

Документ подписан простой электронной подписью  
Информация о владельце:

ФИО: Исаев Игорь Магомедович

Должность: Проректор по безопасности и общим вопросам

Дата подписания: 28.01.2023 10:51:07

Уникальный программный ключ:

d7a26b9e8ca85e98ac3de2ab454b4659d961f749

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования

«Национальный исследовательский технологический университет «МИСиС»

## Рабочая программа дисциплины (модуля)

# Разработка клиент-серверных приложений

Закреплена за подразделением

Кафедра инфокоммуникационных технологий

Направление подготовки

09.03.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ

Профиль

Квалификация

**Бакалавр**

Форма обучения

**очная**

Общая трудоемкость

**3 ЗЕТ**

Часов по учебному плану

108

Формы контроля в семестрах:

в том числе:

экзамен 4

аудиторные занятия

34

самостоятельная работа

30

часов на контроль

44

### Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	4 (2.2)		Итого	
	18			
Неделя	УП	РП	УП	РП
Лекции	17	17	17	17
Лабораторные	17	17	17	17
Итого ауд.	34	34	34	34
Контактная работа	34	34	34	34
Сам. работа	30	30	30	30
Часы на контроль	44	44	44	44
Итого	108	108	108	108

Программу составил(и):  
*к.тн, доцент, Стучилин В.В.*

Рабочая программа

**Разработка клиент-серверных приложений**

Разработана в соответствии с ОС ВО:

Самостоятельно устанавливаемый образовательный стандарт высшего образования - бакалавриат Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский технологический университет «МИСиС» по направлению подготовки 09.03.02 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ (приказ от 02.04.2021 г. № 119 о.в.)

Составлена на основании учебного плана:

09.03.02 Информационные системы и технологии, 09.03.02-БИСТ-22.plx , утвержденного Ученым советом ФГАОУ ВО НИТУ "МИСиС" в составе соответствующей ОПОП ВО 22.09.2022, протокол № 8-22

Утверждена в составе ОПОП ВО:

09.03.02 Информационные системы и технологии, , утвержденной Ученым советом ФГАОУ ВО НИТУ "МИСиС" 22.09.2022, протокол № 8-22

Рабочая программа одобрена на заседании

**Кафедра инфокоммуникационных технологий**

Протокол от 24.06.2021 г., №8

Руководитель подразделения Калашников Е.А.

**1. ЦЕЛИ ОСВОЕНИЯ**

1.1	Изучение методов с средств программирования для овладения знаниями в области разработки клиент-серверных приложений; подготовка студентов к осознанному использованию, как методов, так и средств программирования клиент-серверных приложений.
-----	---

**2. МЕСТО В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ**

Блок ОП:		Б1.О
<b>2.1</b>	<b>Требования к предварительной подготовке обучающегося:</b>	
2.1.1	Базы данных	
2.1.2	Технологии программирования	
2.1.3	Объектно-ориентированное программирование	
2.1.4	Вычислительные машины, сети и системы	
2.1.5	Программирование и алгоритмизация	
<b>2.2</b>	<b>Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:</b>	
2.2.1	Оптимизация клиент-серверных приложений	
2.2.2	Разработка сетевых приложений на языке программирования Python	
2.2.3	Системы управления технологическими процессами и производствами	
2.2.4	Теория информационных процессов и систем	
2.2.5	Цифровая электроника	
2.2.6	Интернет вещей	
2.2.7	Компьютерное зрение	
2.2.8	Программируемые логические контроллеры	
2.2.9	Производственная практика по получению профессиональных умений и опыта профессиональной деятельности	
2.2.10	Производственная практика по получению профессиональных умений и опыта профессиональной деятельности	
2.2.11	Разработка мобильных приложений	
2.2.12	Технология разработки ПО	
2.2.13	Инфокоммуникационные системы и сети	
2.2.14	Каналы передачи информации	
2.2.15	Методология проектирования информационных систем	
2.2.16	Нормы и правила оформления НИР и ВКР	
2.2.17	Теория систем автоматического управления	
2.2.18	Технологии виртуальной и дополненной реальностей	
2.2.19	Цифровые двойники производственных объектов	
2.2.20	Аппаратные средства хранения и обработки данных	
2.2.21	Инструменты DevOps	
2.2.22	Интеллектуальные информационные системы	
2.2.23	Информационные системы "Умный город"	
2.2.24	Компьютерные технологии управления	
2.2.25	Подготовка к процедуре защиты и защита выпускной квалификационной работы	
2.2.26	Подготовка к процедуре защиты и защита выпускной квалификационной работы	
2.2.27	Программно-аппаратные платформы корпоративных информационных систем	
2.2.28	Проектирование информационных систем	
2.2.29	Типовые интерфейсы и сетевое оборудование	
2.2.30	Решение задач с использованием прикладного ПО	

**3. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ФОРМИРУЕМЫМИ КОМПЕТЕНЦИЯМИ**

**ОПК-5: Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем**

**Знать:**

ОПК-5-31 Особенности установки и использования ПО для разработки клиент-серверных приложений.

