

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

**КАФЕДРА АСЦТ**

«Согласована»  
Председатель Методического совета  
факультета МИТ  
Доцент, канд. п.и.к.: \_\_\_\_\_ Д. Зулпукарова

«\_\_» \_\_\_\_\_ 2020 г.

«Утверждена»  
на заседании кафедры Программирования  
от 27 августа 2020 года, протокол №1  
Зав. каф. АСЦТ, доцент

Молдоярв У.Д. \_\_\_\_\_

# **УЧЕБНО-МЕТОДИЧЕСКИЙ КОМПЛЕКС**

**по дисциплине «Программирование Web-  
ориентированных приложений»**

для магистрантов очного отделения, обучающихся по специальности:  
710100 «Информатика и вычислительная техника»

**Составитель:**

**доцент Молдоярв У.Д.**

2020-2021 – учебный год

## **Данные о преподавателе:**

Молдояров Уларбек Дуйшобекович – доцент кафедры АСЦТ факультета Математика и информационных технологий ОшГУ

Стаж работы – 15 лет.

Образование:

- Высшее, ОшГУ, физико-математический факультет, 2005 г.;
- Магистр, ОшГУ, факультет математики и информационных технологий, 2016 г.

Рабочее место: 723500. главный корпус ОшГУ, ул. Ленина 331, кабинет – 205.

Контактная информация:

Моб. телефон: 0772843108, 0554490100

E-mail: ular\_osh@list.ru

Дежурство преподавателя проводится в кабинете 205 по пятницам с 14.00 до 16.00

## Оглавление

<b>I. АННОТАЦИЯ К ДИСЦИПЛИНЕ “УПРАВЛЕНИЕ ДАННЫМИ”</b> .....	4
Внутренняя и внешняя рецензии на рабочую программу.....	6
<b>II. РАБОЧАЯ ПРОГРАММА</b> .....	8
1. Цели освоения дисциплины .....	10
2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений" .....	10
4. Карта компетенций дисциплины в разрезе тем (разделов) .....	11
5. Технологическая карта дисциплины .....	12
6. Карта накапливаемости баллов по дисциплине .....	12
7. Тематический план распределения часов по видам занятий .....	14
8. Программа дисциплины .....	16
9. Календарно-тематический план по видам занятий.....	16
10. Разработка проекта.....	20
11. Образовательные технологии.....	21
12. Учебно-методическое и информационное обеспечение дисциплины .....	21
13. Политика выставления баллов .....	22
<b>III. СИЛЛАБУС</b> .....	26
1. Цели освоения дисциплины .....	<b>Ошибка! Закладка не определена.</b>
2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений" .....	28
5. Технологическая карта дисциплины .....	28
6. Карта накапливаемости баллов по дисциплине .....	30
7. Тематический план распределения часов по видам занятий .....	32
8. Программа дисциплины .....	32
9. Календарно-тематический план по видам занятий.....	33
10. Разработка проекта.....	<b>Ошибка! Закладка не определена.</b>
11. Образовательные технологии.....	36
12. Учебно-методическое и информационное обеспечение дисциплины .....	38
13. Политика выставления баллов .....	39
14. Политика курса.....	41
<b>IV. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ</b> .....	43
1. Общие положения .....	44
2. Перечень компетенций и этапы их формирования.....	44
3. Критерии оценки на экзамене .....	53
<b>V. ГЛОССАРИЙ (СЛОВАРЬ)</b> .....	54
<b>VI. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ МАГИСТРАНТОВ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ</b> .....	59
<b>VII. КОНСПЕКТЫ ЛЕКЦИЙ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ</b> .....	61
<b>VIII. ЛАБОРАТОРНЫЕ МАТЕРИАЛЫ</b> .....	66

## **I. АННОТАЦИЯ К ДИСЦИПЛИНЕ “УПРАВЛЕНИЕ ДАННЫМИ”**

Дисциплина «Программирование Web-ориентированных приложений» призвана содействовать знакомству магистрантов с компьютерными телекоммуникациями и возможными подходами к разработке гипертекстовых документов, предназначенных для публикации в глобальной компьютерной сети Internet. Она важна с той точки зрения, что позволяет развивать способности магистрантов, связанные с общей культурой работы в глобальной сети. Курс закрепляет навыки работы с текстом и графикой, а также навыков программирования и проектирования и разработки информационных систем, являясь, таким образом, прямым продолжением курсов «Web программирование», «Разработка клиент-серверных прилож», «Проектирование АСОИ», «Базы данных», «Программирование» и многих других.

**Целью** освоения дисциплины является изучение современных направлений в вебпрограммировании: фреймворков (frameworks) для создания приложений, новейших технологий и инструментальных средств разработки веб-сайтов, а также современных СУБД для работы с данными сайтов.

**Основные задачи курса:**

- методы решения научно-технических и информационных задач;
- современные информационные технологии;
- изучение возможностей фреймворков (frameworks) для создания веб-приложений;
- изучение современных инструментальных средств и языков разработки веб-приложений;
- изучение современных СУБД для работы с данными сайтов.

В лекционной части курса рассматриваются общие принципы Web-конструирования. Изучение всех тем сопровождается иллюстрирующими примерами.

Лабораторные работы в компьютерных классах служат для индивидуальной работы магистрантов над учебными задачами и итоговым проектом с целью выработки и закрепления практических навыков Web-конструирования и Web-программирования.

**Результаты обучения дисциплины основывается на РО-8 в ООП:**

При работе в сфере интернет магистрант использовать современные информационные технологии, предусматривающие организацию баз и банков данных, построение схем документооборота и т.д.

**Формирующие компетенции:**

- способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности; (ОК-2)
- способен к дальнейшему самостоятельному обучению; (ИК-4)
- способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления; (ПК-6)
- способен разрабатывать проекты автоматизированных систем различного назначения, обосновывать выбор аппаратно-программных средств автоматизации и информатизации предприятий и организаций; (ПК-9)
- способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов; (ПК-13)

Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций

**Знать:**

- современные инструментальные средства разработки компонентов распределенной информационной системы для работы в сети Интернет;

**Уметь:**

- проектировать компоненты РИС с использованием современных технологий веб-программирования;

**Владеть:**

- навыками разработки веб-приложений с использованием баз данных в рамках заданной предметной области.

## Внутренняя и внешняя рецензии на рабочую программу

### РЕЦЕНЗИЯ

на рабочую программу по дисциплине "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению 710100 «Информатика и вычислительная техника»

Рабочая программа дисциплины "Программирование Web-ориентированных приложений" разработана в соответствии с требованиями ГОС ВПО КР, на основе бюллетеня №19 ОшГУ и ООП специальности по направлению 710100 «Информатика и вычислительная техника».

Рецензируемая программа предназначена для методического обеспечения учебной работы магистрантов очной формы обучения. Содержание представленной на рецензию рабочей учебной программы включает в себя следующие разделы: цели и задачи освоения дисциплины; место дисциплины в структуре ООП; требования к результатам освоения дисциплины; объем дисциплины и виды учебной работы; содержание дисциплины; библиотечно-информационные ресурсы; оценочные средства; материально-техническое обеспечение.

Содержание курса представлено шести разделами, которые в полной мере отражают необходимый объем изучаемого материала. По каждому разделу составлен перечень вопросов, рассмотрение которых позволит сформировать знания, умения и навыки, отвечающие требованиям ВПО КР.

Информация о видах и объеме учебной работы содержит перечень лабораторных работ и тематику лекционных занятий, призванных помочь магистранту получить теоретические знания и практические навыки создания, внедрения, функционирования, применения системы управления базами данных и информационных систем управления, обеспечивающих поддержку работы специалиста.

Программа соответствует всем требованиям бюллетеня №19 ОшГУ и ООП специальности по направлению 710100 «Информатика и вычислительная техника».

Рецензент, старший специалист отдела  
сетевого и технического сопровождения  
ОАО Коммерческий банк Кыргызстан

Асранов Б.Б.

## РЕЦЕНЗИЯ

на рабочую программу по дисциплине "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению 710100 «Информатика и вычислительная техника»

Рабочая программа дисциплины "Программирование Web-ориентированных приложений" разработана в соответствии с требованиями ВПО КР, на основе бюллетеня №19 ОшГУ и ООП специальности по направлению 710100 «Информатика и вычислительная техника».

Рабочая программа содержит: цели дисциплины, результаты обучения и компетенции, формируемые в процессе изучения дисциплины, место дисциплины в структуре ООП, карта компетенций дисциплины в разрезе тем, технологическая карта дисциплины, карта накопления баллов по дисциплине, тематический план распределения часов по видам занятий, программа дисциплины, цели и результаты обучения по темам (разделам) дисциплины, календарно-тематический план по видам занятий с указанием формируемых компетенций, часов, баллов, методов, оценочных средств, литературы, образовательные, технологии учебно-методическое обеспечение дисциплины и политика выставления баллов.

На основании вышеизложенного считаю, что рецензируемая программа полностью соответствует требованиям, предъявляемым к рабочей программе по дисциплине "Программирование Web-ориентированных приложений" и может быть рекомендована для обучения магистрантов по направлению 710100 «Информатика и вычислительная техника».

Рецензент, доцент  
кафедры ПИИБ факультета МИТ:

Эркебаев У.

## II. РАБОЧАЯ ПРОГРАММА

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ

ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА АСЦТ

«Согласована»  
Председатель Методического совета  
факультета МИТ доцент, канд.  
п.и.к.: \_\_\_\_\_ Д. Зулпукарова

«Утверждена»  
на заседании кафедры АСЦТ  
от 27 августа 2020 года,  
протокол №1 Зав. каф. АСЦТ, доцент

«\_\_» \_\_\_\_\_ 2020 г.

Молдоярлов У.Д. \_\_\_\_\_

## РАБОЧАЯ ПРОГРАММА

по дисциплине: "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению: 710100 «Информатика и вычислительная техника»

### Сетка часов по учебному плану

Программирование Web-ориентированных приложений	Всего	Ауд. зан.	Аудит.зан.		СРМ	Отчетность	
			Лекции	Лабор.		3 сем	3 сем
2 курс, 3 сем.	90ч (3 кр.)	36 ч.	20 ч.	16 ч.	56 ч.	РК-2	Экз

Рабочая программа составлена на основании ООП, утвержденной Ученым Советом факультета \_\_\_\_\_ протокол №\_\_ от \_\_\_\_\_ 20\_\_ г.

Составитель: доцент кафедры \_\_\_\_\_ Молдоярлов У. Д.

2020-2021 – учебный год



## СОДЕРЖАНИЕ

1. Цели освоения дисциплины .....	10
2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений" .....	10
3. Место дисциплины в структуре ООП .....	11
4. Карта компетенций дисциплины в разрезе тем (разделов).....	11
5. Технологическая карта дисциплины .....	12
6. Карта накапливаемости баллов по дисциплине .....	12
7. Тематический план распределения часов по видам занятий .....	14
8. Программа дисциплины .....	16
9. Цели и результаты обучения по темам дисциплины <b>Ошибка! Закладка не определена.</b>	
10. Календарно-тематический план по видам занятий.....	16
11. Образовательные технологии .....	20
12. Учебно-методическое и информационное обеспечение дисциплины.....	21
13. Политика выставления баллов.....	22

## 1. Цели освоения дисциплины

Дисциплина «Программирование Web-ориентированных приложений» призвана содействовать знакомству магистрантов с компьютерными телекоммуникациями и возможными подходами к разработке гипертекстовых документов, предназначенных для публикации в глобальной компьютерной сети Internet. Она важна с той точки зрения, что позволяет развивать способности магистрантов, связанные с общей культурой работы в глобальной сети. Курс закрепляет навыки работы с текстом и графикой, а также навыков программирования и проектирования и разработки информационных систем, являясь, таким образом, прямым продолжением курсов «Программирование Web-ориентированных приложений», «Web программирование», «Разработка клиент-серверных прилож», «Проектирование АСОИ», «Базы данных», «Программирование» и многих других.

**Целью** освоения дисциплины является изучение современных направлений в вебпрограммировании: фреймворков (frameworks) для создания приложений, новейших технологий и инструментальных средств разработки веб-сайтов, а также современных СУБД для работы с данными сайтов.

Основные задачи курса:

- методы решения научно-технических и информационных задач;
- современные информационные технологии;
- изучение возможностей фреймворков (frameworks) для создания веб-приложений;
- изучение современных инструментальных средств и языков разработки веб-приложений;
- изучение современных СУБД для работы с данными сайтов.
- 

В лекционной части курса рассматриваются общие принципы Web-конструирования. Изучение всех тем сопровождается иллюстрирующими примерами.

Лабораторные работы в компьютерных классах служат для индивидуальной работы магистрантов над учебными задачами и итоговым проектом с целью выработки и закрепления практических навыков Web-конструирования и Web-программирования.

## 2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений"

1. результате изучения дисциплины магистрант достигнет следующих **результатов обучения (РОд)**, соответствующих ожидаемым **результатам освоения образовательной программы (РОоп)** и заданным для дисциплины **компетенциям**:

Код РОоп и его формулировка	Код компетенции ООП и его формулировка	Код РО дисциплины (РОд) и его формулировка
<b>РО-8</b> Веб-активность: умеет создавать веб-приложения, создавать базы данных и работать с системами управления базами данных с использованием новейших технологий и клиент-серверных технологий в	<b>ОК-2</b> способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не	<b>Знать:</b> <ul style="list-style-type: none"><li>– современные инструментальные средства разработки компонентов распределенной информационной системы для работы в сети Интернет;</li></ul>

Интернете, создавать и реализовывать бизнес-проекты в Интернете, участвовать в существующих бизнес-проектах	связанных со сферой деятельности; <b>ИК-4</b> способен к дальнейшему самостоятельному обучению; <b>ПК-6</b> способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления; <b>ПК-9</b> способен разрабатывать проекты автоматизированных систем различного назначения, обосновывать выбор аппаратно-программных средств автоматизации и информатизации предприятий и организаций; <b>ПК-13</b> способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;	<b>Уметь:</b> <ul style="list-style-type: none"> <li>– проектировать компоненты РИС с использованием современных технологий веб-программирования;</li> </ul> <b>Владеть:</b> <ul style="list-style-type: none"> <li>– навыками разработки веб-приложений с использованием баз данных в рамках заданной предметной области.</li> </ul>
---	---	---

### 3. Место дисциплины в структуре ООП

Дисциплина "Программирование Web-ориентированных приложений" относится к дисциплинам базовой части цикла профессиональных дисциплин (БЗ), обеспечивающих теоретическую и практическую подготовку магистрантов в области 710100 «Информатика и вычислительная техника».

Изучение данной дисциплины базируется на дисциплинах базовой части цикла профессиональных дисциплин таких как "Технология разработки программного обеспечения".

Основные положения дисциплины должны быть использованы в дальнейшем при изучении дисциплин "Технологии программирования", "Технологии разработки мобильных приложений".

