	Возможности браузеров	Сессии	Задача 13.	Задачи
20	Возможности браузеров	jQuery	Задача 14.	Задачи
21	Безопасность в Сети	Безопасность в Сети	Задача 15.	Задачи
22	Веб-сервисы	Веб-сервис	Задача 16.	Задачи
23	Новейшие веб-технологии	CMS	Введение в CMS Drupal	Задачи
24	Новейшие веб-технологии	Веб-сокеты	Чат на веб-сокетах	Задачи
25	Новейшие веб-технологии	Canvas и SVG	Вывод графики с помощью HTML5 Canvas и SVG	Задачи

2.3.4 Примерная тематика курсовых проектов

Сложные:

1. Скрипт. Многопользовательский графический редактор векторной графики (фигуры, блоки текста, соединенные кривыми) с использованием SVG. Пользователи работают по очереди.
2. Веб-сервис. Файловый сервер на Google Application, Python.
3. Веб-сервис. База данных на Google Application, Python.
4. Библиотека. Генератор схемы вышивки крестом в PDF по растровой картинке и информации о используемых цветах и размере вышивки.
5. Веб-сервис. Импорт постов в соц. сеть чтением постов заданного блога через RSS, авторизация по OAuth (веб-сервис не знает пароля пользователя).
- 5.1. Чат комната на Unity.
- 5.2. Онлайн консультации на JavaScript, Jabber и Google Apps
- 5.3. Приложение для iPhone или iPad: отправляет текущие GPS-координаты на сайт, где они пишутся в файл. Инструкция по установке инструментов разработки, сборке и установке программы.
- 5.4. Приложение Android, синхронизация файлов на веб-сайте и флеш-карте, просмотр файлов.
- 5.5. HTML5-приложение под iPad
- 5.6. WindowsPhone клиент Вконтакте

Средней сложности:

6. Библиотека. CAPTCHA с использованием ASCII ART.
7. Веб-сервис. Графическая кнопка, отображающая проиндексированность страницы в Яндексе, дату последней индексации. С использованием запроса к Yandex.XML.
8. Настольное приложение. Поиск и мониторинг освобождающихся доменов с заданным минимальным возрастом, Яндекс ТИЦ и Google PR.

9. Скрипт. Файловый менеджер. Закачка нескольких файлов, отображение списка файлов в каталоге с размером и правами, удаление файлов, переименование и смена прав, создание и удаление каталогов. Без перезагрузки страницы с использованием XMLHttpRequest. Допускается использование jQuery.
10. Библиотека. Сортировка таблицы, удаление строк таблицы на JavaScript без перезагрузки страницы. Допускается использование jQuery. С использованием БД PostgreSQL и/или MySQL.
11. Веб-сервис подписки по RSS на изменение заданной HTML-страницы. Для определения изменения используется преобразование в текст программой html2text и сравнение программой diff.
12. Библиотека. Чат на JavaScript. Сообщения передаются с использованием JSON и XMLHttpRequest, без перезагрузки страницы, и сохраняются в базу данных.

Простые:

13. Скрипт. Гостевая книга. Сообщения хранятся в текстовых файлах. Возможность удаления и изменения постов администратором. Сохранение ника и контактных данных в Cookies.
14. Настольное приложение. Сохранение локальной копии сайта. Обход по внутренним ссылкам.
15. Библиотека. Калькулятор на JavaScript. Возможность ввода цифр, скобок и основных операций мышкой, ввода формул с клавиатуры, вычисление функцией JavaScript eval(). Создавать калькулятор вызовом функции JavaScript с передачей id элемента, куда вывести калькулятор.
16. Через wget, curl, atom скачать письма с gmail.

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	Web-разработка	<ol style="list-style-type: none"> 1. Синица, С.Г. Веб-программирование и веб-сервисы / Краснодар: Кубанский гос. ун-т, 2013. 156 с. 2. Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод.пособие. Ю.В. Кольцов, А.В.Уварова, С.Г.Синица [и др.] – Краснодар: Кубанский гос.ун-т, 2017

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

В соответствии с требованиями ФГОС программа дисциплины предусматривает использование в учебном процессе следующих образовательных технологий: чтение лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

- Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.

- Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.

- Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.

- Информационно-коммуникационные технологии (ИКТ) – расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:

- Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.

- Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.

- Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.

- Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.

- Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.

- Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.

- Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

Комплексное использование в учебном процессе всех вышеназванных технологий стимулируют личностную, интеллектуальную активность, развивают познавательные процессы, способствуют формированию компетенций, которыми должен обладать будущий специалист.

Основные виды интерактивных образовательных технологий включают в себя:

- работа в малых группах (команде) – совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;

- проектная технология – индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

– анализ конкретных ситуаций - анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;

– развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

Подход разбора конкретных задач и ситуаций широко используется как преподавателем, так и студентами во время лекций, лабораторных занятий и анализа результатов самостоятельной работы. Это обусловлено тем, что при исследовании и решении каждой конкретной задачи имеется, как правило, несколько методов, а это требует разбора и оценки целой совокупности конкретных ситуаций.

Семестр	Вид занятия	Используемые интерактивные образовательные технологии	количество интерактивных часов
6	Л, ЛР	Занятия в режимах взаимодействия «преподаватель – студент» и «студент – студент»	16
Итого			16

Темы, задания и вопросы для самостоятельной работы призваны сформировать навыки поиска информации, умения самостоятельно расширять и углублять знания, полученные в ходе лекционных и практических занятий.

Подход разбора конкретных ситуаций широко используется как преподавателем, так и студентами при проведении анализа результатов самостоятельной работы.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

4. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации

4.1 Фонд оценочных средств для проведения текущего контроля.

Выбор языка программирования для реализации веб-приложения остается за студентом, однако задачи курса веб-программирования не предполагают использования сторонних фреймворков, библиотек, систем управления контентом, высокоуровневых компонентов работы с HTTP. Для студентов направлений, ориентированных на программирование, выполнение задач 8-16 предполагает написание своего фреймворка с единой точкой входа,

модульностью и роутингом запросов, HTTP-авторизацией, разделением вёрстки и кода с использованием шаблонов. Студентам направлений, связанных с администрированием программного обеспечения, рекомендуется использовать ОС на базе Linux и настроить веб-сервер самостоятельно.

Задачи текущего контроля

Перечень компетенций, проверяемых оценочным средством:

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

ПК-5 готовностью к использованию современных систем программных средств; операционных систем и оболочек, сервисных программ

- 1. Написать гиперссылки в документе HTML5 и опробовать их работу в браузере:
- 1) абсолютную гиперссылку на главную страницу сайта example.com;
- 2) абсолютную на главную сайта example.com в протоколе https;
- 3) ссылку на файл на сервере FTP без авторизации;
- 4) ссылку на файл на сервере FTP с авторизацией;
- 5) ссылку на фрагмент страницы некоторого сайта;
- 6) ссылку на фрагмент текущей страницы;
- 7) ссылку с двумя параметрами в URL;
- 8) список ссылок основной навигации сайта с подписями title;
- 9) ссылку без href;
- 10) ссылку с пустым href;
- 11) ссылку, по которой запрещен переход поисковикам;
- 12) запрещенную для индексации поисковиками;
- 13) контекстную в тексте абзаца;
- 14) ссылку-изображение;
- 15) ссылки из прямоугольных и круглых областей картинки (HTML-тег map);
- 16) относительную на страницу в текущем каталоге;
- 17) относительную на страницу в каталоге about;
- 18) относительную на страницу в каталоге уровнем выше текущего;
- 19) относительную на страницу в каталоге двумя уровнями выше;
- 20) сокращенную на главную;
- 21) сокращенную ссылку на внутреннюю.
-
- 2. С помощью программы telnet или Putty выполнить задания отправкой HTTP-запросов к веб-серверу:
- 1) получить главную страницу методом GET в протоколе HTTP 1.0;
- 2) получить внутреннюю страницу методом GET в протоколе HTTP 1.1;
- 3) определить размер файла file.tar.gz, не скачивая его;
- 4) определить медиатип ресурса /image.png;
- 5) отправить комментарий на сервер по адресу /index.php;
- 6) получить первые 100 байт файла /file.tar.gz;
- 7) определить кодировку ресурса /index.php.
-

3. Выполнить задание 2 отправкой запросов с помощью объекта XMLHttpRequest в браузере синхронно и асинхронно.

4. Выполнить задание 2 отправкой запросов с помощью библиотеки jQuery в браузере синхронно и асинхронно.

5. Выполнить задание 2 отправкой запросов с помощью сокетов в программе на C, C++ или Delphi без высокоуровневых компонентов работы с сетью и HTTP. При получении ответа определить размер ответа разбором заголовка Content-Length и вычитать точно передаваемое количество байт для быстрого завершения операции чтения из сокета без таймаута.