<b>ОПК-6: Способен осуществлять моделирование, анализ и эксперименты в целях проведения детального исследования, разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием</b>
<b>Знать:</b>
ОПК-6-31 Методы проектирования клиент-серверных приложений.
<b>ПК-1: Способность создавать (модифицировать) и сопровождать информационные системы и инфокоммуникационные технологии цифровой экономики</b>
<b>Знать:</b>
ПК-1-31 Языки программирования для разработки клиент-серверных приложений.
<b>ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности</b>
<b>Знать:</b>
ОПК-2-31 Основные понятия и методы разработки клиент-серверных приложений.
<b>ПК-1: Способность создавать (модифицировать) и сопровождать информационные системы и инфокоммуникационные технологии цифровой экономики</b>
<b>Уметь:</b>
ПК-1-У1 Разрабатывать алгоритмическое и программное обеспечение клиент-серверных систем.
<b>ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности</b>
<b>Уметь:</b>
ОПК-2-У1 Применять полученные теоретические знания при практических задачах разработки клиент-серверных приложений.
<b>ПК-1: Способность создавать (модифицировать) и сопровождать информационные системы и инфокоммуникационные технологии цифровой экономики</b>
<b>Владеть:</b>
ПК-1-В1 Иметь навыки разработки клиент-серверных приложений.
<b>ОПК-2: Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности</b>
<b>Владеть:</b>
ОПК-2-В1 Иметь навыки использования современных программных средств анализа и тестирования клиент-серверных приложений.

#### 4. СТРУКТУРА И СОДЕРЖАНИЕ

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Формируемые индикаторы компетенций	Литература и эл. ресурсы	Примечание	КМ	Выполняемые работы
	<b>Раздел 1. Разработка клиентских приложений</b>							
1.1	Язык разметки HTML. Таблицы стилей CSS. Основы работы с языком JavaScript. /Лек/	4	2	ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.2Л2.4 Э1			
1.2	Разработка клиентских приложений на языке программирования Си#. /Лек/	4	2	ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.1 Э1 Э2			
1.3	Разработка клиентских приложений на языке программирования Си#. /Лаб/	4	4	ОПК-2-У1 ОПК-2-В1 ПК-1-У1 ПК-1-В1	Л1.1 Э1 Э2		КМ1	Р1

1.4	Проработка лекционного материала в электронной системе обучения Canvas Самостоятельное изучение литературы. Подготовка к лабораторным работам. Работа над рефератом и домашней работой. /Ср/	4	6	ОПК-6-31 ОПК-5-31 ОПК-2-31 ОПК-2-У1 ОПК-2-В1 ПК-1-31 ПК-1-У1 ПК-1-В1	Л1.2Л1.1 Л2.4 Э1 Э2			
	<b>Раздел 2. Разработка серверных приложений</b>							
2.1	Основные понятия серверного программирования. Методы разработки серверных приложений. Инструментальные средства разработки серверных приложений. /Лек/	4	2	ОПК-6-31 ОПК-2-31 ОПК-5-31 ПК-1-31	Л1.1Л1.2 Л2.2 Э1 Э3			
2.2	Стандартные алгоритмы серверных приложений. Синтаксис языка программирования PHP. /Лек/	4	2	ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.1Л2.2 Э1 Э3			
2.3	Управляющие конструкции PHP. Функции в PHP. Объекты и классы в PHP. Работа с файловой системой. /Лек/	4	2	ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.1Л2.2 Э1 Э3			
2.4	Базы данных и СУБД. Взаимодействие PHP и MySQL. Работа с сессиями в PHP. /Лек/	4	2	ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.1Л2.2 Э1 Э3			
2.5	Межсерверное взаимодействие. Взаимодействие PHP и XML. Информационная безопасность серверных приложений. /Лек/	4	3	ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.1Л2.2 Э1 Э3			
2.6	Работа с графическими изображениями на стороне сервера. /Лек/	4	2	ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.1Л2.2 Э1 Э3			
2.7	Основы работы с языком программирования PHP. /Лаб/	4	2	ОПК-2-У1 ОПК-2-В1 ПК-1-У1 ПК-1-В1	Л1.1Л2.2 Э1 Э3		КМ2	Р2
2.8	Работа с методами протокола http (https). /Лаб/	4	2	ОПК-2-У1 ОПК-2-В1 ПК-1-У1 ПК-1-В1	Л1.1Л2.2 Э1 Э3		КМ3	Р3
2.9	Работа с сессиями в PHP. /Лаб/	4	2	ОПК-2-У1 ПК-1-У1 ОПК-2-В1 ПК-1-В1	Л1.1Л2.2 Э1 Э3		КМ4	Р4
2.10	Взаимодействие с СУБД MySQL. /Лаб/	4	2	ОПК-2-У1 ОПК-2-В1 ПК-1-У1 ПК-1-В1	Л1.1Л2.2 Э1 Э3		КМ5	Р5
2.11	Межсерверное взаимодействие. Разработка синтаксического анализатора XML документа. /Лаб/	4	3	ОПК-2-У1 ОПК-2-В1 ПК-1-У1 ПК-1-В1	Л1.1Л2.2 Э1 Э3		КМ6	Р6
2.12	Работа с графическими изображениями средствами PHP. /Лаб/	4	2	ОПК-2-У1 ОПК-2-В1 ПК-1-У1 ПК-1-В1	Л1.1Л2.2 Э1 Э3		КМ7	Р7

2.13	Проработка лекционного материала в электронной системе обучения Canvas Самостоятельное изучение литературы. Подготовка к лабораторным работам. Работа над рефератом и домашней работой. /Ср/	4	24	ОПК-2-У1 ОПК-2-В1 ПК-1-У1 ПК-1-В1 ОПК-6-31 ОПК-5-31 ОПК-2-31 ПК-1-31	Л1.1Л2.2 Э1 Э3			
------	--	---	----	--	-------------------	--	--	--

### 5. ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ

#### 5.1. Контрольные мероприятия (контрольная работа, тест, коллоквиум, экзамен и т.п), вопросы для самостоятельной подготовки