### 4. Карта компетенций дисциплины в разрезе тем (разделов)

Разделы, № и название темы	Компетенции					Σ общее кол-во комп-ций
	ОК-2	ИК-4	ПК-6	ПК-9	ПК-13	
<b>I модуль</b>						
<b>Тема 1.</b> Введение в библиотеке JQuery, технологию AJAX. Разработка мобильных веб-приложений		+		+		2
<b>Тема 2.</b> Интеграция и взаимодействие в сети Веб	+		+		+	3
<b>Тема 3.</b> Синдикация и агрегирование веб-контента	+	+		+	+	4
<b>Тема 4.</b> Веб-порталы. Классификация веб-	+		+		+	3

порталов						
<b>Тема 5.</b> Введение в Веб 2.0	+		+	+		3
<b>II модуль</b>						
<b>Тема 6.</b> Современные клиентские (Frontend) фреймворки для создания веб-приложений	+	+	+	+	+	5
<b>Тема 7.</b> Серверные (Backend) фреймворки создания сайтов	+	+	+	+	+	5
<b>Тема 8.</b> Изучение современных инструментальных сред и языков создания веб-приложений		+		+	+	3
<b>Тема 9.</b> Использование возможностей серверных СУБД для работы с данными веб-приложений	+		+	+	+	4
<b>Тема 10.</b> Приложения для социальных сетей	+	+		+		3

### 5. Технологическая карта дисциплины

Всего часов	Ауд. часов	СРМ	1 модуль				2 модуль				Итоговый контроль (ИК)					Всего	
			Ауд. часы		СРМ	РК-1	Ауд. часы		СРМ	РК-2	Лекция	Лаборатория	СРМ	Итог. контр. (ИК)	Дополн. балл (Д)		
			Лекция	Лаборатория			Лекция	Лаборатория									
90	36	54	10	8	26			10	8	28							
Баллы			30	30	30	<b>30</b>	30	30	30	<b>30</b>	30	30	30	<b>30</b>	<b>10</b>		
Итоги модулей и итогового контроля			M1=(Лек+Лаб+СРМ + РК1)/4				M2=(Лек+Лаб+СРМ + М+ РК2)/4				ИК=(Лек+Лаб+СРМ)/3, Экз=M1+M2+ИК+Д					<b>100</b>	

Ауд. – аудиторные, ТК – текущий контроль, РК – рубежный контроль, СРМ – самостоятельная работа магистрантов, ИК – итоговый контроль, Д – дополнительный балл, М – модуль, Лек – лекции, Лаб – лабораторные.

### 6. Карта накапливаемости баллов по дисциплине

I модуль						
№	Лекции		Лабораторные работы		СРМ	
	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11
<b>ТК-1</b>						
№1 тема	1	30		30	2	30
№2 тема	1	30		30	2	30
№3 тема	1	30	1	30	2	30
№4 тема	1	30	1	30	2	30
№5 тема	1	30	1	30	4	30
<b>Всего</b>	<b>10</b>	<b>30</b>	<b>3</b>	<b>30</b>	<b>12</b>	<b>30</b>
<b>ТК-2</b>						
№1 тема	1	30	1	30	2	30

№2 тема	1	30	1	30	4	30
№3 тема	1	30	1	30	2	30
№4 тема	1	30	1	30	4	30
№5 тема	1	30	1	30	2	30
<b>Всего</b>	<b>10</b>	<b>30</b>	<b>5</b>	<b>30</b>	<b>14</b>	<b>30</b>
<b>Итого</b>	<b>10</b>	<b>(TK1+ TK2)/2</b>	<b>8</b>	<b>(TK1+ TK2)/2</b>	<b>26</b>	<b>(TK1+TK2)/2</b>
		<b>30</b>		<b>30</b>		<b>30</b>

**I модуль**

Текущий контроль			PK1	M1=(Лек+Лаб+СРМ+PK1)/4
Лек.	Лаб.	СРМ		
30	30	30	30	30

**II модуль**

№	Лекции		Лабораторные работы		СРМ	
	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11
<b>TK-3</b>						
№1 тема	1	30		30	2	30
№2 тема	1	30		30	2	30
№3 тема	1	30	1	30	4	30
№4 тема	1	30	1	30	2	30
№5 тема	1	30	1	30	4	30
<b>Всего</b>	<b>10</b>	<b>30</b>	<b>3</b>	<b>30</b>	<b>14</b>	<b>30</b>
<b>TK-4</b>						
№1 тема	1	30	1	30	2	30
№2 тема	1	30	1	30	4	30
№3 тема	1	30	1	30	2	30
№4 тема	1	30	1	30	4	30
№5 тема	1	30	1	30	2	30
<b>Всего</b>	<b>10</b>	<b>30</b>	<b>5</b>	<b>30</b>	<b>14</b>	<b>30</b>
<b>Итого</b>	<b>10</b>	<b>(TK3+ TK4)/2</b>	<b>8</b>	<b>(TK3+ TK4)/2</b>	<b>28</b>	<b>(TK3+TK4)/2</b>
		<b>30</b>		<b>30</b>		<b>30</b>

**II модуль**

Текущий контроль			PK2	M1=(Лек+Лаб+СРМ+PK2)/4
Лек.	Лаб.	СРМ		
30	30	30	30	30

## 7. Тематический план распределения часов по видам занятий

Разделы, № и название темы	Всего	Лекции	Лаб. работа.	СРМ	Обр. технол огии	Оценоч. средства
<b>I модуль</b>						
<b>Тема 1.</b> Введение в библиотеке JQuery, технологию AJAX. Разработка мобильных веб-приложений	6	2		4	КОТ, ИТО, ИТ, ГрТ	Устный опрос, презентация, демонстрация
<b>Тема 2.</b> Интеграция и взаимодействие в сети Веб	8	2	2	6	КОТ, ИТО, ИТ, ГрТ	Устный опрос, презентация, демонстрация
<b>Тема 3.</b> Синдикация и агрегирование веб-контента	8	2	2	4	КОТ, ИТО, ИТ, ГрТ	Устный опрос, игры, презентация, демонстрация
<b>Тема 4.</b> Веб-порталы. Классификация веб-порталов	10	2	2	6	КОТ, ИТО, ИТ, ГрТ	Устный опрос, игры, презентация, демонстрация
<b>Тема 5.</b> Введение в Веб 2.0	10	2	2	6	КОТ, ИТО, ИТ, ГрТ	Устный опрос, презентация, демонстрация
<b>Итого Модуль 1:</b>	<b>42</b>	<b>10</b>	<b>8</b>	<b>26</b>		
<b>II модуль</b>						
<b>Тема 6.</b> Современные клиентские (Frontend) фреймворки для создания веб-приложений	6	2		4	КОТ, ИТО, ИТ, ГрТ	Устный опрос, презентация, демонстрация
<b>Тема 7.</b> Серверные (Backend) фреймворки создания сайтов	8	2	2	6	КОТ, ИТО, ИТ, ГрТ	Устный опрос, игры, презентация, демонстрация
<b>Тема 8.</b> Изучение современных инструментальных сред и языков создания веб-приложений	8	2	2	4	КОТ, ИТО, ИТ, ГрТ	Устный опрос, презентация, демонстрация
<b>Тема 9.</b> Использование возможностей серверных СУБД для работы с данными веб-приложений	10	2	2	6	КОТ, ИТО, ИТ, ГрТ	Устный опрос, презентация, демонстрация

<b>Тема 10.</b> Приложения для социальных сетей	10	2	2	6	КОТ, ИТО, ИТ, ГрТ	Устный опрос, игры, презентация, демонстрация
<b>Итого Модуль 2:</b>	<b>42</b>	<b>10</b>	<b>8</b>	<b>28</b>		
<b>Всего:</b>	<b>90</b>	<b>20</b>	<b>16</b>	<b>54</b>		

## 8. Программа дисциплины

Введение в технологию AJAX. Разработка мобильных веб-приложений. Интеграция и взаимодействие в сети Веб. Синдикация и агрегирование веб-контента. Веб-порталы. Классификация веб-порталов. Введение в Веб 2.0. Современные клиентские (Frontend) фреймворки для создания веб-приложений. Серверные (Backend) фреймворки создания сайтов. Изучение современных инструментальных сред и языков создания веб-приложений. Использование возможностей серверных СУБД для работы с данными веб-приложений. Приложения для социальных сетей.

## 9. Календарно-тематический план по видам занятий

№ и название темы	№ лекции, комп.	Наименование изучаемых вопросов	Лит-ра	Лекции	Лаб. работа.	СРМ	Обр. Технологии
<b>I модуль</b>							
<b>Тема 1.</b> Введение в библиотеке JQuery, технологию AJAX. Разработка мобильных веб-приложений	ИК-4, ПК-9	<p><i>План лекции:</i></p> <p>1.1. Введение в технологию AJAX.</p> <p>1.2. Создание интерактивных приложений на основе AJAX</p> <p>1.3. Разработка мобильных веб-приложений</p> <p><i>Контрольные вопросы:</i></p> <p>Есть ли способ скрыть страницу отправки метода jquery ajax?</p> <p>Как и из какого источника можно подключить библиотеки JQuery?</p> <p><i>План СРМ:</i></p> <p>Компоненты технологии AJAX.</p> <p>Методы обработки событий.</p> <p>Методы передачи данных.</p>	1,3	2		4	КОТ, ИТО, ИТ, ГрТ
<b>Тема 2.</b> Интеграция и взаимодействие в сети Веб	ОК-2, ПК-6, ПК-13	<p><i>План лекции:</i></p> <p>Веб-интеграция.</p> <p>Подходы к веб-интеграции.</p> <p>Интеграция на основе XML.</p> <p>Веб-сервисы.</p> <p><i>Контрольные вопросы:</i></p> <p>Стандарт DISCO.</p> <p>Спецификация UDDI.</p> <p><i>План лаб. работы:</i></p> <p>Разработка веб-службы в ASPNET</p> <p><i>План СРМ:</i></p> <p>Разработка веб-службы в ASP.NET. Создание прокси-сборки для веб-службы</p>	2 4	2	2	6	КОТ, ИТО, ИТ, ГрТ
<b>Тема 3.</b> Синдикация и агрегирование веб-	ОК-2, ИК-4, ПК-9,	<p><i>План лекции:</i></p> <p>Веб-синдикация.</p> <p>Веб-поток.</p>	1 3,4	2	2	4	КОТ, ИТО, ИТ,



контента	ПК-13	<p>Агрегатор потоков. Преимущества веб-потоков. RSS.</p> <p><i>Контрольные вопросы:</i> Веб-поток. Агрегатор потоков.</p> <p><i>План лаб. работы:</i> Примеры разработки RSS-источников и RSS-ридеров</p> <p><i>План СРМ:</i> Форматирование и преобразование XML документа с помощью CSS и XSL. XSLT преобразование XML документа</p>					ГрТ
<b>Тема 4.</b> Веб-порталы. Классификация веб-порталов	ОК-2, ПК-6, ПК-13	<p><i>План лекции:</i> Портал. Портлеты. Веб-портал. Классификация порталов.</p> <p><i>Контрольные вопросы:</i> Горизонтальные порталы. Вертикальные порталы. Корпоративные порталы.</p> <p><i>План лаб. работы:</i> Программная обработка XML документов с помощью XML DOM</p> <p><i>План СРМ:</i> Разработка проекта.</p>	1,2 4	2	2	6	КОТ, ИТО, ИТ, ГрТ
<b>Тема 5.</b> Введение в Веб 2.0	ОК-2, ПК-6, ПК-9	<p><i>План лекции:</i> Веб 2.0: ключевые технологии и недостатки. Облачные технологии. Сервисы Web 2.0</p> <p><i>Контрольные вопросы:</i> Сравнение способов организации исследовательской деятельности в условиях Веб 1.0. и Веб 2.0</p> <p><i>План лаб. работы:</i> Географические сетевые сервисы Web 2.0</p> <p><i>План СРМ:</i> Мэшапы: архитектура и классификация.</p>	1,2 4	2	2	6	КОТ, ИТО, ИТ, ГрТ
<b>II модуль</b>							

<p><b>Тема 6.</b> Современные клиентские (Frontend) фреймворки для создания веб-приложений</p>	<p>ОК-2, ИК-4, ПК-6, ПК-9 , ПК-13</p>	<p><i>План лекции:</i>  Основное назначение фреймворков при создании веб-приложений.  Разработка приложений, работающих на клиентской стороне браузера.  Обзор современных frontend фреймворков.</p> <p><i>Контрольные вопросы:</i>  Что нужно знать для frontend разработки?</p> <p><i>План лаб. работы:</i>  Разработка веб-приложений с использованием клиентских фреймворков (<i>Разработка проекта</i>)</p> <p><i>План СРМ:</i>  Использование фреймворки клиентской части для ускорения написания кода клиентской части.  Возможность создания различных анимации и одностраничных приложений.</p>	<p>1,2 4</p>	<p>2</p>		<p>4</p>	<p>КОТ, ИТО, ИТ, ГрТ</p>
<p><b>Тема 7.</b> Серверные (Backend) фреймворки создания сайтов</p>	<p>ОК-2, ИК-4, ПК-6, ПК-9 , ПК-13</p>	<p><i>План лекции:</i>  Связь серверных фреймворков с логикой приложения.  Ограниченность серверных фреймворков в своей функциональности.  Связь серверных фреймворков с базами данных приложений.  <i>Контрольные вопросы:</i>  Что нужно знать для backend разработки?</p> <p><i>План лаб. работы:</i>  Разработка веб-приложений с использованием серверных фреймворков (<i>Разработка проекта</i>)</p> <p><i>План СРМ:</i>  Ответственность бэкенд фреймворков за внутренне устройство приложения, за критически важные части приложения, без которых оно не сможет нормально работать.</p>	<p>1,2 3</p>	<p>2</p>	<p>2</p>	<p>6</p>	<p>КОТ, ИТО, ИТ, ГрТ</p>
<p><b>Тема 8.</b> Изучение современных инструментальных сред и языков создания веб-приложений</p>	<p>ИК-4, ПК-9 , ПК-13</p>	<p><i>План лекции:</i>  Язык PHP.  Работа в инструментальной среде JetBrains PhpStorm.</p> <p><i>Контрольные вопросы:</i></p>	<p>1,2 3</p>	<p>2</p>	<p>2</p>	<p>4</p>	<p>КОТ, ИТО, ИТ, ГрТ</p>

		<p>Типы данных в PHP. Функции языка PHP для работы с базами данных.</p> <p><i>План лаб. работы:</i> Создание серверного веб-приложения с помощью современных инструментальных средств (JetBrains PhpStorm)</p> <p><i>План СРМ:</i> Возможности редактора для веб-разработки - CodeLobster.</p>					
<p><b>Тема 9.</b> Использование возможностей серверных СУБД для работы с данными веб-приложений</p>	<p>ОК-2, ПК-6, ПК-9 , ПК-13</p>	<p><i>План лекции:</i> Особенности работы с базами данных веб-сайтов, возможности современных серверных СУБД, работающих в сети Интернет. Особенности создания базы данных MySQL и PostgreSQL.</p> <p><i>Контрольные вопросы:</i> Удаление таблиц базы данных. Удаление базы данных.</p> <p><i>План лаб. работы:</i> Создание однотабличной базы данных с использованием возможностей современных серверных СУБД (MySQL, PostgreSQL)</p> <p><i>План СРМ:</i> Добавление информации в базу данных. Поиск информации в базе данных.</p>	<p>1 3,4</p>	<p>2</p>	<p>2</p>	<p>6</p>	<p>КОТ, ИТО, ИТ, ГрТ</p>
<p><b>Тема 10.</b> Приложения для социальных сетей</p>	<p>ОК-2, ИК-4, ПК-9</p>	<p><i>План лекции:</i> Понятие "Социальный Веб". Социальное ПО. Программные системы, относящиеся к социальному ПО.</p> <p><i>Контрольные вопросы:</i> Проблемы масштабирования СУБД в мобильных приложениях</p> <p><i>План лаб. работы:</i> Исследование возможностей серверных фреймворков при создании сайтов</p> <p><i>План СРМ:</i> Фолксонмия. Семантическая веб-сеть. Онтология. Семантические веб-сервисы.</p>	<p>1 3,4</p>	<p>2</p>	<p>2</p>	<p>6</p>	<p>КОТ, ИТО, ИТ, ГрТ</p>

## 10. Разработка проекта.

Разработать серверное веб-приложение, реализующее работу с базой данных в сети Интернет. При создании сайта следует основываться на использовании фреймворков, а также современных интегрированных сред для создания веб-приложений на языке PHP (JetBrains PhpStorm). Для хранения данных веб-сайта следует использовать базу данных СУБД PostgreSQL. База данных должна быть нормализованной, содержать минимум 10 связанных таблиц. В каждой таблице должно быть минимум 15 строк записей.