6. Выполнить задание 2 отправкой запросов из Python, Perl, PHP, Java или другого языка программирования. При получении ответа определить размер ответа разбором заголовка Content-Length и вычитать точно передаваемое количество байт для быстрого завершения операции чтения из сокета без таймаута.

7. Составить HTML-форму с полями:

- имя (текстовое поле);
- e-mail (текстовое поле);
- год рождения (выбор из списка);
- пол (радиокнопки);
- количество конечностей (радиокнопки);
- сверхспособности: бессмертие, прохождение сквозь стены, левитация (множественный выбор из списка);
- биография (многострочное текстовое поле);
- с контрактом ознакомлен (чекбокс);
- кнопка «Отправить».

Оформить страницу красиво с использованием CSS.

8. Реализовать скрипт на веб-сервере на PHP или другом языке, сохраняющий в XML-файл заполненную форму задания 7. При отправке формы на сервере создается новый файл с уникальным именем.

9. Написать HTTP-запрос, который браузер будет отправлять на веб-сервер при отправке заполненной формы задания 7.

10. Реализовать вход администратора с использованием HTTP-авторизации для просмотра и удаления результатов.

11. Реализовать управление пользователями для задачи 10 с хранением данных в XML или базе данных.

12. Реализовать проверку заполнения обязательных полей формы в задаче 7 с использованием Cookies, а также заполнение формы по умолчанию ранее введенными значениями.

13. Реализовать возможность входа с паролем и логином с использованием сессии

для изменения отправленных данных в задаче 7, пароль и логин генерируются автоматически при первоначальной отправке формы.

—

- 14. Реализовать отправку формы в задании 7 с помощью jQuery, если в браузере включен JavaScript. В противном случае форма отправляется как обычно. Результат отправки на сервере принимает один и тот же скрипт. Проверка правильности заполнения формы также по возможности происходит на клиенте и повторяется на сервере.

—

- 15. Проверьте написанное вами веб-приложение для решения предыдущих задач на наличие уязвимостей безопасности. Откомментируйте потенциально уязвимые места и устраните уязвимости.

—

- 16. Реализовать RESTful веб-сервис для получения списка сохраненных результатов и самих результатов в XML, отправки и сохранения новых результатов для задачи 7. Запрос на сохранение отправляет XML-документ. Использовать HTTP-авторизацию. Провести аудит безопасности кода веб-сервиса.

17. Разработать корпоративный сайт с каталогом товаров и формой заказа на CMS Drupal.

—

- Примерные темы рефератов:
- Основные возможности WebGL.
- Основные возможности SVG.
- Основные возможности Canvas.
- Поддержка специальных событий в iOS Safari.
- Геолокация в HTML5.
- Хранение информации на клиенте в HTML5.
- Технологии клиент-серверного взаимодействия в HTML5 (Server Events, WebSockets).

4.2 Фонд оценочных средств для проведения промежуточной аттестации.

Примерная тематика курсовых работ (проектов)

Перечень компетенций, проверяемых оценочным средством:

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

ПК-5 готовностью к использованию современных систем программных средств; операционных систем и оболочек, сервисных программ

1. Скрипт. Многопользовательский графический редактор векторной графики (фигуры, блоки текста, соединенные кривыми) с использованием SVG. Пользователи работают по очереди.
2. Веб-сервис. Файловый сервер на Google Application, Python.
3. Веб-сервис. База данных на Google Application, Python.
4. Библиотека. Генератор схемы вышивки крестом в PDF по растровой картинке и информации о используемых цветах и размере вышивки.
5. Веб-сервис. Импорт постов в соц. сеть чтением постов заданного блога через RSS,

авторизация по OAuth (веб-сервис не знает пароля пользователя).