Код КМ	Контрольное мероприятие	Проверяемые индикаторы компетенций	Вопросы для подготовки
КМ1	Защита результатов работы №1.		Примеры вопросов для самопроверки: 1. Особенности разработки сетевых программ в Си#. 2. Описание протокола прикладного уровня http. 3. Методы работы с протоколами стека TCP/IP.
КМ2	Защита результатов работы №2.		Примеры вопросов для самопроверки: 1. Технология CGI. Назначение и примеры реализации. 2. Синтаксис PHP. Пример программы. 3. Типы переменных в PHP (кроме строкового типа). Конвертация типов. Примеры. 4. Строковые переменные в PHP. Способы инициализации. Примеры. 5. Массивы в PHP. Примеры использования. Примеры. 6. Комментарии в PHP. Присвоение значений переменным. Примеры. 7. Операции инкремента/декремента, арифметические и строковые операции. Примеры программ. 8. Условные операции. Операторы сравнения. Примеры. 9. Операторы цикла в PHP. Примеры. 10. Оператор foreach. Назначение и примеры использования. 11. Оператор switch. Назначение и примеры использования. 12. Операторы include и echo. Назначение и примеры использования.
КМ3	Защита результатов работы №3.		Примеры вопросов для самопроверки: 1. Понятие клиент-серверных технологий. Назначение и реализация Web-сервера. Примеры. 2. Протокол http. Назначение и принцип работы. Примеры. 3. Форма запроса клиента в протоколе http. Примеры. 4. Методы в протоколе http. Краткое описание. Примеры. 5. Метод GET в протоколе http. Назначение и пример использования. 6. Метод POST в протоколе http. Назначение и пример использования. 7. Обработка данных формы в PHP. Примеры. 8. Использование суперглобальных массивов для работы с формами (\$_POST и т.д.). Примеры.

КМ4	Защита результатов работы №4.		<p>Примеры вопросов для самопроверки:</p> <ol style="list-style-type: none"> <li>1. Сессии в PHP. Модель «клиент-сервер». Примеры.</li> <li>2. Cookies. Назначение и методы работы. Примеры</li> <li>3. Создание и использование сессионных переменных. Примеры.</li> <li>4. Регистрация переменных сессии. Примеры.</li> <li>5. Удаление переменных сессии. Примеры.</li> <li>6. Проблемы обеспечения безопасности при работе с сессиями.</li> <li>7. Аутентификация при помощи встроенных средств http. Примеры.</li> <li>8. Аутентификация при помощи сессионных переменных. Примеры.</li> <li>9. Массив \$_SESSION. Назначение и примеры использования.</li> <li>10. Передача переменных при помощи query string.</li> </ol>
КМ5	Защита результатов работы №5.		<p>Примеры вопросов для самопроверки:</p> <ol style="list-style-type: none"> <li>1. Алгоритм работы с БД в PHP. Примеры.</li> <li>2. Функция mysql_connect. Примеры работы.</li> <li>3. Функция mysql_select_db. Примеры работы.</li> <li>4. Функция mysql_query. Примеры работы.</li> <li>5. Функция mysql_fetch_array. Примеры работы.</li> <li>6. Добавление записи в БД. Запрос INSERT. Примеры кода PHP.</li> <li>7. Выборка записей из БД. Запрос SELECT. Примеры кода PHP.</li> <li>8. Обновление записей в БД. Запрос UPDATE. Примеры кода PHP.</li> <li>9. Удаление записей из БД. Запрос DELETE. Примеры кода PHP.</li> <li>10. Запросы в СУБД для создания БД и таблиц. Примеры кода PHP.</li> </ol>
КМ6	Защита результатов работы №6.		<p>Примеры вопросов для самопроверки:</p> <ol style="list-style-type: none"> <li>1. Использование библиотеки cURL. Примеры.</li> <li>2. Функция curl_init. Назначение и примеры использования.</li> <li>3. Функция curl_setopt. Назначение и примеры использования.</li> <li>4. Функция curl_exec. Назначение и примеры использования.</li> </ol>
КМ7	Защита результатов работы №7.		<p>Примеры вопросов для самопроверки:</p> <ol style="list-style-type: none"> <li>1. Функция imageCreate. Назначение и примеры использования.</li> <li>2. Функция imagePng. Назначение и примеры использования.</li> <li>3. Функция imageColorAllocate. Назначение и примеры использования.</li> <li>4. Функция imageLine. Назначение и примеры использования.</li> <li>5. Функция imageRectangle. Назначение и примеры использования.</li> </ol>

КМ8	Реферат.		<p>Примеры тем рефератов:</p> <ol style="list-style-type: none"> <li>1. Математические функции в PHP.</li> <li>2. Использование функций GMP.</li> <li>3. Работа с массивами в PHP. Индексные и ассоциативные массивы.</li> <li>4. Строковые функции в PHP.</li> <li>5. Использование SimpleXML.</li> <li>6. Функции даты и времени в PHP.</li> <li>7. Работа с файлами.</li> <li>8. Функции для работы с каталогами.</li> <li>9. Использование протокола FTP при программировании серверных приложений.</li> <li>10. Функции сжатия Vzip2.</li> <li>11. Сетевые функции в PHP.</li> <li>12. Отслеживание и обработка ошибок в PHP.</li> <li>13. Управление сессиями.</li> <li>14. Работа с WWW.</li> <li>15. Управление выводом в PHP.</li> </ol>
КМ9	Контрольная работа №1.		<p>Примерные вопросы для самопроверки по контрольной работе:</p> <ol style="list-style-type: none"> <li>1. Технология CGI. Назначение и примеры реализации.</li> <li>2. Синтаксис PHP. Пример программы.</li> <li>3. Операторы include и echo. Назначение и примеры использования.</li> <li>4. Понятие клиент-серверных технологий. Назначение и реализация Web-сервера. Примеры.</li> <li>5. Протокол http. Назначение и принцип работы. Примеры.</li> <li>6. Создание файла. Функция fopen. Примеры.</li> <li>7. Параметры функции fopen. Типы доступа к файлу. Примеры.</li> <li>8. Сессии в PHP. Модель «клиент-сервер». Примеры.</li> <li>9. Создание и использование сессионных переменных. Примеры.</li> <li>10. Регистрация переменных сессии. Примеры.</li> </ol>
КМ10	Контрольная работа №2.		<p>Примерные вопросы для самопроверки по контрольной работе:</p> <ol style="list-style-type: none"> <li>1. Создание файла. Функция fopen. Примеры.</li> <li>2. Параметры функции fopen. Типы доступа к файлу. Примеры.</li> </ol> <p>Чтение строк из файла в массив. Примеры.</p> <ol style="list-style-type: none"> <li>3. Удаление переменных сессии. Примеры.</li> <li>4. Алгоритм работы с БД в PHP. Примеры.</li> <li>5. Функция mysql_connect. Примеры работы.</li> <li>6. Добавление записи в БД. Запрос INSERT. Примеры кода PHP.</li> <li>7. Выборка записей из БД. Запрос SELECT. Примеры кода PHP.</li> </ol>