При разработке сайта необходимо предусмотреть возможность работы с базой данных средствами веб-приложения: добавление и редактирование данных о товарах, добавление и чтение данных о заказах покупателей, формирование запросов и отчетов о покупках за период.

Варианты заданий

- 1) Бронирование номеров в гостинице;
- 2) Продажа билетов в театр;
- 3) Оформление договоров на аренду офисных помещений;
- 4) Продажа путевок в турагентство;
- 5) Регистрация транспортных средств;
- 6) Продажа билетов на авиарейсы;
- 7) Продажа фотоаппаратов в магазине;
- 8) Выдача книг в библиотеке;
- 9) Продажа автомобилей в автосалоне;
- 10) Учет ремонта оргтехники;
- 11) Продажа билетов на концерт;
- 12) Продажа билетов на теплоход;
- 13) Учет пациентов скорой помощи;
- 14) Учет выдачи кредитов клиентам банка;
- 15) Продажа стиральных машин в магазине бытовой техники;
- 16) Учет посещения студентов в университете;
- 17) Продажа билетов на железнодорожные перевозки;
- 18) Учет успеваемости студентов по дисциплинам;
- 19) Учет посещения студентов в общежитиях;
- 20) Введение карточек в больнице;
- 21) Учет продажи автомобилей;
- 22) Введение статистики оплаты обучающихся;
- 23) Учет лекарств в аптечном пункте;
- 24) Продажа оружия в оружейном магазине;
- 25) Учет выплаты заработной платы на предприятии.

При разработке веб-сайта связать между собой 5-15 страниц. В отчете к лабораторной работе использовать комментарии.

**Отчет к РП должен содержать следующие пункты:**

Титульный лист.

Постановка задачи.

Содержание.

Введение.

1. Теоретические предпосылки.

2. Основы серверных языков.
  3. Возможности использования СУБД.
- Заключение.
- Список использованных источников.
- Приложение А. Программный код.
- Приложение Б. Схема сайта.
- Приложение В. Даталогическая модель БД.

## 11. Образовательные технологии

**Групповые технологии (ГТ)** - обеспечение активности учебного процесса, достижение высокого уровня усвоения содержания. Особенности организации:

- магистранты на уроке делится на группы для решения конкретных учебных задач,
- каждая группа получает определённое задание и выполняет его сообща под руководством лидера группы или учителя; -задания в группе выполняются таким способом, который позволяет учитывать и оценивать индивидуальный вклад каждого члена группы,
- состав группы непостоянный, он подбирается с учётом того, чтобы могли реализовываться учебные возможности каждого члена группы, в зависимости от содержания и характера предстоящей работы.

**Игровые технологии (ИТ)** - цель ставится перед учащимися в форме игровой задачи, учебная деятельность подчиняется правилам игры, учебный материал используется в качестве средства игры, в учебную деятельность включается элемент соревнования, успешное выполнение задания связывается с игровым результатом.

Педагогические игры по характеру педагогического процесса подразделяются на группы:

- а) обучающие, тренировочные, контролирующие и обобщающие;
- б) познавательные, воспитательные, развивающие;
- в) репродуктивные, продуктивные, творческие.

**Информационные технологии обучения (ИТО)**- формирование умений работать с информацией, развитие коммуникативных способностей, подготовка личности «информационного общества», дать магистранту так много учебного материала, как только он может усвоить, формирование исследовательских умений, умений принимать оптимальные решения. Компьютер используется на всех этапах процесса обучения- при объяснении нового материала, при закреплении знаний.

## 12. Учебно-методическое и информационное обеспечение дисциплины

### 12.1. Основные литературы

[1. Android. Программирование для профессионалов / Б. Харди \[и др.\]. – 2-е изд. – СПб.: Питер, 2016. – 640 с.](#)

[2. Основы создания приложений // Developers \[Электронный ре- сурс\]. – Режим доступа: <https://developer.android.com/guide>. – Дата доступа: 11.08.2019.](#)

### 12.2. Дополнительные литературы

[3. Программирование под Android. / Медникс З. \[и др.\]. – 2-е изд. – СПб.: Питер, 2013. – 560 с.](#)

[4. Start Android: учебник по Android для начинающих и продвинутых // STARTANDROID \[Электронный ресурс\]. – Режим доступа: <https://startandroid.ru/ru/>. – Дата доступа: 11.08.2019.](#)

### 12.3. Материально-техническое и информационное обеспечение дисциплины

При освоении дисциплины для выполнения лабораторных работ необходимы персональные компьютеры с набором программного обеспечения: Adobe Photoshop, пакет Subline Text 3, OpenServer, web-браузер.

УМК по дисциплине в электронном виде находится в KelBil, AVN и кабинете кафедры АСЦТ.

#### 12.4. Методические указания магистрантам

**Изучение программы курса.** На лекциях преподаватель рассматривает вопросы программы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удается осветить в полном объеме, поэтому преподаватель, по своему усмотрению, некоторые вопросы выносит на самостоятельную работу магистрантов, рекомендуя ту или иную литературу.

Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. В случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в пункте 12.1. «Учебно-методическое обеспечение дисциплины».

В целом, на один час аудиторных занятий отводится один час самостоятельной работы.

**Лабораторные работы.** При изучении курса «Программирование Web-ориентированных приложений» необходимо выполнять и вовремя сдавать преподавателю индивидуальные лабораторные работы.

#### 13. Политика выставления баллов

Выставление оценок на экзаменах осуществляется на основе принципов объективности, справедливости, всестороннего анализа качества знаний магистрантов, и других положений, способствующих повышению надежности оценки знаний, обучающихся и устранению субъективных факторов.

Оценка знаний (академической успеваемости) магистранту осуществляется по 30 и 100 балльной системам (шкале) следующим образом:

30 балльная система	Баллы (Рейтинг)	Оценка по буквенной системе	Цифровой эквивалент оценки по GPA	Оценка по традиционной системе
26- 30	87 – 100	A	4,0	Отлично
24- 25	80–86	B	3,33	Хорошо
22- 23	74–79	C	3,0	
20- 21	68–73	D	2,33	Удовлетворительно
18- 19	61–67	E	2,0	
9-17	31 -60	FX	0	Неудовлетворительно
0- 8	0-30	F	0	

#### 1. Критерий оценки магистранта на лекционных занятиях Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен успешно обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

**Шкала оценивания 18 – 19 баллов:**

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет частично систематизировать, структурировать и аргументировать материал.

**Шкала оценивания 9 – 17 баллов:**

- магистрант не полностью понимает темы согласно цели пройденной лекции;
- магистрант не способен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

**Шкала оценивания 0– 18 баллов:**

- магистрант непонимает темы согласно цели пройденной лекции;
- магистрант не способен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

## **2. Критерий оценки магистранта на лабораторных занятиях**

**Шкала оценивания 26 – 30 баллов:**

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 24 – 25 баллов:**

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 22 – 23 баллов:**

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- неуверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 20 – 21 баллов:**

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 18 – 19 баллов:**

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;

- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 9 – 17 баллов:**

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 0 – 18 баллов:**

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- не может изложить свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

### **3. Критерий оценки магистранта на самостоятельных работах магистрантов**

**Шкала оценивания 26 – 30 баллов:**

- магистрант демонстрирует полное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 24 – 25 баллов:**

- магистрант демонстрирует значительное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 22 – 23 баллов:**

- магистрант демонстрирует частичное понимание проблемы;
- магистрант выполняет большинство требования, предъявляемые к заданию;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 20 – 21 баллов:**

- магистрант демонстрирует небольшое понимание проблемы;
- магистрант выполняет немногие требования, предъявляемые к заданию;
- неполностью излагает свое решение при выполнении лабораторной работы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 18 – 19 баллов:**

- магистрант демонстрирует непонимание проблемы;
- неполностью излагает свое решение при выполнении лабораторной работы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 9 – 17 баллов:**

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.



**Шкала оценивания 0 – 18 баллов:**

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- нет ответа, не было попытки решить задачу;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

### III. СИЛЛАБУС

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ

ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА АСЦТ

«Согласована»

Председатель Методического совета  
факультета МИТ доцент, канд.  
п.и.к.: \_\_\_\_\_ Д. Зулпукарова

«Утверждена»

на заседании кафедры АСЦТ  
от 27 августа 2020 года, протокол №1  
Зав. каф. АСЦТ, доцент

« \_\_\_\_ » \_\_\_\_\_ 2020 г.

Молдоярлов У.Д. \_\_\_\_\_

### СИЛЛАБУС

по дисциплине: "Программирование Web-ориентированных приложений" для магистрантов очного отделения, обучающихся по направлению: 710100 «Информатика и вычислительная техника»

#### *Сетка часов по учебному плану*

<u>Программирование Web-ориентированных приложений</u>	Всего	Ауд. зан.	Аудит.зан.		СРМ	Отчетность	
			Лекции	Лабор.		3 сем	3 сем
2 курс, 3 сем.	90 ч (3 кр.)	36 ч.	20 ч.	16 ч.	56 ч.	РК-2	Экз

Данные о преподавателе:

Молдоярлов Уларбек Дуйшобекович – доцент кафедры АСЦТ  
факультета Математика и информационных технологий ОшГУ

Стаж работы: 15 лет.

Рабочее место:

723500. Главный корпус ОшГУ, ул. Ленина 331, кабинет – 205.

Контактная информация:

Моб. телефон: 0772843108, 0554490100

E-mail: ular@oshsu.kg

Дежурство преподавателя проводится в кабинете 205 по пятницам с 14.00 до 16.00

Дата: 2020 -2021 учебный год

## СОДЕРЖАНИЕ

<u>1. Цели освоения дисциплины</u> .....	10
<u>2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений"</u> .....	10
<u>3. Место дисциплины в структуре ООП</u> .....	11
<u>4. Карта компетенций дисциплины в разрезе тем (разделов)</u> .....	11
<u>5. Технологическая карта дисциплины</u> .....	12
<u>6. Карта накапливаемости баллов по дисциплине</u> .....	12
<u>7. Тематический план распределения часов по видам занятий</u> .....	14
<u>8. Программа дисциплины</u> .....	16
<u>9. Цели и результаты обучения по темам дисциплины</u> . <b>Ошибка! Закладка не определена.</b>	
<u>10. Календарно-тематический план по видам занятий</u> .....	16
<u>11. Образовательные технологии</u> .....	20
<u>12. Учебно-методическое и информационное обеспечение дисциплины</u> .....	21
<u>13. Политика выставления баллов</u> .....	22
<u>14. Политика курса</u> .....	<b>Ошибка! Закладка не определена.</b>

## 1. Цели освоения дисциплины

Дисциплина «Программирование Web-ориентированных приложений» призвана содействовать знакомству магистрантов с компьютерными телекоммуникациями и возможными подходами к разработке гипертекстовых документов, предназначенных для публикации в глобальной компьютерной сети Internet. Она важна с той точки зрения, что позволяет развивать способности магистрантов, связанные с общей культурой работы в глобальной сети. Курс закрепляет навыки работы с текстом и графикой, а также навыков программирования и проектирования и разработки информационных систем, являясь, таким образом, прямым продолжением курсов «Программирование Web-ориентированных приложений», «Web программирование», «Разработка клиент-серверных прилож», «Проектирование АСОИ», «Базы данных», «Программирование» и многих других.

**Целью** освоения дисциплины является изучение современных направлений в вебпрограммировании: фреймворков (frameworks) для создания приложений, новейших технологий и инструментальных средств разработки веб-сайтов, а также современных СУБД для работы с данными сайтов.

Основные задачи курса:

- методы решения научно-технических и информационных задач;
- современные информационные технологии;
- изучение возможностей фреймворков (frameworks) для создания веб-приложений;
- изучение современных инструментальных средств и языков разработки веб-приложений;
- изучение современных СУБД для работы с данными сайтов.
- 

В лекционной части курса рассматриваются общие принципы Web-конструирования. Изучение всех тем сопровождается иллюстрирующими примерами.

Лабораторные работы в компьютерных классах служат для индивидуальной работы магистрантов над учебными задачами и итоговым проектом с целью выработки и закрепления практических навыков Web-конструирования и Web-программирования.

## 2. Результаты обучения и компетенции, формируемые в процессе изучения дисциплины "Программирование Web-ориентированных приложений"

2. результате изучения дисциплины магистрант достигнет следующих **результатов обучения (РОд)**, соответствующих ожидаемым **результатам освоения образовательной программы (РОоп)** и заданным для дисциплины **компетенциям**:

Код РОоп и его формулировка	Код компетенции ООП и его формулировка	Код РО дисциплины (РОд) и его формулировка
<b>РО-8</b> Веб-активность: умеет создавать веб-приложения, создавать базы данных и работать с системами управления базами данных с использованием новейших технологий и клиент-серверных технологий в	<b>ОК-2</b> способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не	<b>Знать:</b> <ul style="list-style-type: none"><li>– современные инструментальные средства разработки компонентов распределенной информационной системы для работы в сети Интернет;</li></ul>

<p>Интернете, создавать и реализовывать бизнес-проекты в Интернете, участвовать в существующих бизнес-проектах</p>	<p>связанных со сферой деятельности;  <b>ИК-4</b> способен к дальнейшему самостоятельному обучению;  <b>ПК-6</b> способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления;  <b>ПК-9</b> способен разрабатывать проекты автоматизированных систем различного назначения, обосновывать выбор аппаратно-программных средств автоматизации и информатизации предприятий и организаций;  <b>ПК-13</b> способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;</p>	<p><b>Уметь:</b></p> <ul style="list-style-type: none"> <li>– проектировать компоненты РИС с использованием современных технологий веб-программирования;</li> </ul> <p><b>Владеть:</b></p> <ul style="list-style-type: none"> <li>– навыками разработки веб-приложений с использованием баз данных в рамках заданной предметной области.</li> </ul>
--	--	---

### 3. Место дисциплины в структуре ООП

Дисциплина "Программирование Web-ориентированных приложений" относится к дисциплинам базовой части цикла профессиональных дисциплин (БЗ), обеспечивающих теоретическую и практическую подготовку магистрантов в области 710100 «Информатика и вычислительная техника».

Изучение данной дисциплины базируется на дисциплинах базовой части цикла профессиональных дисциплин таких как “Технология разработки программного обеспечения”.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении дисциплин “Технологии программирования”, “Технологии разработки мобильных приложений”.

### 5. Технологическая карта дисциплины

Всего часов	Ауд. часов	СРМ	1 модуль				2 модуль				Итоговый контроль (ИК)					Всего	
			Ауд.часы		СРМ	ПК-1	Ауд.часы		СРМ	ПК-2	Лекция	Лаборатория	СРМ	Итог.контр. (ИК)	Дополн. балл (П)		
			Лекция	Лаборатория			Лекция	Лаборатория									
90	36	54	10	8	26			10	8	28							
Баллы			30	30	30	<b>30</b>	30	30	30	<b>30</b>	30	30	30	<b>30</b>	<b>10</b>		
Итоги модулей и итогового			M1=(Лек+Лаб+СРМ +ПК1)/4				M2=(Лек+Лаб+СРМ +ПК2)/4				ИК=(Лек+Лаб+СРМ)/3, Экз=M1+M2+ИК+Д					<b>100</b>	

контроля				
----------	--	--	--	--

Ауд. – аудиторные, ТК – текущий контроль, РК – рубежный контроль, СРМ – самостоятельная работа магистрантов, ИК – итоговый контроль, Д – дополнительный балл, М – модуль, Лек – лекции, Лаб – лабораторные.