- 5.1. Чат комната на Unity.
- 5.2. Онлайн консультации на JavaScript, Jabber и Google Apps
- 5.3. Приложение для iPhone или iPad: отправляет текущие GPS-координаты на сайт, где они пишутся в файл. Инструкция по установке инструментов разработки, сборке и установке программы.
- 5.4. Приложение Android, синхронизация файлов на веб-сайте и флеш-карте, просмотр файлов.
- 5.5. HTML5-приложение под iPad
- 5.6. WindowsPhone клиент Вконтакте
6. Библиотека. CAPTCHA с использованием ASCII ART.
7. Веб-сервис. Графическая кнопка, отображающая проиндексированность страницы в Яндексе, дату последней индексации. С использованием запроса к Yandex.XML.
8. Настольное приложение. Поиск и мониторинг освобождающихся доменов с заданным минимальным возрастом, Яндекс ТИЦ и Google PR.
9. Скрипт. Файловый менеджер. Закачка нескольких файлов, отображение списка файлов в каталоге с размером и правами, удаление файлов, переименование и смена прав, создание и удаление каталогов. Без перезагрузки страницы с использованием XMLHttpRequest. Допускается использование jQuery.
10. Библиотека. Сортировка таблицы, удаление строк таблицы на JavaScript без перезагрузки страницы. Допускается использование jQuery. С использованием БД PostgreSQL и/или MySQL.
11. Веб-сервис подписки по RSS на изменение заданной HTML-страницы. Для определения изменения используется преобразование в текст программой html2text и сравнение программой diff.
12. Библиотека. Чат на JavaScript. Сообщения передаются с использованием JSON и XMLHttpRequest, без перезагрузки страницы, и сохраняются в базу данных.
13. Скрипт. Гостевая книга. Сообщения хранятся в текстовых файлах. Возможность удаления и изменения постов администратором. Сохранение ника и контактных данных в Cookies.
14. Настольное приложение. Сохранение локальной копии сайта. Обход по внутренним ссылкам.
15. Библиотека. Калькулятор на JavaScript. Возможность ввода цифр, скобок и основных операций мышкой, ввода формул с клавиатуры, вычисление функцией JavaScript eval(). Создавать калькулятор вызовом функции JavaScript с передачей id элемента, куда вывести калькулятор.
16. Через wget, curl, atom скачать письма с gmail.

Вопросы для промежуточной аттестации по итогам освоения дисциплины:

Перечень компетенций, проверяемых оценочным средством:

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

ПК-5 готовностью к использованию современных систем программных средств; операционных систем и оболочек, сервисных программ

1. Регистрация и хостинг доменов, доменные имена второго и третьего уровня, DNS.
2. Схема взаимодействия браузера и серверов при запросе веб-страницы.

3. Общий вид URI. Сравнение URI (чувствительность к регистру, <</> и <<.>>). Полные, сокращенные и относительные адреса в гиперссылках.
4. Аксиомы URI и URL. Прозрачность URI.
5. Схема работы HTTP. Понятие клиента, сервера, ресурса, представления, сущности, прокси, шлюза. Медиатипы. Отличия HTTP 0.9/1.0/1.1.
6. Виды и стоимость хостинга: Dedicated, Colocation, VDS, VPS, Shared, Cloud. Виртуальные хосты по имени и по IP-адресу.
7. Общий вид запроса и ответа в HTTP: метод, представление, заголовки запроса, ответа и сущности. Стандартные методы HTTP: GET, POST, HEAD, PUT, DELETE. Понятие состояния ресурса и побочного эффекта запроса. Безопасные и идиempотентные методы.
8. Формы в браузерах. Пример отправки формы методом GET и POST. Шаблон приложения обработки форм Post-Redirect-Get.
9. Механизм Content-Negotiation (договаривание) в HTTP. Основные заголовки запроса HTTP: Host, Accept, Accept-Encoding, Accept-Language, User-Agent, Referer. Основные заголовки ответа HTTP: Content-Type, Content-Encoding, Content-Length, Content-Language, Location, Connection, Date, Allow. Коды статуса ответа HTTP. Семантика кодов 200, 301, 302, 303, 307, 404.
10. Схема работы Basic и Digest HTTP Authentication. Код ответа 401, заголовок ответа WWW-Authenticate, заголовок запроса Authorization.
11. Условный GET-запрос. Заголовок запроса If-Modified-Since и заголовок ответа Last-Modified, заголовок запроса If-Match и заголовок ответа ETag, код ответа 304. Схема работы кэширующего прокси-сервера.
12. Возможности современного браузера: HTTP, HTML, DOM, CSS, JavaScript, XMLHttpRequest. Принцип разделения содержимого и представления при использовании HTML и CSS.
13. HTML, различия в версиях, связь с XML. Семантика основных HTML-тегов.
14. Алгоритм отображения документа в браузере: поток, абсолютное позиционирование, боксовая модель, блочные/строчные элементы.
15. Float-элементы. Основные параметры шрифта в CSS. Способы задания стилей.
16. Селекторы CSS1. Каскад (правила применения стилей). Фон элементов в CSS.
17. Вёрстка веб-страниц слоями и таблицами. Преимущества и недостатки вёрстки слоями и таблицами.
18. JavaScript, события и обработчики событий, манипуляция DOM.
19. Объект XMLHttpRequest в браузере IE и прочих. Создание объекта. Синхронные и

асинхронные HTTP-запросы. Основные методы XMLHttpRequest. Преимущества и недостатки AJAX-приложений.