KM11	Защита домашней работы.		<p>Примерные вопросы при защите курсовой работы:</p> <ol style="list-style-type: none"> <li>1. Сессии в PHP. Модель «клиент-сервер». Примеры.</li> <li>2. Cookies. Назначение и методы работы. Примеры</li> <li>3. Создание и использование сессионных переменных. Примеры.</li> <li>4. Регистрация переменных сессии. Примеры.</li> <li>5. Удаление переменных сессии. Примеры.</li> <li>6. Проблемы обеспечения безопасности при работе с сессиями.</li> <li>7. Аутентификация при помощи встроенных средств http. Примеры.</li> <li>8. Аутентификация при помощи сессионных переменных. Примеры.</li> <li>9. Массив \$_SESSION. Назначение и примеры использования.</li> <li>10. Передача переменных при помощи query string. Примеры.</li> <li>11. Алгоритм работы с БД в PHP. Примеры.</li> <li>12. Функция mysql(i)_connect. Примеры работы.</li> <li>13. Функция mysql(i)_select_db. Примеры работы.</li> <li>14. Функция mysql(i)_query. Примеры работы.</li> <li>15. Функция mysql(i)_fetch_array. Примеры работы.</li> <li>16. Добавление записи в БД. Запрос INSERT. Примеры кода PHP.</li> <li>17. Выборка записей из БД. Запрос SELECT. Примеры кода PHP.</li> <li>18. Обновление записей в БД. Запрос UPDATE. Примеры кода PHP.</li> <li>19. Удаление записей из БД. Запрос DELETE. Примеры кода PHP.</li> <li>20. Запросы в СУБД для создания БД и таблиц. Примеры кода PHP.</li> <li>21. Функция imageCreate. Назначение и примеры использования.</li> <li>22. Функция imagePng. Назначение и примеры использования.</li> <li>23. Функция imageColorAllocate. Назначение и примеры использования.</li> <li>24. Функция imageLine. Назначение и примеры использования.</li> <li>25. Функция imageRectangle. Назначение и примеры использования.</li> <li>26. Использование библиотеки cURL. Примеры.</li> <li>27. Функция curl_init. Назначение и примеры использования.</li> <li>28. Функция curl_setopt. Назначение и примеры использования.</li> <li>29. Функция curl_exec. Назначение и примеры</li> </ol>
------	-------------------------	--	---

KM12	Экзамен.		<p>Примерные вопросы:</p> <ol style="list-style-type: none"> <li>1. Технология CGI. Назначение и примеры реализации.</li> <li>2. Синтаксис PHP. Пример программы.</li> <li>3. Типы переменных в PHP (кроме строкового типа). Конвертация типов. Примеры.</li> <li>4. Строковые переменные в PHP. Способы инициализации. Примеры.</li> <li>5. Массивы в PHP. Примеры использования. Примеры.</li> <li>6. Комментарии в PHP. Присвоение значений переменным. Примеры.</li> <li>7. Операции инкремента/декремента, арифметические и строковые операции. Примеры программ.</li> <li>8. Условные операции. Операторы сравнения. Примеры.</li> <li>9. Операторы цикла в PHP. Примеры.</li> <li>10. Оператор foreach. Назначение и примеры использования.</li> <li>11. Оператор switch. Назначение и примеры использования.</li> <li>12. Операторы include и echo. Назначение и примеры использования.</li> <li>13. Понятие клиент-серверных технологий. Назначение и реализация Web-сервера. Примеры.</li> <li>14. Протокол http. Назначение и принцип работы. Примеры.</li> <li>15. Форма запроса клиента в протоколе http. Примеры.</li> <li>16. Методы в протоколе http. Краткое описание. Примеры.</li> <li>17. Метод GET в протоколе http. Назначение и пример использования.</li> <li>18. Метод POST в протоколе http. Назначение и пример использования.</li> <li>19. Обработка данных формы в PHP. Примеры.</li> <li>20. Использование суперглобальных массивов для работы с формами (\$_POST и т.д.). Примеры.</li> <li>21. Создание файла. Функция fopen. Примеры.</li> <li>22. Параметры функции fopen. Типы доступа к файлу. Примеры.</li> <li>23. Закрытие соединения с файлом. Назначение и примеры.</li> <li>24. Запись данных в файл. Функция fwrite. Примеры.</li> <li>25. Чтение данных из файла. Функция fread. Примеры.</li> <li>26. Функция filesize. Назначение и примеры использования.</li> <li>27. Чтение строки из файла. Функция fgets. Примеры использования.</li> <li>28. Отличие функций fgets и fgetss. Примеры.</li> <li>29. Функция readfile. Назначение и примеры использования.</li> <li>30. Чтение строк из файла в массив. Примеры.</li> </ol>
------	----------	--	---