### 6. Карта накапливаемости баллов по дисциплине

I модуль						
№	Лекции		Лабораторные работы		СРМ	
	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11
<b>TK1</b>						
№1 тема	1	30		30	2	30
№2 тема	1	30		30	2	30
№3 тема	1	30	1	30	2	30
№4 тема	1	30	1	30	2	30
№5 тема	1	30	1	30	4	30
<b>Всего</b>	<b>10</b>	<b>30</b>	<b>3</b>	<b>30</b>	<b>12</b>	<b>30</b>
<b>TK2</b>						
№1 тема	1	30	1	30	2	30
№2 тема	1	30	1	30	4	30
№3 тема	1	30	1	30	2	30
№4 тема	1	30	1	30	4	30
№5 тема	1	30	1	30	2	30
<b>Всего</b>	<b>10</b>	<b>30</b>	<b>5</b>	<b>30</b>	<b>14</b>	<b>30</b>
<b>Итого</b>	<b>10</b>	$(TK1+TK2)/2$	<b>8</b>	$(TK1+TK2)/2$	<b>26</b>	$(TK1+TK2)/2$
		<b>30</b>		<b>30</b>		<b>30</b>

### I модуль

Текущий контроль			РК1	M1=(Лек+Лаб+СРМ+РК1)/4
Лек.	Лаб.	СРМ		
30	30	30	30	30

### II модуль

№	Лекции		Лабораторные работы		СРМ	
	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11	Часы	Балл (№1 тема+№2 тема+...+№11 тема)/11
<b>TK3</b>						
№1 тема	1	30		30	2	30
№2 тема	1	30		30	2	30
№3 тема	1	30	1	30	4	30
№4 тема	1	30	1	30	2	30
№5 тема	1	30	1	30	4	30

<b>Всего</b>	<b>10</b>	<b>30</b>	<b>3</b>	<b>30</b>	<b>14</b>	<b>30</b>
<b>TK4</b>						
<b>№1 тема</b>	1	30	1	30	2	30
<b>№2 тема</b>	1	30	1	30	4	30
<b>№3 тема</b>	1	30	1	30	2	30
<b>№4 тема</b>	1	30	1	30	4	30
<b>№5 тема</b>	1	30	1	30	2	30
<b>Всего</b>	<b>10</b>	<b>30</b>	<b>5</b>	<b>30</b>	<b>14</b>	<b>30</b>
<b>Итого</b>	<b>10</b>	$(TK3+ TK4)/2$	<b>8</b>	$(TK3+ TK4)/2$	<b>28</b>	$(TK3+TK4)/2$
		<b>30</b>		<b>30</b>		<b>30</b>

**II модуль**

<b>Текущий контроль</b>			<b>PK2</b>	<b>M1=(Лек+Лаб+СРМ+PK2)/4</b>
<b>Лек.</b>	<b>Лаб.</b>	<b>СРМ</b>		
30	30	30	30	30

## 7. Тематический план распределения часов по видам занятий

Разделы, № и название темы	Всего	Лекции	Лаб. работа.	СРМ
<b>I модуль</b>				
<b>Тема 1.</b> Введение в библиотеке JQuery, технологию AJAX. Разработка мобильных веб-приложений	6	2		4
<b>Тема 2.</b> Интеграция и взаимодействие в сети Веб	8	2	2	6
<b>Тема 3.</b> Синдикация и агрегирование веб-контента	8	2	2	4
<b>Тема 4.</b> Веб-порталы. Классификация веб-порталов	10	2	2	6
<b>Тема 5.</b> Введение в Веб 2.0	10	2	2	6
<b>Итого Модуль 1:</b>	<b>42</b>	<b>10</b>	<b>8</b>	<b>26</b>
<b>II модуль</b>				
<b>Тема 6.</b> Современные клиентские (Frontend) фреймворки для создания веб-приложений	6	2		4
<b>Тема 7.</b> Серверные (Backend) фреймворки создания сайтов	8	2	2	6
<b>Тема 8.</b> Изучение современных инструментальных сред и языков создания веб-приложений	8	2	2	4
<b>Тема 9.</b> Использование возможностей серверных СУБД для работы с данными веб-приложений	10	2	2	6
<b>Тема 10.</b> Приложения для социальных сетей	10	2	2	6
<b>Итого Модуль 2:</b>	<b>42</b>	<b>10</b>	<b>8</b>	<b>28</b>
<b>Всего:</b>	<b>90</b>	<b>20</b>	<b>16</b>	<b>54</b>

## 8. Программа дисциплины

Введение в технологию AJAX. Разработка мобильных веб-приложений. Интеграция и взаимодействие в сети Веб. Синдикация и агрегирование веб-контента. Веб-порталы. Классификация веб-порталов. Введение в Веб 2.0. Современные клиентские (Frontend) фреймворки для создания веб-приложений. Серверные (Backend) фреймворки создания сайтов. Изучение современных инструментальных сред и языков создания веб-приложений. Использование возможностей серверных СУБД для работы с данными веб-приложений. Приложения для социальных сетей.



## 9. Календарно-тематический план по видам занятий

№ и название темы	№ лекции, комп.	Наименование изучаемых вопросов	Лит-ра	Лекции	Лаб. работа.	СРМ	Обр. Технол огни
<b>I модуль</b>							
<b>Тема 1.</b> Введение в библиотеке JQuery, технологию AJAX. Разработка мобильных веб-приложений	ИК-4, ПК-9	<p><i>План лекции:</i></p> <p>1.1. Введение в технологию AJAX.</p> <p>1.2. Создание интерактивных приложений на основе AJAX</p> <p>1.3. Разработка мобильных веб-приложений</p> <p><i>Контрольные вопросы:</i></p> <p>Есть ли способ скрыть страницу отправки метода jquery ajax?</p> <p>Как и из какого источника можно подключить библиотеки JQuery?</p> <p><i>План СРМ:</i></p> <p>Компоненты технологии AJAX.</p> <p>Методы обработки событий.</p> <p>Методы передачи данных.</p>	1,3	2		4	КОТ, ИТО, ИТ, ГрТ
<b>Тема 2.</b> Интеграция и взаимодействие в сети Веб	ОК-2, ПК-6, ПК-13	<p><i>План лекции:</i></p> <p>Веб-интеграция.</p> <p>Подходы к веб-интеграции.</p> <p>Интеграция на основе XML.</p> <p>Веб-сервисы.</p> <p><i>Контрольные вопросы:</i></p> <p>Стандарт DISCO.</p> <p>Спецификация UDDI.</p> <p><i>План лаб. работы:</i></p> <p>Разработка веб-службы в ASPNET</p> <p><i>План СРМ:</i></p> <p>Разработка веб-службы в ASP.NET. Создание прокси-сборки для веб-службы</p>	2 4	2	2	6	КОТ, ИТО, ИТ, ГрТ
<b>Тема 3.</b> Синдикация и агрегирование веб-контента	ОК-2, ИК-4, ПК-9, ПК-13	<p><i>План лекции:</i></p> <p>Веб-синдикация.</p> <p>Веб-поток.</p> <p>Агрегатор потоков.</p> <p>Преимущества веб-потоков. RSS.</p> <p><i>Контрольные вопросы:</i></p> <p>Веб-поток.</p> <p>Агрегатор потоков.</p> <p><i>План лаб. работы:</i></p> <p>Примеры разработки RSS-источников и RSS-ридеров</p> <p><i>План СРМ:</i></p>	1 3,4	2	2	4	КОТ, ИТО, ИТ, ГрТ

		Форматирование и преобразование XML документа с помощью CSS и XSL. XSLT преобразование XML документа					
<b>Тема 4.</b> Веб-порталы. Классификация веб-порталов	ОК-2, ПК-6, ПК-13	<p><i>План лекции:</i> Портал. Портлеты. Веб-портал. Классификация порталов.</p> <p><i>Контрольные вопросы:</i> Горизонтальные порталы. Вертикальные порталы. Корпоративные порталы.</p> <p><i>План лаб. работы:</i> Программная обработка XML документов с помощью XML DOM</p> <p><i>План СРМ:</i> Разработка проекта.</p>	1,2 4	2	2	6	КОТ, ИТО, ИТ, ГрТ
<b>Тема 5.</b> Введение в Веб 2.0	ОК-2, ПК-6, ПК-9	<p><i>План лекции:</i> Веб 2.0: ключевые технологии и недостатки. Облачные технологии. Сервисы Web 2.0</p> <p><i>Контрольные вопросы:</i> Сравнение способов организации исследовательской деятельности в условиях Веб 1.0. и Веб 2.0</p> <p><i>План лаб. работы:</i> Географические сетевые сервисы Web 2.0</p> <p><i>План СРМ:</i> Мэшапы: архитектура и классификация.</p>	1,2 4	2	2	6	КОТ, ИТО, ИТ, ГрТ
<b>II модуль</b>							
<b>Тема 6.</b> Современные клиентские (Frontend) фреймворки для создания веб-приложений	ОК-2, ИК-4, ПК-6, ПК-9, ПК-13	<p><i>План лекции:</i> Основное назначение фреймворков при создании веб-приложений. Разработка приложений, работающих на клиентской стороне браузера. Обзор современных frontend фреймворков.</p> <p><i>Контрольные вопросы:</i> Что нужно знать для frontend разработки?</p> <p><i>План лаб. работы:</i> Разработка веб-приложений с использованием клиентских фреймворков (<i>Разработка проекта</i>)</p>	1,2 4	2		4	КОТ, ИТО, ИТ, ГрТ

		<p><i>План СРМ:</i> Использование фреймворки клиентской части для ускорения написания кода клиентской части. Возможность создания различных анимации и одностраничных приложений.</p>					
<p><b>Тема 7.</b> Серверные (Backend) фреймворки создания сайтов</p>	<p>ОК-2, ИК-4, ПК-6, ПК-9 , ПК-13</p>	<p><i>План лекции:</i> Связь серверных фреймворков с логикой приложения. Ограниченность серверных фреймворков в своей функциональности. Связь серверных фреймворков с базами данных приложений. <i>Контрольные вопросы:</i> Что нужно знать для backend разработки?</p> <p><i>План лаб. работы:</i> Разработка веб-приложений с использованием серверных фреймворков (<i>Разработка проекта</i>)</p> <p><i>План СРМ:</i> Ответственность бэкенд фреймворков за внутренне устройство приложения, за критически важные части приложения, без которых оно не сможет нормально работать.</p>	<p>1,2 3</p>	<p>2</p>	<p>2</p>	<p>6</p>	<p>КОТ, ИТО, ИТ, ГрТ</p>
<p><b>Тема 8.</b> Изучение современных инструментальных сред и языков создания веб-приложений</p>	<p>ИК-4, ПК-9 , ПК-13</p>	<p><i>План лекции:</i> Язык PHP. Работа в инструментальной среде JetBrains PhpStorm.</p> <p><i>Контрольные вопросы:</i> Типы данных в PHP. Функции языка PHP для работы с базами данных.</p> <p><i>План лаб. работы:</i> Создание серверного веб-приложения с помощью современных инструментальных средств (JetBrains PhpStorm)</p>	<p>1,2 3</p>	<p>2</p>	<p>2</p>	<p>4</p>	<p>КОТ, ИТО, ИТ, ГрТ</p>

		<i>План СРМ:</i> Возможности редактора для веб-разработки - CodeLobster.					
<b>Тема 9.</b> Использование возможностей серверных СУБД для работы с данными веб-приложений	ОК-2, ПК-6, ПК-9, ПК-13	<i>План лекции:</i> Особенности работы с базами данных веб-сайтов, возможности современных серверных СУБД, работающих в сети Интернет. Особенности создания базы данных MySQL и PostgreSQL.  <i>Контрольные вопросы:</i> Удаление таблиц базы данных. Удаление базы данных.  <i>План лаб. работы:</i> Создание однотабличной базы данных с использованием возможностей современных серверных СУБД (MySQL, PostgreSQL)  <i>План СРМ:</i> Добавление информации в базу данных. Поиск информации в базе данных.	1 3,4	2	2	6	КОТ, ИТО, ИТ, ГрТ
<b>Тема 10.</b> Приложения для социальных сетей	ОК-2, ИК-4, ПК-9	<i>План лекции:</i> Понятие "Социальный Веб". Социальное ПО. Программные системы, относящиеся к социальному ПО.  <i>Контрольные вопросы:</i> Проблемы масштабирования СУБД в мобильных приложениях  <i>План лаб. работы:</i> Исследование возможностей серверных фреймворков при создании сайтов  <i>План СРМ:</i> Фолксономия. Семантическая веб-сеть. Онтология. Семантические веб-сервисы.	1 3,4	2	2	6	КОТ, ИТО, ИТ, ГрТ

## 10. Разработка проекта.

Разработать серверное веб-приложение, реализующее работу с базой данных в сети Интернет. При создании сайта следует основываться на использовании фреймворков, а также современных интегрированных сред для создания веб-приложений на языке PHP (JetBrains PhpStorm). Для хранения данных веб-сайта следует использовать базу данных СУБД PostgreSQL. База данных должна быть нормализованной, содержать минимум 10 связанных таблиц. В каждой таблице должно быть минимум 15 строк записей.

При разработке сайта необходимо предусмотреть возможность работы с базой данных средствами веб-приложения: добавление и редактирование данных о товарах, добавление и чтение данных о заказах покупателей, формирование запросов и отчетов о покупках за период.

Варианты заданий

- 1) Бронирование номеров в гостинице;
- 2) Продажа билетов в театр;
- 3) Оформление договоров на аренду офисных помещений;
- 4) Продажа путевок в турагентство;
- 5) Регистрация транспортных средств;
- 6) Продажа билетов на авиарейсы;
- 7) Продажа фотоаппаратов в магазине;
- 8) Выдача книг в библиотеке;
- 9) Продажа автомобилей в автосалоне;
- 10) Учет ремонта оргтехники;
- 11) Продажа билетов на концерт;
- 12) Продажа билетов на теплоход;
- 13) Учет пациентов скорой помощи;
- 14) Учет выдачи кредитов клиентам банка;
- 15) Продажа стиральных машин в магазине бытовой техники;
- 16) Учет посещения студентов в университете;
- 17) Продажа билетов на железнодорожные перевозки;
- 18) Учет успеваемости студентов по дисциплинам;
- 19) Учет посещения студентов в общежитиях;
- 20) Введение карточек в больнице;
- 21) Учет продажи автомобилей;
- 22) Введение статистики оплаты обучающихся;
- 23) Учет лекарств в аптечном пункте;
- 24) Продажа оружия в оружейном магазине;
- 25) Учет выплаты заработной платы на предприятии.

При разработке веб-сайта связать между собой 5-15 страниц. В отчете к лабораторной работе использовать комментарии.

**Отчет к РП должен содержать следующие пункты:**

Титульный лист.

Постановка задачи.

Содержание.

Введение.

1. Теоретические предпосылки.

2. Основы серверных языков.

3. Возможности использования СУБД.

Заключение.

Список использованных источников.

Приложение А. Программный код.

Приложение Б. Схема сайта.

Приложение В. Даталогическая модель БД.

## 11. Образовательные технологии

**Групповые технологии (ГТ)** - обеспечение активности учебного процесса, достижение высокого уровня усвоения содержания. Особенности организации:

- магистранты на уроке делится на группы для решения конкретных учебных задач,
- каждая группа получает определённое задание и выполняет его сообща под руководством лидера группы или учителя; задания в группе выполняются таким способом, который позволяет учитывать и оценивать индивидуальный вклад каждого члена группы,
- состав группы непостоянный, он подбирается с учётом того, чтобы могли реализовываться учебные возможности каждого члена группы, в зависимости от содержания и характера предстоящей работы.

**Игровые технологии (ИТ)** - цель ставится перед учащимися в форме игровой задачи, учебная деятельность подчиняется правилам игры, учебный материал используется в качестве средства игры, в учебную деятельность включается элемент соревнования, успешное выполнение задания связывается с игровым результатом.

Педагогические игры по характеру педагогического процесса подразделяются на группы:

- а) обучающие, тренировочные, контролирующие и обобщающие;
- б) познавательные, воспитательные, развивающие;
- в) репродуктивные, продуктивные, творческие.

**Информационные технологии обучения (ИТО)**- формирование умений работать с информацией, развитие коммуникативных способностей, подготовка личности «информационного общества», дать магистранту так много учебного материала, как только он может усвоить, формирование исследовательских умений, умений принимать оптимальные решения. Компьютер используется на всех этапах процесса обучения- при объяснении нового материала, при закреплении знаний.

