МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ «ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

Рабочая программа профессионального модуля

ПМ.09 «ПРОЕКТИРОВАНИЕ, РАЗРАБОТКА И ОПТИМИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЙ»

для специальности

09.02.07 «Информационные системы и программирование»

Екатеринбург

2025

Чистяков Валерий Николаевич
я подтверждаю этот документ своей удостоверяющей подписью
место подписания
2026.04.08 17:06:01+05'00'

Рабочая программа профессионального модуля ПМ.09 «Проектирование, разработка и оптимизация веб-приложений» разработана на основе федерального государственного образовательного стандарта среднего профессионального образования по специальности **09.02.07 «Информационные системы и программирование».**

Организация-разработчик: ГАПОУ СО «Екатеринбургский монтажный колледж»

Разработчики:
Софьина Н.А., Кузьмин В.П., Черноскутов Д.А. - преподаватели ГАПОУ СО «Екатеринбургский монтажный колледж»

Рассмотрена методическим объединением Информационных технологий
Руководитель МО _____Софьина Н.А.
7.05.25

УТВЕРЖДАЮ
Заместитель директора по учебной работе
_____ Л.С.Хоринова
«13» мая 2025 г.

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ ПМ09. «ПРОЕКТИРОВАНИЕ, РАЗРАБОТКА И ОПТИМИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЙ»

1.1. ОБЛАСТЬ ПРИМЕНЕНИЯ РАБОЧЕЙ ПРОГРАММЫ

Рабочая программа профессионального модуля является частью основной профессиональной образовательной программы среднего профессионального образования подготовки специалистов среднего звена по специальности 09.02.07 «Информационные системы и программирование». Квалификация выпускника – Разработчик веб и мультимедийных приложений .

1.2. ЦЕЛЬ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

В результате изучения профессионального модуля обучающиеся должны освоить основной вид деятельности «Проектирование, разработкаи оптимизация веб - приложений» и соответствующие ему профессиональные и общие компетенции.

Выпускник, освоивший образовательную программу, должен обладать следующими компетенциями:

1.2.1. Перечень общих компетенций

Код	**Наименование общих компетенций**
ОК 01.	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам;
ОК 02.	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;
ОК 03.	Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях;
ОК 04.	Эффективно взаимодействовать и работать в коллективе и команде;
ОК 05.	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста;
ОК 09.	Пользоваться профессиональной документацией на государственном и иностранном языках. (п. 3.2 в ред. Приказа Минпросвещения России от 01.09.2022 N 796)

1.2.2. Перечень профессиональных компетенций

Код	**Наименование видов деятельности и профессиональных компетенций**
ВД 9.	Проектирование, разработка и оптимизация веб - приложений
ПК 9.1.	Разрабатывать техническое задание на веб-приложение в соответствии с требованиями заказчика.
ПК 9.2.	Разрабатывать веб-приложение в соответствии с техническим заданием.
ПК 9.3.	Разрабатывать интерфейс пользователя веб-приложений в соответствии с техническим заданием.
ПК 9.4.	Осуществлять техническое сопровождение и восстановление веб-приложений всоответствии с техническим заданием.
ПК 9.5.	Производить тестирование разработанного веб приложения.
ПК 9.6.	Размещать веб приложения в сети в соответствии с техническим заданием.
ПК 9.7.	Осуществлять сбор статистической информации о работе веб-приложений для анализа эффективности его работы.

ПК 9.8	Осуществлять аудит безопасности веб-приложения в соответствии с регламентами по безопасности.
ПК 9.9	Модернизировать веб-приложение с учетом правил и норм подготовки информации для поисковых систем
ПК 9.10	Реализовывать мероприятия по продвижению веб-приложений в сетиИнтернет.

В результате освоения профессионального модуля студент должен:

Иметь практический опыт	– использовании специальных готовых технических решений при разработке веб-приложений; – выполнении разработки и проектирования информационных систем; – модернизации веб-приложений с учетом правил и норм подготовки информациидля поисковых систем; – реализации мероприятий по продвижению веб-приложений в сети Интернет.
уметь	– разрабатывать программный код клиентской и серверной части веб- приложений; – осуществлять оптимизацию веб-приложения с целью повышения его рейтинга в сети Интернет; – разрабатывать и проектировать информационные системы.
знать	– языки программирования и разметки для разработки клиентской и серверной части веб-приложений; – принципы функционирования поисковых сервисов и особенности оптимизации веб-приложений под них; – принципы проектирования и разработки информационных систем.