## 5.2. Перечень работ, выполняемых по дисциплине (Курсовая работа, Курсовой проект, РГР, Реферат, ЛР, ПР и т.п.)

Код работы	Название работы	Проверяемые индикаторы компетенций	Содержание работы
P1	Лабораторная работа №1. Разработка клиентских приложений на языке программирования Си#.		<p>Пример задания:</p> <p>Для своего варианта разработать программное обеспечение, которое предоставляет интерфейс для работы с сервером системы. Программное обеспечение должно быть разработано на языке высокого уровня (Си#, или аналоге).</p>
P2	Лабораторная работа №2. Основы работы с языком программирования PHP.		<p>Пример задания:</p> <p>Разработайте программное обеспечение, которое осуществляет следующий функционал:</p> <ol style="list-style-type: none"> <li>1. Создайте отдельную страницу, на которой будут размещаться ссылки на php скрипты.</li> <li>2. Реализуйте указанный пример, выясните его принцип работы.</li> <li>3. Получите информацию о настройках php с помощью команды phpinfo();.</li> <li>4. Создайте php-скрипт, который создает страницу "Hello World!!!";.</li> <li>5. Создайте php-скрипт, который загружает в web-страницу файл с информацией о разработчике.</li> </ol>

P3	Лабораторная работа №3. Работа с методами протокола http (https).		<p>Пример задания: Создайте форму ввода данных о пользователе (ФИО, e-mail, телефон). Напишите скрипт, который проверяет правильность заполнения полей формы и выводит данные на экран. Необходимо проверять: Для ФИО - только русские буквы; - первые буквы должны быть заглавными. Для e-mail - формат (английские буквы, цифры, _ ) @ (английские буквы, цифры, _ ) . (английские буквы, цифры, _ ) Для телефона - либо +71231234567, либо 81231234567</p>
P4	Лабораторная работа №4. Работа с сессиями в PHP.		<p>Пример задания: В текстовом файле хранится информация о пользователях в виде: имя_пользователя1:пароль:ключевое_слово имя_пользователя2:пароль:ключевое_слово ... Организуите аутентификацию с чтением данных о правильности имени и пароля из файла. При неправильном вводе должно быть выдано сообщение о неправильном имени пользователя или пароле. Если пользователь правильно ввел имя_пользователя, а пароль неверно отобразите его пароль после правильного ввода ключевого слова. Реализуйте функцию задания нового пароля.</p>
P5	Лабораторная работа №5. Взаимодействие с СУБД MySQL.		<p>Пример задания: Разработать систему «телефонный справочник».</p> <p>База данных системы должна включать, как минимум, следующую информацию:</p> <ol style="list-style-type: none"> <li>1. Фамилия, имя, отчество.</li> <li>2. Телефон.</li> <li>3. Адрес проживания.</li> </ol> <p>Система должна выполнять следующие операции:</p> <ol style="list-style-type: none"> <li>1. Ввод и корректировка данных администратором системы.</li> <li>2. Поиск и вывод информации о человеке по фамилии.</li> <li>3. Поиск и вывод информации о человеке по телефону.</li> </ol> <p>Система состоит из двух независимых блоков:</p> <ol style="list-style-type: none"> <li>1. Блок ввода информации в базу администратором</li> <li>2. Блок вывода информации по запросам пользователей</li> </ol> <p>Блок вывода информации по запросам пользователей состоит из двух модулей. Первый модуль запрашивает информацию для поиска в базе (фамилию или телефон). Второй модуль, на основании критериев поиска, формирует запрос к базе, получает ответ и формирует вывод результатов поиска в виде таблицы</p> <p>Блок ввода информации в базу позволяет вводить и изменять данные только администратору БД. Перед работой в этом блоке необходима проверка прав доступа. Эту проверку необходимо осуществлять штатными средствами браузера, при этом</p>
P6	Лабораторная работа №6. Межсерверное взаимодействие. Разработка синтаксического анализатора XML документа.		<p>Пример задания: Разработать серверное программное обеспечение для получения данных из удаленной RSS-ленты (новостной). Должно быть получено не менее 10 записей, в каждой из которых должна быть следующая информация:</p> <ol style="list-style-type: none"> <li>1. Название новости.</li> <li>2. Аннотация.</li> <li>3. Дата публикации.</li> <li>4. Ссылка на полный текст новости.</li> </ol>

P7	Лабораторная работа №7. Работа с графическими изображениями средствами PHP.		<p>Пример задания: Создать приложение, выводящее столбчатую диаграмму со случайным количеством столбцов, случайной высотой, и случайным цветом. При загрузке скрипта должна быть построена столбчатая диаграмма. Количество столбцов, их высота и цвет генерируются генератором псевдослучайных чисел. Над столбцами диаграммы необходимо указать их числовое значение (посередине столбца). Горизонтальная ось X и вертикальная ось Y должны иметь фиксированный размер в пикселях (например, 500*800 точек) и не изменяться в зависимости от количества столбцов и их высоты. Необходимо осуществить масштабирование высоты столбцов по вертикальной оси так, чтобы абсолютная высота столбца была пропорциональна максимальному столбцу (т.е. максимальный столбец занимает всю ось Y, остальные масштабируются по этой высоте). Ширина столбца должна зависеть от количества столбцов (больше столбцов меньше ширина) исходя из длины горизонтальной оси. Столбцы должны быть одинаковой ширины и расстояние между ними должно быть равно ширине столбца. Последний столбец должен заканчиваться на границе оси X.</p>
P8	Реферат.		<p>Необходимо написать реферат на выбранную студентом тему. Реферат должен быть оформлен по шаблону, должен содержать ссылки на источники. Должно быть использовано не менее трех источников.</p>
P9	Контрольная работа №2.		<p>Содержание контрольной работы:</p> <ol style="list-style-type: none"> <li>1. Работа с сессиями.</li> <li>2. Работа с БД.</li> <li>3. сURL.</li> <li>4. Регулярные выражения.</li> <li>5. Работа с изображениями.</li> </ol>