## 12. Учебно-методическое и информационное обеспечение дисциплины

### 12.1. Основные литературы

1. Томас А. Пауэлл. Ajax. Настольная книга программиста. –М: Эксмо, 2017. - 720 стр.

2. [Основы создания приложений // Developers \[Электронный ре- сурс\]. – Режим доступа: https://developer.android.com/guide.](https://developer.android.com/guide) – Дата доступа: 11.08.2019.

### 12.2. Дополнительные литературы

3. [Программирование под Android. / Медникс З. \[и др.\]. – 2-е изд. – СПб.: Питер, 2013. – 560 с.](#)

4. [Start Android: учебник по Android для начинающих и продвинутых // STARTANDROID \[Электронный ресурс\]. – Режим доступа: https://startandroid.ru/ru/.](https://startandroid.ru/ru/) – Дата доступа: 11.08.2019.

### 12.3. Материально-техническое и информационное обеспечение дисциплины

При освоении дисциплины для выполнения лабораторных работ необходимы персональные компьютеры с набором программного обеспечения: Adobe Photoshop, пакет Subline Text 3, web-браузер.

УМК по дисциплине в электронном виде находится в KelBil, AVN и кабинете кафедры АСЦТ.

### 12.4. Методические указания магистрантам

**Изучение программы курса.** На лекциях преподаватель рассматривает вопросы программы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удастся осветить в полном объеме, поэтому преподаватель, по своему усмотрению, некоторые вопросы выносит на самостоятельную работу магистрантов, рекомендуя ту или иную литературу.

Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. В

случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в пункте 12.1. «Учебно-методическое обеспечение дисциплины».

В целом, на один час аудиторных занятий отводится один час самостоятельной работы.

**Лабораторные работы.** При изучении курса «Программирование Web-ориентированных приложений» необходимо выполнять и вовремя сдавать преподавателю индивидуальные лабораторные работы.

### 13. Политика выставления баллов

Выставление оценок на экзаменах осуществляется на основе принципов объективности, справедливости, всестороннего анализа качества знаний магистрантов, и других положений, способствующих повышению надежности оценки знаний, обучающихся и устранению субъективных факторов.

Оценка знаний (академической успеваемости) магистранту осуществляется по 30 и 100 балльной системам (шкале) следующим образом:

30 балльная система	Баллы (Рейтинг)	Оценка по буквенной системе	Цифровой эквивалент оценки по GPA	Оценка по традиционной системе
26- 30	87 – 100	A	4,0	Отлично
24- 25	80–86	B	3,33	Хорошо
22- 23	74–79	C	3,0	
20- 21	68–73	D	2,33	Удовлетворительно
18- 19	61–67	E	2,0	
9-17	31 -60	FX	0	Неудовлетворительно
0- 8	0-30	F	0	

#### 1. Критерий оценки магистранта на лекционных занятиях Шкала оценивания 26 – 30 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен успешно обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 24 – 25 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 22 – 23 баллов:

- магистрант демонстрирует полное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 20 – 21 баллов:

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 18 – 19 баллов:

- магистрант демонстрирует неполное понимание темы согласно цели пройденной лекции;
- магистрант способен неполностью обосновывать свою точку зрения;
- умеет частично систематизировать, структурировать и аргументировать материал.

#### Шкала оценивания 9 – 17 баллов:

- магистрант не полностью понимает темы согласно цели пройденной лекции;
- магистрант не способен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

**Шкала оценивания 0– 18 баллов:**

- магистрант непонимает темы согласно цели пройденной лекции;
- магистрант неспособен обосновывать свою точку зрения;
- не умеет систематизировать, структурировать и аргументировать материал.

**2. Критерий оценки магистранта на лабораторных занятиях****Шкала оценивания 26 – 30 баллов:**

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 24 – 25 баллов:**

- магистрант демонстрирует полное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно выполняет поставленную задачу;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 22 – 23 баллов:**

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- неуверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 20 – 21 баллов:**

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант самостоятельно и правильно не может выполнить поставленную задачу;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 18 – 19 баллов:**

- магистрант демонстрирует неполное понимание цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 9 – 17 баллов:**

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- не соблюдает правила техники безопасности при выполнении лабораторной работы.

**Шкала оценивания 0 – 18 баллов:**

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;



- не может изложить свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

### **3. Критерий оценки магистранта на самостоятельных работах магистрантов**

#### **Шкала оценивания 26 – 30 баллов:**

- магистрант демонстрирует полное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

#### **Шкала оценивания 24 – 25 баллов:**

- магистрант демонстрирует значительное понимание проблемы;
- магистрант выполняет все требования, предъявляемые к заданию;
- неполностью уверенно, логично, последовательно и аргументировано излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

#### **Шкала оценивания 22 – 23 баллов:**

- магистрант демонстрирует частичное понимание проблемы;
- магистрант выполняет большинство требований, предъявляемые к заданию;
- неуверенно излагает свое решение при выполнении лабораторной работы, используя понятия профессиональной сферы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

#### **Шкала оценивания 20 – 21 баллов:**

- магистрант демонстрирует небольшое понимание проблемы;
- магистрант выполняет немногие требования, предъявляемые к заданию;
- неполностью излагает свое решение при выполнении лабораторной работы;
- соблюдает правила техники безопасности при выполнении лабораторной работы.

#### **Шкала оценивания 18 – 19 баллов:**

- магистрант демонстрирует непонимание проблемы;
- неполностью излагает свое решение при выполнении лабораторной работы;
- частично соблюдает правила техники безопасности при выполнении лабораторной работы.

#### **Шкала оценивания 9 – 17 баллов:**

- магистрант не понимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- неуверенно излагает свое решение при выполнении лабораторной работы;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

#### **Шкала оценивания 0 – 18 баллов:**

- магистрант непонимает цели поставленной лабораторной работы;
- магистрант не понимает поставленную задачу;
- нет ответа, не было попытки решить задачу;
- несоблюдает правила техники безопасности при выполнении лабораторной работы.

### **14. Политика курса**

Основные требования к компонентам курса и его изучению:

- Магистрант должен посещать занятия, принимать активное участие в работе группы при выполнении СРМ и на лабораторных занятиях.
- На лекционных занятиях делать записи содержания лекций, внимательно слушать, не нарушая дисциплину.
- На практическом занятии важно не только выступать, но и внимательно слушать своих сокурсников, оценивать их ответы, вести запись новой информации.

- Не опаздывать, в аудиторию входить до звонка.
- Отключать мобильные телефоны.
- Не перебивать преподавателя и своих сокурсников в ходе беседы или при чтении лекции.

## **IV. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ  
РЕСПУБЛИКИ**

**ОШСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ФАКУЛЬТЕТ МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ**

**КАФЕДРА АСЦТ**

«Согласована»

Председатель Методического совета  
факультета МИТ доцент, канд.  
п.и.к.: \_\_\_\_\_ Д. Зулпукарова

« \_\_\_\_ » \_\_\_\_\_ 2020 г.

«Утверждена»

на заседании кафедры АСЦТ  
от 27 августа 2020 года, протокол №1  
Зав. каф. АСЦТ, доцент

доцент  
Молдоярлов У.Д. \_\_\_\_\_

## **ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

по дисциплине: "Программирование Web-ориентированных  
приложений" для магистрантов очного отделения, обучающихся  
по направлению: 710100 «Информатика и вычислительная техника»

Составитель: доцент кафедры

Молдоярлов У.Д.

## 1. Общие положения

### 1.1. Область применения фонда оценочных средств (ФОС)

ФОС по дисциплине является частью программы подготовки специалистов среднего звена по специальности 710100 «Информатика и вычислительная техника»

## 2. Перечень компетенций и этапы их формирования

### Компетенции обучающегося, формируемые в результате освоения дисциплины

1. ОК-2: способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности;
2. ИК-4: способен к дальнейшему самостоятельному обучению;
3. ПК-6: способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления;
4. ПК-9: способен разрабатывать проекты автоматизированных систем различного назначения, обосновывать выбор аппаратно-программных средств автоматизации и информатизации предприятий и организаций;
5. ПК-13: способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;

### Этапы формирования компетенций

<b>Тема 1.</b> Введение в библиотеке JQuery, технологию AJAX. Разработка мобильных веб-приложений	ИК-4, ПК-9
<b>Тема 2.</b> Интеграция и взаимодействие в сети Веб	ОК-2, ПК-6, ПК-13
<b>Тема 3.</b> Синдикация и агрегирование веб-контента	ОК-2, ИК-4, ПК-9, ПК-13
<b>Тема 4.</b> Веб-порталы. Классификация веб-порталов	ОК-2, ПК-6, ПК-13
<b>Тема 5.</b> Введение в Веб 2.0	ОК-2, ПК-6, ПК-9
<b>Тема 6.</b> Современные клиентские (Frontend) фреймворки для создания веб-приложений	ОК-2, ИК-4, ПК-6, ПК-9, ПК-13
<b>Тема 7.</b> Серверные (Backend) фреймворки создания сайтов	ОК-2, ИК-4, ПК-6, ПК-9, ПК-13
<b>Тема 8.</b> Изучение современных инструментальных сред и языков создания веб-приложений	ИК-4, ПК-9, ПК-13
<b>Тема 9.</b> Использование возможностей серверных СУБД для работы с данными веб-приложений	ОК-2, ПК-6, ПК-9, ПК-13
<b>Тема 10.</b> Приложения для социальных сетей	ОК-2, ИК-4, ПК-9

В результате освоения дисциплины «Программирование Web-ориентированных приложений» формируется компетенция ОК-2: способен самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой деятельности; ИК-4: способен к дальнейшему самостоятельному обучению; ПК-6: способен к постановке и модернизации отдельных лабораторных работ и практикумов по дисциплинам направления; ПК-9: способен разрабатывать проекты автоматизированных систем различного назначения, обосновывать выбор аппаратно-

программных средств автоматизации и информатизации предприятий и организаций; ПК-13: способен проектировать и применять инструментальные средства реализации программно-аппаратных проектов;

**Показатели, критерии и шкала оценивания компетенций текущего контроля знаний по учебной дисциплине «Программирование Web-ориентированных приложений»**

Текущий контроль знаний, согласно положению о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся (далее Положение) в рамках изучения дисциплины «Программирование Web-ориентированных приложений» предполагает тестирование, выполнение заданий по лабораторным работам и выполнение заданий по практическим работам.

## ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ ЗНАНИЙ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ “ПРОГРАММИРОВАНИЕ WEB- ОРИЕНТИРОВАННЫХ ПРИЛОЖЕНИЙ”

### 1. Текущий контроль (ТК) № 1

**Тема занятия:** 1.1. Работа с файлами

**Метод и форма контроля:** Практическая работа (Информационно-аналитический)

**Вид контроля:** Практическая работа (по вариантам)

#### Задание №1

1. Создать страницу, на которой можно загрузить несколько фотографий в галерею. Все загруженные фото должны помещаться в папку и выводиться на странице в виде таблицы
2. Создать гостевую книгу, где любой человек может оставить комментарий в текстовом поле и добавить его. Все добавленные комментарии выводятся над текстовым поле

<i>Балл</i>	<i>Показатели оценки</i>
3	Создана страница, оформлена с помощью стилей CSS.
4	Создана страница или гостевая книга, оформленные с помощью стилей CSS.
8	Созданы страница и гостевая книга, оформленные с помощью стилей CSS.

#### Задание №2

Разработать динамическое мини Web приложение

- создать шаблон (каркас) сайта добавить стили
- создать файл с навигацией по сайт
- подключить к основному шаблону (каркасу) нашего сайта.

<i>Балл</i>	<i>Показатели оценки</i>
3	Создано примитивное (с малым количеством разделов и переходов) динамическое Web приложение
4	Создано динамическое Web приложение с небольшими недочетами в переходах между разделами
8	Создано полноценно-работающее динамическое мини Web приложение

### 2. Текущий контроль (ТК) № 2

**Тема занятия:** 2. Защита Web-приложений

**Метод и форма контроля:** Письменный опрос (Опрос)

**Вид контроля:** Практическая работа (по вариантам)

#### Задание №1

Сформулируйте ответы по "Безопасность Web- приложений"

1. Что такое SQL инъекции?
2. На какие два вида делятся HTML инъекции?
3. Перечислите 22 вида уязвимостей веб сайтов.
4. Дайте характеристику 10 видам уязвимостей веб сайтов.
5. Назовите виды сетевых атак.

<i>Балл</i>	<i>Показатели оценки</i>
2	Сформулированы ответы на три вопроса на выбор.
3	Сформулированы ответы на четыре вопроса на выбор, включая обязательно четвертый вопрос.
5	Сформулированы все ответы, приведены примеры.

#### Задание №2

Ответить на вопросы:

1. Что такое Web приложение?
2. В чем особенность работы Web-сервера? Приведите примеры.
3. Перечислите и раскройте функции прокси-сервера.

<i>Балл</i>	<i>Показатели оценки</i>
2	Представлен ответ на 1 вопрос
3	Представлены ответы на 2 вопроса
5	Представлены ответы на 3 вопроса

### Задание №1

Ответить на вопросы:

1. Дайте определение масштабируемости
2. Что такое оптимизация и чем она отличается от масштабируемости
3. Перечислите способы масштабируемости их охарактерисуйте их.

<i>Оценка</i>	<i>Показатели оценки</i>
2	Представлено общее понятие масштабируемости. Перечислены виды.
3	Представлено общее понятие масштабируемости. Перечислены виды.
	Дана характеристика оптимизации и отличия ее от масштабируемости. Приведены примеры масштабируемости.
5	Представлено общее понятие масштабируемости. Перечислены виды.
	Дана характеристика оптимизации и отличия ее от масштабируемости. Приведены примеры масштабируемости.
	Как происходит масштабируемость БД. Дано определение механизма партиционирование, шардинг, масштабируемость ФС и масштабируемость кода (краткая характеристика)

### 3. Текущий контроль (ТК) № 3

#### Задание №1 (из текущего контроля)

1. Создать страницу, на которой можно загрузить несколько фотографий в галерею. Все загруженные фото должны помещаться в папку и выводиться на странице в виде таблицы
2. Создать гостевую книгу, где любой человек может оставить комментарий в текстовом поле и добавить его. Все добавленные комментарии выводятся над текстовым поле

<i>Балл</i>	<i>Показатели оценки</i>
3	Создана страница, оформлена с помощью стилей CSS.
4	Создана страница или гостевая книга, оформленные с помощью стилей CSS.
8	Созданы страница и гостевая книга, оформленные с помощью стилей CSS.

#### Задание №2

Разработать один из фрагментов динамического web-приложения:

1. Создать гостевую книгу, где любой человек может оставить комментарий в текстовом поле и добавить его.  
Все добавленные комментарии выводятся над текстовым полем
- 2 Создать страницу, на которой можно загрузить несколько фотографий в галерею. Все загруженные фото должны помещаться в папку uploads и выводиться на странице в виде таблицы.

<i>Балл</i>	<i>Показатели оценки</i>
3	Разработать 1 фрагмент динамического web-приложения, в котором комментарий не выводятся над текстовым полем.
4	Разработать 1 фрагмент динамического web-приложения, в котором комментарии выводятся над текстовым полем.
8	Разработать 2 фрагмент динамического web-приложения, в котором выполняются все требования.

## 2.4 Текущий контроль (ТК) № 4

**Тема занятия:** 2.3.6.Разработка статических страниц с использованием CSS3

**Метод и форма контроля:** Практическая работа (Информационно-аналитический)

**Вид контроля:** Практическая работа

### Занятие(-я):

2.3. Разработка статических страниц с использованием CSS

2.3.1.Введение в CSS3.

2.3.2. Введение в CSS3

2.3.3. Стиль CSS3: Фон. Цвет. Градиент.