20. Cookies. Схема аутентификации и сохранения состояния на сервере (клиенте) с помощью Cookies.

21. Сессии. Схема авторизации с помощью сессии. Безопасность сессии.

22. Веб-сервисы. Технологии XML/JSON over HTTP, XML-RPC, SOAP/WSDL.

23. Архитектурные стили веб-сервисов RPC, SOA, RESTful. Безопасность веб-сервисов.

24. Архитектурный стиль REST: задачи, ограничения, принципы построения интерфейса.

25. Понятие фреймворка веб-приложения и библиотеки. Схема веб-приложения и фреймворков. Единая точка входа. MVC.

26. Методы спама сайтов: спам POST-форм, referer-спам, trackback-спам. Методы защиты POST-форм от спама: аутентификация, CAPTCHA, черные и белые списки.

27. Архитектура поисковых механизмов в сети. Индексация сайта поисковиками. Внутренние и внешние факторы ранжирования страниц в поисковиках. Понятие релевантности. Google PageRank и Yandex ТИЦ.

28. Понятия приватности и безопасности в Сети. SSL шифрование и сертификаты безопасности. Соккрытие IP-адреса (прокси, TOR). Утечки referer. Приватные данные в браузере. Уязвимости веб-приложений. Схема кражи Cookies при помощи XSS.

29. Уязвимости веб-приложений CSRF, SQL-injection, Include и Upload. Способы выполнения произвольного кода на сервере.

30. Права доступа к файлам в Linux и уязвимости Shared-хостинга. Мотивы взлома клиента и хоста хакерами. Защита клиента и сервера веб-приложений.

31. Обзор возможностей современных веб-технологий и их поддержки браузерами и мобильными платформами: Canvas, WebGL, SVG, WebSockets, CSS3, WebWorkers, SessionStorage, SQL LocalStorage, Geolocation API, HTML5 Audio/Video.

Критерием оценивания результатов освоения дисциплины (зачет) являются результаты текущего контроля. В случае неудовлетворительных результатов по какой-либо индивидуальной задаче, студенту предоставляется возможность повторной сдачи соответствующего элемента контроля.

Критерии оценки:

Оценка	
Незачет	Зачтено
менее 9 задач	9 или более задач

Задача считается решенной если корректно выполнены все условия задания, студент может продемонстрировать работоспособность решения и ответить на любые вопросы о деталях работы программного кода решения и реализованных алгоритмов.

Результат сдачи экзамена заноситься преподавателем в экзаменационную ведомость и зачетную книжку.

Оценивание уровня освоения дисциплины основывается на качестве ответов на вопросы экзамена, решений задач на экзамене. За ответы вопросы на экзамене дается до 20 баллов. За решение задачи на экзамене дается 20 баллов.

Вопросы:

0 баллов: непонимание сущности излагаемых вопросов, грубые ошибки в ответе, неуверенные и неточные ответы на дополнительные вопросы экзаменаторов.

1-10 баллов: знание и понимание основных вопросов программы, студент указал направление решения индивидуальной задачи; частично ответил на два вопроса билета или достаточно полно ответил хотя бы на один вопрос.

11-15 баллов: твёрдые и достаточно полные знания всего программного материала, последовательные, правильные, конкретные ответы на поставленные вопросы при свободном реагировании на замечания по отдельным вопросам; достаточно полно ответил на два вопроса.

16-20 баллов: глубокие исчерпывающие знания всего программного материала, логически последовательные, полные, грамматически правильные и конкретные ответы на вопросы экзаменационного билета и дополнительные вопросы.

Задачи:

0 баллов: решение отсутствует.

10 баллов: верное направление решения, исправление ошибок в решении после наводящих вопросов преподавателя.

20 баллов: верное решение задачи в соответствии со всеми требованиями условия задачи.

Дополнительно 10, 20 или 40 баллов дается за индивидуальные проекты по веб-разработке, тематики которых обозначены в списке примерных тем курсовых работ выше.