P10	Домашняя работа.	<p>Примерные задания:</p> <p>Вариант 1. Необходимо разработать ПО, позволяющее загружать изображения формата JPG в отдельную папку на сервере. Перед загрузкой на сервер необходимо уменьшать изображение до заданного размера. Если изображение меньше заданного размера, выдавать сообщение об ошибке. Необходимо учесть, что имена файлов с изображениями могут совпадать.</p> <p>Вариант 2. Необходимо разработать систему мониторинга заданного пользователем сайта в поисковых системах Yandex, Rambler, Google по указанному пользователем запросу. ПО должно выводить список сайтов с 1-ого места по место указанного сайта (но не далее 50 места).</p> <p>Вариант 3. Разработать органайзер на текущий месяц. На главном экране должен выводиться календарь на текущий месяц. При щелчке на календарном дне, должна выводиться информация по этому дню. В том случае, если пользователь авторизован, то должна быть обеспечена возможность редактирования и добавления информации по указанному дню.</p> <p>Вариант 4. Разработать ПО, которое меняет заданную текстовую строку на другую строку во всех файлах в указанной директории и ее поддиректориях (не более 3-его уровня вложения).</p> <p>Вариант 5. Разработать ПО, которое позволяют загружать на сервер файл по протоколу FTP. Предусмотреть возможность совпадения имен файлов. При правильной авторизации выводить список всех доступных файлов с их размерами и предоставлять возможность скачать выбранный файл.</p> <p>Требования к разрабатываемому ПО:</p> <ol style="list-style-type: none"> <li>1. Все программное обеспечение должно быть написано на языке программирования PHP. Допускается обоснованное использование фрагментов кода на языке SQL и HTML.</li> <li>2. Все программное обеспечение должно работать на локальном сервере без выхода в сеть Интернет (кроме заранее согласованных с преподавателем вариантов).</li> <li>3. Допускается использование любых методов, функций, классов и т.д., которые не противоречат настоящему заданию.</li> <li>4. Не допускается использование разработанных другими авторами блоков кода.</li> <li>5. При разработке допускается использование любого вспомогательного ПО при условии лицензионной чистоты этого ПО.</li> <li>6. Все данные, если это не указано отдельно, должны храниться в БД (СУБД MySQL).</li> <li>7. Для разработки интерфейса допускается использование любых языков и технологий локального программирования (Си#, Java, Java Script, VB Script, AJAX и т.д.). При этом основное задание должно быть выполнено средствами серверного программирования.</li> </ol> <p>Требования к оформлению:</p> <ol style="list-style-type: none"> <li>1. Отчет по курсовой работе должен содержать следующие разделы: <ul style="list-style-type: none"> <li>- титульный лист;</li> <li>- задание;</li> <li>- блок-схемы всех скриптов;</li> <li>- пример выполнения программы;</li> <li>- листинг;</li> <li>- выводы;</li> <li>- список используемой литературы.</li> </ul> </li> <li>2. Отчет должен быть распечатан на листах А4 и скреплен.</li> </ol>
-----	------------------	--

			Отчет и архив проекта должен быть загружен в систему Canvas  Перед выполнением задания, студент должен уточнить у преподавателя задание (получит отметку на листе задания). Допускается разработка КР по индивидуальному заданию (студент предлагает свой вариант). Обязательно должно быть согласовано с лектором.
P11	Контрольная работа №1.		Содержание контрольной работы: 1. Синтаксис PHP. 2. Переменные, типы данных, константы. 3. Арифметические и логические операции. 4. Условные операторы. Операторы перехода. 5. Циклы. 6. Обмен данными между клиентом и сервером. Методы протокола http. 7. Работа с файлами в PHP. 8. Работа с сессиями.

### 5.3. Оценочные материалы, используемые для экзамена (описание билетов, тестов и т.п.)

Экзамен проводится в письменной форме. Билет состоит из 3 теоретических вопросов и двух задач.

Оценочные материалы (средства) представляют собой задания, процедуры защиты отчетов: а) по выполненной лабораторных работ; б) по представленному реферату; в) по результатам проверки контрольных работ.

Указанные оценочные средства позволяют оценить компетенции, сформированные у обучающихся в процессе освоения дисциплины.

Оформленный реферат, отчет по домашней и лабораторным работам в электронной форме загружаются обучающимися в ЭИОС «Canvas».