2.3.4.Стиль CSS3: Шрифт. Текст. Прозрачность.

2.3.5. Стиль CSS3: Трансформация. Переходы. Анимация.

### Задание №1

Создать документ содержащий элементы, оформленные при помощи стилей CSS:

1. Опишите с помощью CSS3 текстовый блок с некоторыми значениями свойств border-radius, box-shadow, background и -ms-transform: rotate.
2. Добавьте линейные градиенты с помощью CSS3.
3. Добавьте эффекты перехода используя функции:  
linear  
ease  
ease-in  
ease-out  
ease-in-out  
cubic-bezier(x,x,x,x)

Балл	Показатели оценки
5	Представленный документ содержит все задания, освещающие пункты с CSS3, но не все свойства и функции.
10	Представленный документ содержит все задания, освещающие пункты с CSS3, но не функции.
15	Представленный документ содержит все задания, освещающие пункты с CSS3.



## ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ РУБЕЖНОГО КОНТРОЛЯ

### Рубежный контроль (РК) № 1

**Метод и форма контроля:** Практическая работа (Опрос)

**Вид контроля:** выполнить одно теоретическое задание (по вариантам) и одно практическое задание (по вариантам предметной области)

#### Задание №1

Создать HTML-документ, который будет содержать в себе базовую структуру, doctype, head, html, body.

Требования:

1. Вывести в созданном документе заглавия - h1, h2, h3
2. Вывести жирный, курсив и подчеркнутый текст
3. Вывести текст в абзаце
4. Вывести ссылки на внешний ресурс. Первая ссылка должна открываться в текущей вкладке браузера, а вторая ссылка, должна открывать внешний ресурс в новой вкладке.
5. Создать блочный и строчный тег (div, span)
6. Вывести таблицу, где в ней будет 5 столбцов: ID, имя, фамилия, отчество, дата рождения. Таблицу заполнить 2 записями
7. Создайте переменную name и присвойте ей значение, содержащее ваше имя, например "Василий"
8. Создайте переменную age и присвойте ей значение, содержащее ваш возраст, например 21.
9. Выведите с помощью echo фразу "Меня зовут: %ваше\_имя%", например: "Меня зовут: Василий".
10. Выведите фразу "Мне %ваш\_возраст% лет", например: "Мне 21 лет"
11. Напишите конструкцию if, которая выводит фразу: "Вам еще работать и работать" при условии, что значение переменной age попадает в диапазон чисел от 18 до 59 (включительно).
12. Расширьте конструкцию if из предыдущего пункта, выводя фразу: "Вам пора на пенсию" при условии, что значение переменной age больше 60.
13. Расширьте конструкцию if из предыдущего пункта, выводя фразу: "Вам еще рано работать" при условии, что значение переменной age попадает в диапазон чисел от 0 до 17 (включительно)
14. Расширьте конструкцию if из из предыдущего пункта, выводя фразу: "Неизвестный возраст" при условии, что значение переменной age является отрицательным числом, или вовсе числом не является.
15. Создать HTML-форму. Внутри формы создать поля для ввода: имени, фамилии, отчества, даты рождения, пароля и подтверждения пароля. Также, ко всем полям ввода нужно привязать метки label, в которых будет выведено имя поля. Метки должны находиться с левой стороны от полей ввода. Форма должна содержать поле-флаг, который будет нужен для подтверждения лицензионного соглашения и радио-кнопки для выбора гендерной принадлежности. Внизу формы должна быть кнопка для отправки введенных данных.
16. Вывести список элементов, построчно: яблоко, груша, персик. Список должен быть нумерованным.
17. Вывести выпадающий список элементов: яблоко, груша, персик. По умолчанию должна быть выбрана груша.
18. Нарисовать горизонтальную линию с шириной линии 150px. Ширину задать с помощью CSS правил (в атрибуте style).
19. Вывести изображение, которое должно быть сжато до 200px по ширине. Также у картинки должен быть заполнен атрибут альтернативного имени.

<i>Оценка</i>	<i>Показатели оценки</i>
3	Выполнено 9 требований в задании на выбор.
4	Выполнено 13 требований в задании на выбор.
5	Выполнено все требования в задании.

## Задание №2

Ответить на вопросы:

1. Дайте определение масштабируемости
2. Что такое оптимизация и чем она отличается от масштабируемости
3. Перечислите способы масштабируемости их охарактерисуйте их.

<i>Балл</i>	<i>Показатели оценки</i>
2	Представлено общее понятие масштабируемости. Перечислены виды.
3	Представлено общее понятие масштабируемости. Перечислены виды.
	Дана характеристика оптимизации и отличия ее от масштабируемости. Приведены примеры масштабируемости.
5	Представлено общее понятие масштабируемости. Перечислены виды.
	Дана характеристика оптимизации и отличия ее от масштабируемости. Приведены примеры масштабируемости.
	Как происходит масштабируемость БД. Дано определение механизма партиционирование, шардинг, масштабируемость ФС и масштабируемость кода (краткая характеристика)

## Задание №3

Выполнить задание:

1. Как сделать ссылку на телефонный номер, скайп или адрес электронной почты?
2. Как сделать гиперссылки на сайте? Приведите 5 способов с примерами.
3. Как создать таблицу, строки (ряды) таблицы и сделать ячейку заголовка столбца таблицы. Приведите примеры.

<i>Балл</i>	<i>Показатели оценки</i>
2	Даны ответы на 2 вопроса, но без примеров.
3	Даны ответы на 2 вопроса с примерами.
5	Даны развернуты ответы на 3 вопроса с примерами.

## Рубежный контроль (РК) № 2

**Метод и форма контроля:** Практическая работа

**Вид контроля:** выполнить практическое задание

## Задание №4

Создать HTML-документ, который будет содержать в себе базовую структуру, doctype, head, html, body.

Требования:

1. Вывести в созданном документе заглавия - h1, h2, h3
2. Вывести жирный, курсив и подчеркнутый текст
3. Вывести текст в абзаце
4. Вывести ссылки на внешний ресурс. Первая ссылка должна открываться в текущей вкладке браузера, а вторая ссылка, должна открывать внешний ресурс в новой вкладке.
5. Создать блочный и строчный тег (div, span)
6. Вывести таблицу, где в ней будет 5 столбцов: ID, имя, фамилия, отчество, дата рождения. Таблицу заполнить 2 записями
7. Создайте переменную name и присвойте ей значение, содержащее ваше имя, например "Василий"
8. Создайте переменную age и присвойте ей значение, содержащее ваш возраст, например 21.
9. Выведите с помощью echo фразу "Меня зовут: %ваше\_имя%", например: "Меня зовут: Василий".
10. Выведите фразу "Мне %ваш\_возраст% лет", например: "Мне 21 лет"
11. Напишите конструкцию if, которая выводит фразу: "Вам еще работать и работать" при условии, что значение переменной age попадает в диапазон чисел от 18 до 59 (включительно).

12. Расширьте конструкцию if из предыдущего пункта, выводя фразу: "Вам пора на пенсию" при условии, что значение переменной age больше 60.
13. Расширьте конструкцию if из предыдущего пункта, выводя фразу: "Вам еще рано работать" при условии, что значение переменной age попадает в диапазон чисел от 0 до 17 (включительно)
14. Расширьте конструкцию if из из предыдущего пункта, выводя фразу: "Неизвестный возраст" при условии, что значение переменной age является отрицательным числом, или вовсе числом не является.
15. Создать HTML-форму. Внутри формы создать поля для ввода: имени, фамилии, отчества, даты рождения, пароля и подтверждения пароля. Также, ко всем полям ввода нужно привязать метки label, в которых будет выведено имя поля. Метки должны находиться с левой стороны от полей ввода. Форма должна содержать поле-флаг, который будет нужен для подтверждения лицензионного соглашения и радио-кнопки для выбора гендерной принадлежности. Внизу формы должна быть кнопка для отправки введенных данных.
16. Вывести список элементов, построчно: яблоко, груша, персик. Список должен быть нумерованным.
17. Вывести выпадающий список элементов: яблоко, груша, персик. По умолчанию должна быть выбрана груша.
18. Нарисовать горизонтальную линию с шириной линии 150px. Ширину задать с помощью CSS правил (в атрибуте style).
19. Вывести изображение, которое должно быть сжато до 200px по ширине. Также у картинки должен быть заполнен атрибут альтернативного имени.

<i>Балл</i>	<i>Показатели оценки</i>
2	Выполнено 9 требований в задании на выбор.
3	Выполнено 13 требований в задании на выбор.
5	Выполнено все требования в задании.

#### **Задание №5 (из текущего контроля)**

1. Напишите правильно способы подключения JavaScript, заполнив пропуски. Приведите примеры.
  1. `cript>`
  2. `cript>`
2. Для чего используются следующие функции и метод, заполните пропуски:
  - `document.write()` для.....
  - `document.writeln()` для .....
  - `alert()` для .....
3. Ответьте на вопросы
  1. Перечислите 3 требования к Идентификаторам (identifiers) — имена переменных, методов и объектов:
  2. Перечислите функции преобразования данных.
  3. Какие методы для вывода модальных окон в javascript Вы знаете?
  4. Какой метод позволяет вывести модальное окно для ввода данных?

<i>Балл</i>	<i>Показатели оценки</i>
2	Даны ответы на 4 вопроса
3	Даны ответы на 4 вопроса и заполнены пропуски в функциях и методах
5	Даны ответы на 4 вопроса и заполнены пропуски в функциях и методах и способах подключения. Приведены примеры в пером задании.

#### **Задание №6 (из текущего контроля)**

Сформулируйте ответы по "Безопасность Web- приложений"

1. Что такое SQL инъекции?
2. На какие два вида делятся HTML инъекции?
3. Перечислите 22 вида уязвимостей веб сайтов.
4. Дайте характеристику 10 видам уязвимостей веб сайтов.
5. Назовите виды сетевых атак.

<b>Балл</b>	<b>Показатели оценки</b>
2	Сформулированы ответы на три вопроса на выбор.
3	Сформулированы ответы на четыре вопроса на выбор, включая обязательно четвертый вопрос.
5	Сформулированы все ответы, приведены примеры.

### Карта накапливаемости баллов по дисциплине

#### І модуль

Текущий контроль (ТК1+ТК2)/2	РК1	$M1=(TK1+TK2+PK1)/4$
30 балл	30 балл	30 балл

#### ІІ модуль

Текущий контроль (ТК3+ТК4)/2	РК2	$M2=(TK3+TK4+PK2)/4$
30 балл	30 балл	30 балл

### 3. Критерии оценки на экзамене

Выставление оценок на экзаменах осуществляется на основе принципов объективности, справедливости, всестороннего анализа качества знаний магистрантов, и других положений, способствующих повышению надежности оценки знаний обучающихся и устранению субъективных факторов.

В соответствии с действующими нормативными актами и рекомендациями Министерства образования и науки КР устанавливаются следующие критерии выставления оценок на экзаменах по гуманитарным, естественным, техническим и другим дисциплинам:

- оценка **"отлично"** выставляется магистранту, который обнаружил на экзамене всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, который усвоил основную литературу и ознакомился с дополнительной литературой, рекомендованной программой. Как правило, оценка "отлично" выставляется магистрантам, усвоившим взаимосвязь основных понятий дисциплины и их значений для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;

- оценка **"хорошо"** выставляется магистранту, который на экзамене обнаружил полное знание учебно-программного материала, успешно выполнил предусмотренные в программе задания, усвоил основную литературу, рекомендованную в программе. Как правило, оценка "хорошо" выставляется магистрантам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному выполнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности;

- оценка **"удовлетворительно"** выставляется магистранту, обнаружившему знание основного учебного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющемуся с выполнением заданий, предусмотренных программой, который ознакомился с основной литературой, рекомендованной программой. Как правило, оценка "удовлетворительно" выставляется магистрантам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя;

- оценка **"неудовлетворительно"** выставляется магистранту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не ознакомившемуся с основной литературой, предусмотренной программой, и не овладевшему базовыми знаниями, предусмотренными по данной дисциплине и определенными соответствующей программой курса (перечень основных знаний и умений, которыми должны овладеть магистранты, является обязательным элементом рабочей программы курса).

## V. ГЛОССАРИЙ (СЛОВАРЬ)

### **Апплет**

Апплет — это программный компонент (программный код), который загружается браузером и реализует определенные функции на веб-странице. Апплеты Java обеспечивают интерактивные функции в веб-браузерах с помощью Java Virtual Machine (JVM).

### **Установить/Выбрать**

Один из методов, с помощью которого пользователи вводят данные в диалоговом окне или на веб-странице, либо установив флажок с помощью курсора, либо переместив курсор на графическую селективную кнопку и щелкнув на ней.

### **Диалоговое окно**

Специальное окно в графическом интерфейсе пользователя, которое создает уведомления для пользователя определенного ресурса или запрашивает входные данные.

### **Включить/Отключить**

Доступные для пользователя настройки программного обеспечения, которые включают или отключают определенные функции или возможности. Используется для устранения ошибок конфигурации Java, когда пользователям может потребоваться использовать диалоговое окно для включения или отключения определенной настройки или функции.

### **Брандмауэр**

Интернет-брандмауэры могут быть как программными, так и аппаратными. Они защищают компьютер от внешних интернет-атак, которые могут представлять угрозу интернет-безопасности и повредить файлы на компьютере. В брандмауэре можно задавать правила, которые разрешают или запрещают передачу интернет-трафика.

### **Архив Java (JAR)**

Архив Java (.jar) — это формат файла, используемый для объединения нескольких файлов в один архивный файл. Как правило, JAR-файл содержит файлы класса и вспомогательные ресурсы, ассоциированные с апплетами и приложениями.

### **Подключаемый модуль Java**

Технология подключаемого модуля Java является частью среды выполнения Java, которая подключает популярные браузеры к платформе Java. С помощью такого подключения возможен запуск апплетов на веб-сайтах и их отображение в окне установленного на компьютере браузера.

### **Кэш среды выполнения Java (JRE)**

Область хранения на консоли Java, которую иногда необходимо очищать вручную, чтобы обеспечить возможность загрузки и установки последней версии Java.

### **Виртуальная машина Java (JVM)**

В Java набор программ, обеспечивающих поддержку выполнения инструкций, обычно написанных на Java. JVM доступны для большинства аппаратных и программных платформ.

### **javaw.exe**

Процесс javaw.exe - программа, принадлежащая корпорации Oracle, которая используется в браузере Internet Explorer в качестве подключаемого модуля Java. Она является аналогом программы java.exe. Единственным отличием является то, что при работе процесса javaw.exe не отображается окно консоли. Если вы не хотите, чтобы отображалось окно командной строки, вполне вероятно, что может использоваться процесс javaw.exe. javaw.exe является исполняемым файлом, отображающим диалоговое окно при сбоях запуска программы.

### **jucheck.exe**

Процесс jucheck.exe устанавливается вместе с Java на компьютер под управлением Windows и используется для проверки новых версий Java. Процесс не устанавливает Java самостоятельно, а только предупреждает о доступных для загрузки новых версиях. » Дополнительная информация

### **IFTW**

Установка через Интернет, или интерактивная установка.

### **Процедуры установки**

Пользователи Java могут выбрать наиболее подходящую из трех процедур установки – интерактивную, автономную или ручную.

Интерактивная установка выполняется автоматически при наличии подключения к Интернету после нажатия кнопки "Загрузить Java бесплатно" на странице Загрузить Java бесплатно.

Для установки в автономном режиме необходимо загрузить исполняемый файл, доступный на странице ручной загрузки Java где можно найти все файлы, необходимые для полной установки среды по усмотрению пользователя. Во время данной процедуры установки подключение к Интернету не требуется. Данный файл также можно скопировать и установить на другой компьютер, не подключенный к Интернету.

Во время установки вручную загружается исполняемый программный файл IFTW (Install From The Web). Такой режим установки требует минимального участия пользователя. При запуске этой программы все необходимые файлы загружаются из Интернета, поэтому во время данной процедуры установки требуется постоянное подключение к Интернету.