Критерии оценки:

Код	Соответствие уровней освоения компетенции планируемым результатам обучения и критериям их оценивания		
	Пороговый	Базовый	продвинутый
	Оценка		
	Удовлетворительно	Хорошо	Отлично
ПК-8	20-30 баллов	31-35 баллов	36-40 баллов

Критерии оценки курсового проекта:

Оценка выставляется на основе: доклада, и представленного текста курсовой работы.

Оценка	Критерий
Отлично	<p>текст стилистически грамотно, логически правильно оформлен; продемонстрирована системность и глубина знаний при выполнении работы</p> <p>стилистически грамотно, логически правильно излагает ответы на дополнительные вопросы;</p> <p>проведен анализ полученных результатов;</p> <p>предложен новый или грамотно обоснован метод исследования/решения задачи;</p>

Оценка	Критерий
Хорошо	текст правильно оформлен; правильно излагает ответы на дополнительные вопросы; предложен новый или грамотно обоснован метод исследования/решения задачи; проведен анализ полученных результатов; продемонстрирован высокий уровень знаний по выбранной тематике
Удовлетворительно	текст оформлен; предложен обоснован метод исследования/решения задачи предложен обоснован метод исследования/решения задачи частично проведен анализ полученных результатов
Неудовлетворительно	не ответил на дополнительные вопросы; не предложен метод исследования/решения задачи; не проведен анализ полученных результатов; не представлен текст

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

- при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

- при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

- при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

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

5.1 Основная литература:

1. Синица С.Г. Веб-программирование и веб-сервисы – учебное пособие, КубГУ, 2013.
2. Богданов, М.Р. Разработка клиентских приложений Web-сайтов : курс / М.Р. Богданов. - Москва : Интернет-Университет Информационных Технологий, 2010. - 228 с. : ил. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233745>.

5.2 Дополнительная литература:

1. Основы web-программирования на PHP: учебное пособие. Маркин А. В. , Шкарин С. С. Москва: Диалог-МИФИ, 2012. Объем: 252 стр. ISBN: 978-5-86404-241-0. URL: http://biblioclub.ru/index.php?page=book_red&id=229742
3. Громов Ю. Ю. , Иванова О. Г. , Данилкин С. В. Основы Web-инжиниринга : разработка клиентских приложений: учебное пособие Издательство ФГБОУ ВПО «ТГТУ», 2012. 240 стр.. URL: http://biblioclub.ru/index.php?page=book_red&id=277648&sr=1

6. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

RFC 2396 URI Generic Syntax <http://www.ietf.org/rfc/rfc2396.txt>

RFC 2616 HTTP/1.1 <http://www.ietf.org/rfc/rfc2616.txt>

The PHP Manual <http://php.net/docs.php>

Cascading Style Sheets, level 1 <http://www.w3.org/TR/CSS1/>

XMLHttpRequest <http://www.w3.org/TR/XMLHttpRequest/>

HTML 5 Specification <http://www.w3.org/TR/html5/>

7. Методические указания для обучающихся по освоению дисциплины

Задания 1-16 выполняются в виде веб-приложений PHP, работающих на сервере Apache. Тексты программ необходимо писать в кодировке UTF-8.

Индивидуальное задание выполняется на любом языке программирования. Также необходимо подготовить доклад на выбранную тему 2-3 страницы.

Объем реферата 2-3 страницы. Во время выступления (10-15 минут) необходима действующая демонстрация выбранной технологии.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

8.1 Перечень информационных технологий.

- Проверка домашних заданий и консультирование посредством электронной почты и социальной сети Вконтакте.
- Использование электронных презентаций при проведении практических занятий.

8.2 Перечень необходимого программного обеспечения.

Notepad2,
telnet,
Putty,
Virtual Box,
Debian/GNU Linux,
Apache,
PHP,
Firefox.
Eclipse IDE + плагины RSE и PDT или другая аналогичная IDE.
Программное обеспечение для безопасного отображения презентаций

8.3 Перечень информационных справочных систем:

1. Справочно-правовая система «Консультант Плюс» (<http://www.consultant.ru>)
2. Электронная библиотечная система eLIBRARY.RU (<http://www.elibrary.ru/>)

9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

№	Вид работ	Материально-техническое обеспечение дисциплины и оснащенность
1.	Лекционные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
2.	Лабораторные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, проектором, программным обеспечением
3.	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
4.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.