Экзаменационные вопросы:

1. Технология CGI. Назначение и примеры реализации.
2. Синтаксис PHP. Пример программы.
3. Типы переменных в PHP (кроме строкового типа). Конвертация типов. Примеры.
4. Строковые переменные в PHP. Способы инициализации. Примеры.
5. Массивы в PHP. Примеры использования. Примеры.
6. Комментарии в PHP. Присвоение значений переменным. Примеры.
7. Операции инкремента/декремента, арифметические и строковые операции. Примеры программ.
8. Условные операции. Операторы сравнения. Примеры.
9. Операторы цикла в PHP. Примеры.
10. Оператор foreach. Назначение и примеры использования.
11. Оператор switch. Назначение и примеры использования.
12. Операторы include и echo. Назначение и примеры использования.
13. Понятие клиент-серверных технологий. Назначение и реализация Web-сервера. Примеры.
14. Протокол http. Назначение и принцип работы. Примеры.
15. Форма запроса клиента в протоколе http. Примеры.
16. Методы в протоколе http. Краткое описание. Примеры.
17. Метод GET в протоколе http. Назначение и пример использования.
18. Метод POST в протоколе http. Назначение и пример использования.
19. Обработка данных формы в PHP. Примеры.
20. Использование суперглобальных массивов для работы с формами (\$\_POST и т.д.). Примеры.
21. Создание файла. Функция fopen. Примеры.
22. Параметры функции fopen. Типы доступа к файлу. Примеры.
23. Закрытие соединения с файлом. Назначение и примеры.
24. Запись данных в файл. Функция fwrite. Примеры.
25. Чтение данных из файла. Функция fread. Примеры.
26. Функция filesize. Назначение и примеры использования.
27. Чтение строки из файла. Функция fgets. Примеры использования.
28. Отличие функций fgets и fgetss. Примеры.
29. Функция readfile. Назначение и примеры использования.
30. Чтение строк из файла в массив. Примеры.
31. Сессии в PHP. Модель «клиент-сервер». Примеры.
32. Cookies. Назначение и методы работы. Примеры.
33. Создание и использование сессионных переменных. Примеры.
34. Регистрация переменных сессии. Примеры.
35. Удаление переменных сессии. Примеры.
36. Проблемы обеспечения безопасности при работе с сессиями.
37. Аутентификация при помощи встроенных средств http. Примеры.
38. Аутентификация при помощи сессионных переменных. Примеры.

39. Массив `$_SESSION`. Назначение и примеры использования.
40. Передача переменных при помощи `query string`. Примеры.
41. Алгоритм работы с БД в PHP. Примеры.
42. Функция `mysql(i)_connect`. Примеры работы.
43. Функция `mysql(i)_select_db`. Примеры работы.
44. Функция `mysql(i)_query`. Примеры работы.
45. Функция `mysql(i)_fetch_array`. Примеры работы.
46. Добавление записи в БД. Запрос `INSERT`. Примеры кода PHP.
47. Выборка записей из БД. Запрос `SELECT`. Примеры кода PHP.
48. Обновление записей в БД. Запрос `UPDATE`. Примеры кода PHP.
49. Удаление записей из БД. Запрос `DELETE`. Примеры кода PHP.
50. Запросы в СУБД для создания БД и таблиц. Примеры кода PHP.
51. Функция `imageCreate`. Назначение и примеры использования.
52. Функция `imagePng`. Назначение и примеры использования.
53. Функция `imageColorAllocate`. Назначение и примеры использования.
54. Функция `imageLine`. Назначение и примеры использования.
55. Функция `imageRectangle`. Назначение и примеры использования.
56. Использование библиотеки `cURL`. Примеры.
57. Функция `curl_init`. Назначение и примеры использования.
58. Функция `curl_setopt`. Назначение и примеры использования.
59. Функция `curl_exec`. Назначение и примеры использования.
60. Алгоритм работы парсера. Примеры.

#### **5.4. Методика оценки освоения дисциплины (модуля, практики. НИР)**

По дисциплине предусмотрен письменный экзамен. Билет состоит из трех теоретических вопросов и двух типовых задач.

Экзаменационная оценка:

Оценка "отлично" выставляется студенту, полностью ответившему на три теоретических вопроса и решившему две задачи экзаменационного билета, обнаружившему всестороннее, систематическое и глубокое знание учебного материала, предусмотренного программой; усвоившему основную и знакомому с дополнительной литературой по программе; умеющему творчески и осознанно выполнять задания, предусмотренные программой; усвоившему взаимосвязь основных понятий и умеющему применять их к анализу и решению практических задач; безупречно выполнившему в процессе изучения дисциплины все задания, предусмотренные формами текущего контроля;

Оценки "хорошо" заслуживает студент, ответивший полностью на два вопроса и решившему одну задачу экзаменационного билета и ответивший частично на другой вопрос, при этом обнаруживший полное знание учебного материала, предусмотренного программой; успешно выполнивший все задания, предусмотренные формами текущего контроля;

Оценка "удовлетворительно" выставляется студенту, ответившему полностью только на один вопрос и решившему одну задачу экзаменационного билета или допустившему погрешности в ответе на вопросы экзаменационного билета и обладающему необходимыми знаниями для их устранения под руководством преподавателя;

Оценка "неудовлетворительно" выставляется студенту, не ответившему на три теоретических вопроса, или ответившему на один вопрос и не решившему ни одной задачи экзаменационного билета, обнаружившему пробелы в знании основного материала, предусмотренного программой, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий; не выполнившему отдельные задания, предусмотренные формами текущего контроля.

Оценка за домашнюю работу:

Оценка «отлично» ставится, если:

- домашняя работа выполнена в полном объеме и соответствует заданию;
- пояснительная записка составлена аккуратно, последовательно с учетом требований стандартов по составлению текстовых документов;
- практическая часть домашней работы выполнена в полном объеме;
- защита домашней работы проведена грамотно с демонстрацией всех возможностей рассмотренных методов проектирования клиент-серверных приложений.

Оценка «хорошо» допускает:

- некоторые отступления от графика выполнения домашней работы;
- существование незначительных погрешностей в оформлении пояснительной записки и реализации методов разработки клиент-серверных приложений.
- недостаточно полными рекомендациями по формированию политики безопасности организации.