### **Манифест**

Манифест — это специальный файл, который может содержать информацию о файлах, упакованных в JAR-файле.

### **Операционная система**

Общий термин, обозначающий программное обеспечение, которое используется для управления основными задачами ресурсов компьютера и которое используется программистами для получения доступа к этим ресурсам. Наиболее

распространенные настольные операционные системы включают в себя Linux, Mac OS X, Microsoft Windows и Solaris.

### **Параметр**

В программировании параметром называется значение, которое передается функции. Функция либо использует параметр в соответствующей задаче, либо выполняет операцию с параметром.

### **Физическая память**

Наиболее часто на сайте java.com этот термин указывает на форму полупроводниковой памяти на компьютерах, которая называется оперативной памятью (ОЗУ).

### **Процесс**

Общий термин, который используется для описания программы, запущенной для выполнения определенной задачи – часто одновременно с другими программами.

### **Прокси-сервер**

Компьютер, выступающий посредником при подключении пользователя к сети Интернет. Он используется для ведения журнала сеансов подключения к Интернету и для блокирования доступа к отдельным веб-сайтам. Брандмауэр, устанавливаемый на прокси-сервере, может по разным причинам блокировать отдельные веб-сайты и веб-страницы. По этой причине пользователи могут испытывать затруднения при попытке загрузить Java или запустить те или иные Java-апплеты без настройки параметров определенного прокси-сервера в веб-браузере компьютера.

### **Настройки прокси-сервера**

Правильно настроенные параметры прокси-сервера обеспечивают для пользователей возможность подключения к сети Интернет при использовании прокси-сервера. Как правило, пользователям необходимо связаться со своими администраторами сети для получения необходимых сведений для настройки параметров прокси-сервера.

### **Кэш веб-браузера**

Кэшем называется область временного хранения, где можно сохранить часто используемые данные для быстрой загрузки. Кэш веб-браузера содержит копии страниц, просматриваемых пользователем. Для устранения определенных проблем установки или настройки Java иногда требуется выполнять очистку кэша вручную путем обращения к диалоговому окну.

### **Временное решение**

Временным решением обычно называется решение, которое позволяет обойти определенную проблему в системе до появления решения, которое позволит окончательно устранить эту ошибку.

### **Безопасность**

Список сайтов-исключений

Функция "Exception Site List" позволяет конечным пользователям выполнять



апплеты Java и приложения Java Web Start, которые не соответствуют последним требованиям к безопасности. RIA-приложения, которые размещаются на сайте из списка исключений, разрешается выполнять с применимыми запросами безопасности.

### **Базовый уровень безопасности**

Базовый уровень безопасности — это минимальное рекомендуемое обновление для Java. Пользователям не рекомендуется запускать версии Java, уровень которых ниже последнего базового уровня безопасности. Каждая основная версия Java (например, Java 6, Java 7) может иметь свое собственное обновление базового уровня безопасности.

### **Исправление**

Означает дополнительные изменения для установленного программного обеспечения. Может включать в себя исправления проблем, связанных с общей производительностью и системой безопасности ПО.

### **Критическое обновление (CPU)**

Критические обновления представляют собой регулярно выпускаемые и общедоступные наборы исправлений системы безопасности, предназначенные для снижения затрат клиентов, связанных с управлением безопасностью.

### **Предупреждение безопасности**

В случае крайней необходимости применения исправлений до выхода следующего пакета критических обновлений, Oracle выпускает предупреждение безопасности (т.е. исправление системы безопасности, распространяемое ранее очередного выпуска критического обновления исправлений).

### **Общие уязвимости и внешние воздействия (CVE)**

Номера CVE являются уникальными стандартными идентификаторами для определения общеизвестной информации об уязвимостях безопасности. Программа CVE спонсируется отделом обеспечения кибербезопасности и связи департамента США по национальной безопасности и управляется корпорацией MITRE.

### **Центр сертификации (CA)**

Центром сертификации называется доверенная третья сторона - обычно, коммерческая компания, - которая выпускает цифровые сертификаты. Сертификаты выдаются организациям и частным лицам после процесса удостоверения подлинности. Цифровой сертификат добавляется к компьютерным приложениям для подтверждения того, что приложение выпущено владельцем сертификата.

### **Доверенный сертификат**

Если у приложения имеется сертификат доверенного центра сертификации, отображается синий значок щита вместо желтых предупреждающих сигналов (щита или значка треугольника). Удостоверьтесь, что приложение получено от надежного издателя, при помощи сертификата, выпущенного доверенным центром сертификации.

### **Самоподписанный сертификат**

Самоподписанные сертификаты не удостоверяются доверенным центром сертификации. Использование приложений этого типа обеспечивает самый высокий уровень угрозы безопасности из-за отсутствия удостоверения личности издателя.

### **Приложение "из песочницы"**

Обозначает уровень доступа к системным файлам, таким как файлы на жестком диске, и к сети. Песочница обеспечивает защиту от доступа недоверенных и, возможно, вредоносных приложений к ресурсам системы.

### **Подписанные и неподписанные приложения**

Подписанный код - это технология безопасности, при использовании которой приложения снабжаются цифровой подписью, удостоверяющей автора или издателя программного обеспечения. Подписанными приложениями называются такие приложения, которые были подписаны поставщиком приложения (самоподписанные) или центром сертификации. Это не является гарантией того, что код является доверенным, а означает только то, что приложение получено от указанного источника. Неподписанными приложениями называются такие приложения, которые не были подписаны, поэтому их источник не может быть проверен.

## **VI. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ МАГИСТРАНТОВ ПО ИЗУЧЕНИЮ ДИСЦИПЛИНЫ**

### **6.1. Изучение программы курса**

На лекциях преподаватель рассматривает контрольные вопросы курса, составленной в соответствии с государственным образовательным стандартом. Из-за недостаточного количества аудиторных часов некоторые темы не удастся осветить в полном объеме, в следствии этого эти контрольные вопросы выносятся на самостоятельное изучение магистрантами, с рекомендациями той или иной литературы и с предоставлением ссылок на компьютерные онлайн источники:

1. <https://startandroid.ru/ru/uroki/vse-uroki-spiskom.html>
2. <https://itproger.com/course/java-android>

Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. В случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в конце комплекса.

### **6.2. Контрольные вопросы**

После изучения некоторых разделов практической части курса проводятся контрольные аудиторные работы в различных формах (презентации в группах, в парах, в письменных работах и др.). Готовиться к контрольным работам нужно по материалам лекций и рекомендованным источникам.

Примерный перечень контрольных вопросов приведен в рабочей программе.

Также в конце каждой лабораторной работы необходимо проводить письменный анализ выполненных задач по алгоритмизации и программированию, а также заполнять глоссарий в тетради.

Для итогового оценивания в учебно-методическом комплексе приведен перечень примерных контрольных вопросов и СРМ по которым составляются билеты экзамена.

Структура экзаменационных билетов ориентирована на такие навыки, как умение, способность и владение, то есть отражают обретенные компетенции.

### **6.3. Лабораторные работы**

Знания, умения и навыки, приобретенные в результате освоения практикума, существенно помогут магистрантам при дальнейшем обучении и обобщении навыков программирования.

Структура лабораторного практикума включает в себя комплекс работ, скомпонованных по принципу от простого к сложному, даются примеры работ и рекомендации по выполнению. В лабораторных работах дается подробное описание команд, их опций и указана последовательность действий при решении определенных задач, что позволяет магистранту практически самостоятельно освоить команды, функции и процедуры, которые используются при алгоритмизации и программировании поставленных задач.

Для закрепления каждой пройденной темы приводятся описания лабораторных

работ. При изучении курса необходимо своевременно выполнять и сдавать преподавателю выполненные задания. В учебном процессе магистрантами широко используется учебно – методический комплекс с теоретическим содержанием, лабораторными описаниями и контрольными вопросами по данному курсу.

При изучении дисциплины магистранты должны выполнить лабораторные работы по темам, оговоренным рабочей программой.

На лабораторных работах магистранты осваивают работу с программным обеспечением путем своевременного выполнения индивидуальных заданий и ответов на контрольные. Устный опрос по каждой теме. Лабораторные задания необходимо выполнять в строгой последовательности, так как каждое последующее задание является обобщением предыдущей.

Также для более углубленного изучения и практического освоения принципов алгоритмизации, программирования и управления базами данных, магистрантам по каждой пройденной теме даются задания для самостоятельного выполнения. Самостоятельные задания включают в себя обобщающие задачи и проекты.

Своевременной называется исполнение лабораторной работы в течении недели с момента предоставления задания по плану занятий. По результатам выполнения каждой лабораторной работы магистранту выставляется балл.

#### **6.4. Самостоятельная работа магистрантов**

Самостоятельная работа магистрантов включает в себя следующие части:

Для более углубленного изучения и практического освоения программы, магистрантам по каждой пройденной теме даются задания для самостоятельного выполнения. Самостоятельные задания включают в себя обобщающие задания и проекты. Задания даются для индивидуального выполнения, а также в группах. При работе в группах, первая группа, завершившая практическое задание получает поощрительный накопительный балл. Данные баллы учитываются при итоговом оценивании. Группа не выполнившая задание получает минусовой балл. Данный вид самостоятельной работы выполняется своевременно в течении семестра и принимается в ходе практических занятий и во время дежурств преподавателя.

По итогам пройденных материалов и дополнительно изученных сведений, магистрантами выполняется объемная проектная работа по алгоритмизации и программированию с включением базы данных. Проект выполняется по всем установленным правилам и техникам, с проведением технического анализа, выбором наиболее оптимальных методов и приемов. Результаты работ принимаются и оцениваются в I и II модульном контроле (30+30 баллов).

Все виды самостоятельных работ магистрантов охватывают весь лекционный курс, а также материалы и сведения не включенные в аудиторный курс.

В ходе работы над проектом, магистрант имеет право получать консультацию по выполнению у преподавателя во время дежурств.

## VII. КОНСПЕКТЫ ЛЕКЦИЙ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Конспекты лекций соответствуют содержанию учебного материала и учебно-методической карте.

### №1. Лекция. Введение в технологию AJAX. Разработка мобильных веб-приложений

**Аннотация:** AJAX. Внеполосные обращения. Формат передачи данных JSON. Объект XMLHttpRequest. Разработка мобильных веб-приложений. Протокол WAP. (ОК-2, ПК-9)

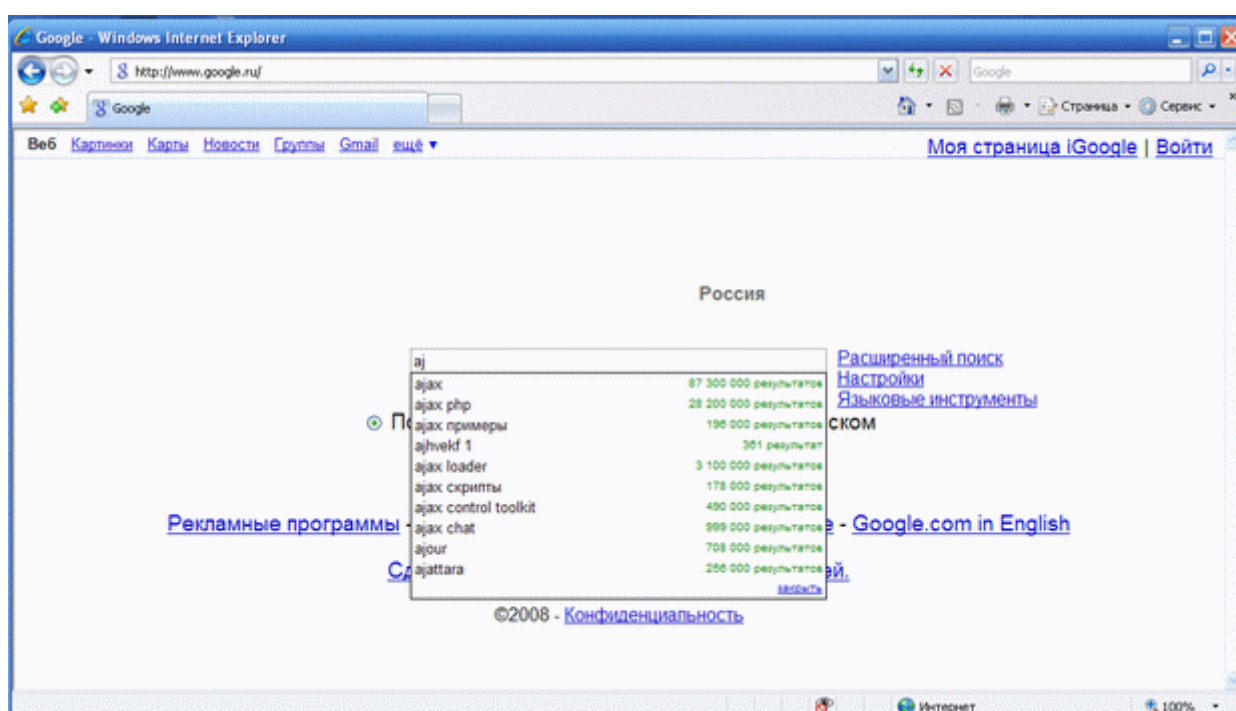
**План лекции:**

- 1.1. Виды сайтов
- 1.2. Этапы проектирования и разработки сайта
- 1.3. Модели проектирования и управление проектами
- 1.4. Модель водопада управления проектами
- 1.5. Итерационная (спиральная) модель

*AJAX (Asynchronous JavaScript and XML)* – это концепция использования нескольких смежных технологий, ориентированная на разработку высокоинтерактивных приложения, быстро реагирующих на действия пользователя, выполняющих большую часть работы на стороне клиента и взаимодействующих с сервером посредством *внеполосных обращений*.

*Внеполосным обращением* называется *запрос* к серверу, который приводит к оперативному обновлению страницы вместо ее замены. Внеполосный вызов *HTTP* – это *HTTP запрос*, который выдается за пределами встроенного модуля, обеспечивающего отправку форм *HTTP*. Вызов инициируется событием, связанным со страницей *HTML* и обслуживается компонентом-посредником, обычно объектом *XmlHttpRequest*.

Популярность *AJAX* связана с появлением сервиса *Google Suggest* в 2005 году. Данный сервис на основе объекта *XMLHttpRequest* предоставляет в распоряжение пользователя достаточно динамический веб-интерфейс. В процессе ввода символов пользователем в *поле* поискового запроса JavaScript отправляет их на *сервер* и получает от него *список* подсказок:



AJAX применяется для разработки веб-приложений, к которым предъявляются следующие требования:

- Приложение должно передавать пользователям свежие данные, полученные с сервера.
- Новые данные должны интегрироваться в существующую страницу без ее полного обновления.

Для работы с такими приложениями в браузере, необходимо, чтобы он соответствовал требованиям:

- Поддержка посредников (для внеполосных вызовов HTTP). Обычно реализуется в форме объекта *XmlHttpRequest*.
- Поддержка обновляемой модели DOM.

Объект *XmlHttpRequest* представляет собой компактную объектную модель для отправки сценарием обращений *HTTP* в обход браузера. Клиентский код сценария не может влиять на процесс размещения запроса и результат отправки запроса. *XmlHttpRequest* позволяет сценарию отправлять *HTTP* запросы и обрабатывать полученные ответы.

В качестве формата передачи данных обычно используются *JSON* или *XML*.

*JSON* (*JavaScript Object Notation*) - текстовый формат обмена данными, основанный на *JavaScript* и обычно используемый именно с этим языком. Несмотря на происхождение от *JavaScript*, формат считается языково-независимым и может использоваться практически с любым языком программирования. Для многих языков существует готовый код для создания и обработки данных в формате *JSON*.

*JSON* строится на двух структура данных:

- Набор пар *имя/значение*. В различных языках это реализовано как объект, запись, структура, словарь, хэш-таблица, список с ключом или ассоциативный массив.
- Пронумерованный набор значений. Во многих языках это реализовано как массив, вектор, список или последовательность.