Оценка «удовлетворительно» допускает:

- существование ошибок, неточностей и непоследовательности при составлении пояснительной записки;
- значительные отступления от требований ЕСКД при выполнении пояснительной записки;
- отсутствие самостоятельности и творческого подхода при формулировке выводов;
- значительное отступление от сроков выполнения домашней работы;
- недостаточно грамотную защиту и неполную демонстрацию возможностей рассматриваемой методов разработки клиент-серверных приложений.

Оценка «неудовлетворительно» допускает:

- несоответствие домашней работы заданию;
- отсутствие учета требований стандартов по оформлению текстовых документов при составлении пояснительной записки;
- существование ошибок и непоследовательности в реализации методов разработки клиент-серверных приложений.
- значительное отступление от сроков выполнения курсовой работы (проекта);
- неспособность грамотно защитить домашнюю работу.

### **6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ**

#### **6.1. Рекомендуемая литература**

##### **6.1.1. Основная литература**

Авторы, составители	Заглавие	Библиотека	Издательство, год
---------------------	----------	------------	-------------------



	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л1.1	Маркин А. В., Шкарин С. С.	Основы web- программирования на PHP: учебное пособие	Электронная библиотека	Москва: Диалог-МИФИ, 2012

### 6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л2.1	Диков А. В.	Веб-технологии HTML и CSS: учебное пособие	Электронная библиотека	Москва: Директ-Медиа, 2012
Л2.2	Савельева Н. В.	Язык программирования PHP	Электронная библиотека	Москва: Национальный Открытый Университет «ИНТУИТ», 2016
Л2.3	Сорокин А. А.	Объектно-ориентированное программирование: учебное пособие: курс лекций	Электронная библиотека	Ставрополь: Северо- Кавказский Федеральный университет (СКФУ), 2014
Л2.4	Костюк А. И., Гушанский С. М., Поленов М. Ю., Катаев Б. В.	Информационные технологии. HTML и XHTML: учебное пособие	Электронная библиотека	Таганрог: Южный федеральный университет, 2015

### 6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Э1	Codenet	<a href="http://www.codenet.ru/">http://www.codenet.ru/</a>
Э2	Документация по Си#	<a href="https://docs.microsoft.com/ru-ru/dotnet/csharp/">https://docs.microsoft.com/ru-ru/dotnet/csharp/</a>
Э3	Руководство по PHP	<a href="https://www.php.net/manual/ru/index.php">https://www.php.net/manual/ru/index.php</a>

### 6.3 Перечень программного обеспечения

П.1	Microsoft Visual Studio 2015
П.2	Microsoft SQL server 2016
П.3	Microsoft Office
П.4	LMS Canvas
П.5	MS Teams

### 6.4. Перечень информационных справочных систем и профессиональных баз данных

И.1	Полнотекстовые российские научные журналы и статьи:
И.2	— Научная электронная библиотека eLIBRARY <a href="https://elibrary.ru/">https://elibrary.ru/</a>
И.3	Иностранные базы данных (доступ с IP адресов МИСиС):
И.4	— аналитическая база (индексы цитирования) Web of Science <a href="https://apps.webofknowledge.com">https://apps.webofknowledge.com</a>
И.5	— аналитическая база (индексы цитирования) Scopus <a href="https://www.scopus.com/">https://www.scopus.com/</a>

## 7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Ауд.	Назначение	Оснащение
Л-728	Учебная аудитория/ Компьютерный класс:	доска аудиторная меловая, экран проекционный, проектор, стационарные компьютеры 15 шт. ПО-Visual Studio; Electronic WorkBench; APACHE; MySQL; XAMPP; Python; комплект учебной мебели, пакет лицензионных программ MS Office
Л-731	Учебная аудитория/ Компьютерный класс:	доска аудиторная меловая, экран проекционный, проектор, стационарные компьютеры 15 шт. ПО-Visual Studio; Electronic WorkBench; APACHE; MySQL; XAMPP; Python, комплект учебной мебели, пакет лицензионных программ MS Office
Читальный зал электронных ресурсов		комплект учебной мебели на 55 мест для обучающихся, 50 ПК с доступом к ИТС «Интернет», ЭИОС университета через личный кабинет на платформе LMS Canvas, лицензионные программы MS Office, MS Teams, ESET Antivirus.
Любой корпус Мультимедийная	Учебная аудитория для проведения занятий лекционного типа и/или для проведения практических занятий:	комплект учебной мебели до 36 мест для обучающихся, мультимедийное оборудование, магнитно-маркерная доска, рабочее место преподавателя, ПКс доступом к ИТС «Интернет», ЭИОС университета через личный кабинет на платформе LMS Canvas, лицензионные программы MS Office, MS Teams, ESET Antivirus

## 8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

Учебно-методическое сопровождение дисциплины реализовано с применением ЭИОС «Canvas» в котором размещаются следующие материалы:

- программа дисциплины;
- методические рекомендации по выполнению курсовой работы;
- учебные, методические и дополнительные материалы;
- образцы отчетов;
- требования к отчетам по лабораторным работам.

### КОНТРОЛЬ КАЧЕСТВА ОСВОЕНИЯ ДИСЦИПЛИНЫ

Контроль качества полученных компетенций при освоении дисциплины проводится в форме текущего контроля успеваемости и на его основе промежуточной аттестации.

Для аттестации обучающихся на соответствие их персональных достижений требованиям образовательной программы используются оценочные средства текущего контроля успеваемости.

Оценка качества подготовки обучающихся проводится с целью оценки уровня сформированности компетенций.