Следующий пример показывает *JSON-представление* объекта, описывающего преподавателя.

```
{
  "firstName": "Сергей",
  "lastName": "Иванов",
  "profile": {
    "position": "доцент, к.ф.-м.н.",
    "department": "кафедра информационных систем",
    "age": 40
  },
  "subjects": [
    "интернет-технологии",
    "информационные технологии"
  ]
}
```

Microsoft *.NET* предоставляет в распоряжение разработчиков свою реализацию *AJAX* технологии - *ASP.NET 2.0 AJAX*.

С архитектурной точки зрения, *инфраструктура ASP.NET 2.0 AJAX* состоит из двух элементов:

- **Библиотека клиентских сценариев** (реализована на JavaScript). Работает в любом современном браузере.
- **Набор серверных расширений.** Полностью интегрируется с серверными службами и управляющими элементами ASP.NET. Разработчики могут создавать веб-страницы с расширенной функциональностью, используя практически такую же методику, которая используется при разработке серверных страниц ASP.NET.

## Разработка мобильных веб-приложений

Для получения доступа к ресурсам *интернет* посредством только мобильного телефона, не прибегая к помощи компьютера или модема был разработан специальный стандарт *WAP*.

*WAP* ( *Wireless Application Protocol* ) - протокол доступа к ресурсам *Интернет* непосредственно с мобильного телефона, минуя компьютера и/или модем.

Для разметки документов при загрузке их в сотовых телефонах и других мобильных устройствах по стандарту *WAP* также был разработан и специальный язык - *WML (Wireless Markup Language)*.

Изначально *WAP* создавался для широкого круга технологий и стандартов беспроводной мобильной связи: сотовой, транковой, пейджинговой и микросотовой, а также для поддержки сетей 3G. Данный стандарт инвариантен к операционному ядру, с которым взаимодействует *WAP-браузер* и разрабатывался как *открытый стандарт* для беспроводной передачи данных, не зависящий от поставщиков устройств и услуг, оптимизированный для мобильных телефонов, имеющих *дисплей* с маленьким разрешением, ограниченной памятью и невысокой производительностью.

*WAP 2.0* - усовершенствованная версия *WAP*, которая использует урезанный вариант *XHTML* и *CSS*. Это позволяет работать с *WAP 2.0* сайтами с помощью обычного браузера на компьютере без установки каких-либо дополнительных *плагинов*.

*XHTML MP (XHTML Mobile Profile)* - язык разметки в *WAP 2.0*, разработанный для мобильных устройств.

*Архитектура WAP* аналогична *WWW*. В *WAP* используется тот же самый способ адресации ресурсов и те же обозначения типов данных. В качестве клиента выступает *мобильное устройство* со встроенным *WAP-браузером*, запросы от которого через *WAP-шлюз* передаются веб-серверу, и ответ от последнего через него же отправляется клиенту.

В качестве *сервера* может выступать самый обычный веб-сервер. В этом случае между *WAP-шлюзом* и сервером используется протокол *HTTP*. С целью уменьшения объема передаваемых данных, текстовые ресурсы, пришедшие от сервера, передаются клиенту в *двоичном виде*.

Язык *WML* напоминает *HTML*, но ориентирован на устройства с экраном низкого разрешения и с небольшим размером памяти. Вся информация в *WML* содержится в так называемых "*деках*".

*Дек* - это минимальный блок данных, который может быть передан сервером. В *деках* находятся "*карты*" (каждая карта ограничена тегами `<card>`). На экране устройства в каждый момент времени отображается только одна *карта*, а пользователь может переключаться между ними переходя по ссылкам. Размер *WML-страницы* должен находиться в пределах от 1-4 килобайт.

Например *WML-страница* :

```

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

```

отобразится на дисплее мобильного телефона таким образом (был использован программный эмулятор телефона):



Многие мобильные устройства могут отображать документы только в *WBXML-формате*.

*WBXML* (*WAP Binary XML*) - формат компактного бинарного представления *XML*. *WBXML* используется для передачи через беспроводные соединения с низкой скоростью.

Некоторые браузеры со специальным плагином, позволяют просматривать *WML-страницы* на обычном компьютере.

В настоящее время наблюдается тенденция перехода от *WML* к *XHTML*.

Microsoft *.NET* для создания веб-приложений для мобильных устройств. В распоряжение разработчиков инструмент *.NET Mobile*

*.NET Mobile* представляет собой расширение Microsoft *ASP.NET* и Microsoft *.NET Framework*. По-сути, *.NET Mobile* представляет собой набор серверных управляющих элементов для форм, ориентированных на использование в беспроводных мобильных устройствах. Эти *элементы управления* генерируют разный код для различных устройств на языках *WML*, *HTML* или *Compact HTML* (сHTML).



Управляющие элементы ASP .NET Mobile расширяют функции SDP (*Smart Device Programmability*) и платформу .NET Compact Framework. Они позволяют использовать возможности платформы .NET Compact Framework и среды разработки Visual Studio .NET для создания мобильных веб-приложений благодаря возможности доставки данных на разнообразные мобильные устройства при помощи технологии ASP.NET. Этот подход позволяет создать в среде Visual Studio .NET единое мобильное веб-приложение, автоматически производящее формирование данных для отображения на разнообразных устройствах: мобильных телефонах, пейджерах, смартфонах, Pocket PC и др. При этом интегрированная среда разработки позволяет создавать мобильные веб-приложения, просто перетаскивая управляющие элементы в дизайнер форм.

Система .NET Mobile не устанавливает никаких компонентов на клиентское устройство. Для адаптации форматирования под конкретные браузеры используется серверная логика. Она генерирует данные в форматах WML, HTML, и сHTML. Помимо адаптивной генерации веб-страниц, технология .NET Mobile предоставляет богатый набор средств индивидуализации и расширения, обеспечивая простой способ осуществления поддержки новых устройств. Кроме того, технология .NET Mobile позволяет разработчикам управлять представлением данных для конкретного устройства или класса устройств в одной программной модели и обеспечивает легкую реализацию поддержки новых устройств без переделки существующих веб-приложений.

1. Мобильное устройство. С него исходит запрос на загрузку веб-страницы.
2. Интернет. Запрос передается через сеть соответствующему веб-серверу.
3. Веб-сервер IIS получает запрос от мобильного устройства и передает его обработчику.
4. .NET Framework выполняет обработку запроса.
5. ASP.NET компилирует запрашиваемый устройством документ.
6. .NET Mobile. Реализует элементы веб-страницы с учетом требований конкретного типа мобильного устройства.
7. Веб-страница возвращается обратно клиентскому устройству.

#### **Контрольные вопросы:**

1. Как и из какого источника можно подключить библиотеки JQuery?
2. По какому событию и с какими параметрами происходит вызов метода ajaxSubmit плагина JQuery.form.js?
3. Какие Вы знаете правила проверки полей форм в библиотеке JQuery.validate.js?
4. Как сформировать сообщение об ошибках при нарушении правил проверки полей форм в библиотеке JQuery.validate.js?
5. Как указать место вывода ошибок в библиотеке JQuery.validate.js?

#### **План СРМ:**

1. JavaScript. Объектная модель документа и работа с ней. Применение библиотек семейства JQuery.

## VIII. ЛАБОРАТОРНЫЕ МАТЕРИАЛЫ

### Лабораторная работа №1: Компоненты технологии AJAX. Методы обработки событий. Методы передачи данных.

#### 1. Цель работы

Изучить компоненты технологии AJAX, методы обработки событий, методы передачи данных, освоить применение технологии AJAX.

#### 2. Краткие теоретические сведения

AJAX – это акроним, раскрывающийся как Asynchronous JavaScript And XML и означающий асинхронный JavaScript и XML. AJAX – это технология, включающая сценарии JavaScript в документе HTML, которые по мере необходимости в фоновом режиме выполняют запросы к серверу (серверным сценариям на каком либо языке web-программирования), и получают необходимые данные, обновляя отдельные части документа и тем самым исключая необходимость его повторной загрузки или загрузки другого документа.

Фоновый запрос выполняется объектом XMLHttpRequest (встроенным в браузер), который позволяет из JavaScript организовать асинхронный доступ к серверу, благодаря чему пользователь имеет возможность продолжать работу с документом в ожидании получения ответа. Сценарий на стороне сервера отправляет свой ответ по протоколу HTTP. Ответ должен иметь такой формат, который может быть разобран кодом сценария JavaScript на стороне клиента. Традиционно используется формат XML, но можно использовать и другой формат, даже простой текст. Наиболее популярная альтернатива XML – JavaScript Object Notation (JSON), представление объектов в JavaScript.

Непосредственное применение методов и свойств объекта XMLHttpRequest, равно как и его создание в зависимости от текущего браузера, заметно увеличивает объем кода и затрудняет его разработку. Библиотека jQuery существенно упрощает жизнь разработчику приложений AJAX. Часто для вызова jQuery-методов используется сокращенный вариант, функция \$ (знак доллара), а к наиболее популярным методам относятся jQuery.get, jQuery.post, jQuerygetJSON. Рассмотрим пример с применением последнего метода. Вначале подключаем библиотеку jQuery:

```
<script type="text/javascript" src="js/jquery-1.2.6.js"></script>
```

затем ожидаем окончания загрузки страницы:

```
<script type="text/javascript">
```

```
jQuery(document).ready(function() {
```

и возникновения нужного события какого-либо тега в ней, например, смены выбранного элемента списка с идентификатором (id) #country:

```
jQuery('#country').change(function() {
```

Прочитав значение выбранного элемента

```
value=$("#select[@id=country] option:selected").val();
```

очищаем элементы связанного (подчиненного) списка с именем (name) city:

```
$('select[@name=city] option').remove();
```

Если в первом списке выбор не сделан, то и во втором выводить нечего:

```
if(!value) {$('select[@name=city]').append('<option value='+0+'>');}
```

Здесь появятся элементы массива</option>');}

Выполняем запрос к серверу методом GET с ожиданием ответа в массив data:

```
$.getJSON("select.php?country="+value, function(data) {
```

Полученные данные разбираем и помещаем в список с именем (name) city:

```
for(i in data) {
```

```
$('#select[@name=city]').append('<option
```

```
value='+i+'>'+data[i]+'</option>');});
```

При запросе данных в формате XML незначительно отличается метод выполнения запроса:

```
$.get("select.php?country="+value, function(data) {
и весьма существенно – разбор полученных данных и добавление их в список:
idarray=data.documentElement.getElementsByTagName('id');
cityarray=data.documentElement.getElementsByTagName('name');
var j=0;
for (var i=0;i<idarray.length;i++) {
    j=idarray.item(i).firstChild.data;
    name=cityarray.item(i).firstChild.data;
    elem[j]=name;
    $('select[@name=city]').append('<option
value='+j+'>'+elem[j]+'</option>');
}
}
```

Сценарии на стороне сервера тоже имеют свои особенности. Они должны давать ответ только на запросы от объекта XMLHttpRequest:

```
if($_SERVER["HTTP_X_REQUESTED_WITH"] == "XMLHttpRequest")
в соответствующем формате (обязательно в кодировке UTF-8):
header('Content-Type: application/json; charset=utf-8');
```

или

```
header("Content-type: text/xml; charset=UTF-8");
echo "<?xml version='1.0' encoding='UTF-8' standalone='yes'?>";
```

При формировании ответа в формате JSON можно использовать функцию json\_encode:

```
echo json_encode(array("1" => 'Алушта', '2' => 'Сочи'));
```

при версии PHP >= 5.2.0 или array\_to\_json в противном случае:

```
echo array_to_json(array("1" => 'Алушта', '2' => 'Сочи'));
```

Сформировать ответ в формате XML можно руками, если структура ответа проста:

```
if($_GET['country']==1) {
echo "<response><cities>";
echo "<city><id>1</id><name>Алушта</name></city>";
echo "<city><id>2</id><name>Сочи</name></city>";
echo "</cities></response>";}
}
```

Или, если версия PHP >= 5.1.0, с помощью объекта DOMDocument:

```
$dom=new DOMDocument();
$response=$dom->createElement('response');
$dom->appendChild($response);
$cities=$dom->createElement('cities');
$response->appendChild($cities);
if($_GET['country']==1) {
$id=$dom->createElement('id');
$idvalue=$dom->createTextNode('1');
$id->appendChild($idvalue);
$name=$dom->createElement('name');
$nametext=$dom->createTextNode('Алушта');
$name->appendChild($nametext);
$city=$dom->createElement('city');
$city->appendChild($id);
$city->appendChild($name);
$cities->appendChild($city);
}
$xmlString=$dom->saveXML();
echo $xmlString;
```

Применяя AJAX необходимо помнить, что на стороне клиента JavaScript может быть

отключен, что сделает приложения AJAX нефункциональными.

Пример применения технологии AJAX для фоновой отправки данных и обновления страниц сайта без перезагрузки см. здесь: <http://gun.cs.nstu.ru/web/labs/lab8>.

### **3. Методические указания**

- 3.1. Для разработки сценариев применяйте любые доступные текстовые редакторы, например, Jview или Notepad++.
- 3.2. Сценарии располагаются на сервере с адресом 217.71.139.74, подключение к которому производится по протоколу FTP с именем учетной записи userNM, где N – порядковый номер группы, M- номер студента по журналу, пароль сообщается преподавателем.
- 3.3. Лабораторная работа выполняется в отдельной папке на сервере, куда копируются сценарии лабораторной работы №3 для модификации.
- 3.4. При настройке клиента FTP необходимо установить пассивный режим.
- 3.5. При разработке страниц периодически сохраняйте их и проверяйте их отображение в любом из доступных браузеров.
- 3.6. При написании сценариев обращайте внимание на кодировку русского текста, правильность написания HTML-тегов, имен переменных и функций PHP, парность открывающих и закрывающих кавычек в операторе echo.
- 3.7. По окончании лабораторной работы запишите документы сайта на сменный носитель для сохранения и распечатки.

### **4. Порядок выполнения работы**

- 4.1. Модифицировать базу данных из лабораторной работы №1, добавив таблицу-справочник, уточняющий существующий справочник, например, для факультета – список групп, для марок автомобилей – модель и т.д.
- 4.2. Модифицировать сценарии отображения записей из лабораторной работы №3 для отображения записей из дополнительного справочника.
- 4.3. Модифицировать сценарии добавления и изменения записей из лабораторной работы №3 для заполнения списка из дополнительного справочника в зависимости от выбранного элемента списка основного справочника.
- 4.4. Отладить сценарии, убедиться в их корректной работе, отображении замечаний, продемонстрировать их работоспособность преподавателю.
- 4.5. Сохранить на локальной машине и распечатать файлы отлаженных сценариев.

### **5. Варианты заданий**

Задания различаются базами данных, реализованными в соответствии с заданиями к лабораторной работе №1, форматами передачи данных от сервера (нечетные варианты – XML, четные JSON), и являются модификацией лабораторной работы №3.

### **6. Содержание отчета**

- 6.1. Цель работы.
- 6.2. Вариант задания.
- 6.3. Листинги сценариев.
- 6.4. Выводы по работе.

### **7. Контрольные вопросы**

- 7.1. Что входит в состав технологии AJAX?
- 7.2. Каковы наиболее популярные методы отправки данных в библиотеке JQuery?
- 7.3. Какие Вы знаете события, пригодные для вызова методов библиотеки JQuery?
- 7.4. Как можно указать тег, чьи события отслеживаются?
- 7.5. Как можно указать формат получаемых из запроса данных?
- 7.6. Каковы особенности сценариев, отвечающих на фоновый запрос?
- 7.7. Как в серверном сценарии указать формат ответа?

- 7.8. Каковы способы формирования ответа в формате JSON?
- 7.9. Каковы способы формирования ответа в формате XML?
- 7.10. Чем отличается разбор данных, пришедших в формате JSON и XML?