1.3. Количество часов на освоение рабочей программы профессионального модуля

Всего часов **996 часов**

Из них на освоение МДК: **564 часов**, на практики, в том числе учебную **108 часов** и производственную: **324 часов**

самостоятельная работа **112 часов**

2. СТРУКТУРА И СОДЕРЖАНИЕ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

2.1.Структура профессионального модуля

Коды профессиональных, общих компетенций	Наименование циклов, дисциплин, профессиональных модулей, МДК, практик	Объем образовательной программы в академических часах	Объем времени, отведенный на освоениемеждисциплинарного курса (курсов)					Практика всего, часов
			Обязательные аудиторныеучебные занятия			Самостоятельная работа		
			всего,часов	в т.ч. лабораторные и практические занятия, часов	Курсовых работ (проектов) *	Учебная	изводственная(если предусмотрена рассредоточенная практика)	
1	2	3	4	5	1	2	3	4
ПК 9.1-9.6, ОК 01- ОК 05, ОК 09	МДК. 09.01 Проектирование и разработка веб-приложений	**274**	**216**	112	30	**108**	**-**	**58**
ПК 9.7, ПК 9.9-9.10, ОК 01- ОК 05, ОК 09	МДК. 09.02. Оптимизация веб-приложений	**150**	**136**	52	30		**-**	**14**
ПК 9.8, ОК 01-ОК 05, ОК 09	МДК. 09.03 Обеспечение безопасности веб-приложений	**140**	**134**	80	-	**-**	**-**	**6**
ПК 9.1 - ПК 9.10 ОК 01- ОК 05, ОК 09	Производственная практика (по профилю специальности), часов	**180**					**324**	**-**
	Всего:	**744**	**486**	244	60	**108**	**324**	**78**

2.2. Тематический план и содержание профессионального модуля (ПМ)

Наименование разделов и тем профессионального модуля (ПМ), междисциплинарных курсов (МДК)	**Содержание учебного материала, лабораторные работы и практические занятия, внеаудиторная (самостоятельная) учебная работа обучающихся, курсовая работа (проект)**	**Объем часов**	**Коды компетенций, формированию которых способствует элемент программы**
Раздел 1. Проектирование и разработка веб-приложений			
МДК 09.01 Проектирование и разработка веб-приложений			
6 семестр			
Темы	**Содержание**	**38**	
Схема клиент-серверного взаимодействия. Веб-окружение (Apache+PHP+MySQL)	Рассмотрение структуры веб-запроса и ответа, URL, HTTP, установка и настройка связки Apache+PHP+MySQL.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Основы PHP: синтаксис, переменные и типы данных	Типы данных PHP, область видимости, приведение типов, вывод информации, базовые операции.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Управляющие конструкции и обработка ошибок	if/else, switch, циклы (for, while, foreach), break, continue, match, declare, goto. Исключения: try/catch/finally, throw.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Подключение модулей: include, require	Принцип модульности, отличие include и require, include_once/require_once.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Массивы в PHP	Индексные, ассоциативные и многомерные массивы, перебор, сортировка.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6

Формы и методы передачи данных	HTTP-методы GET и POST, суперглобальные массивы $_GET и $_POST, параметры URL.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Валидация и регулярные выражения	Проверка данных пользователя, фильтры PHP, регулярные выражения для e-mail, телефона и текста.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Cookie и сессии	Хранение состояния пользователя, суперглобальные массивы $_COOKIE и $_SESSION.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Работа с файлами в PHP	Чтение и запись файлов, загрузка файлов на сервер ($_FILES), работа с изображениями.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Проектирование базы данных для веб-приложения	Проектирование сущностей и связей, нормализация, пример БД «Интернет-магазин».	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Подключение PHP к MySQL	Подключение с помощью mysqli/PDO, обработка ошибок подключения.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Выборка и модификация данных из БД	SELECT, INSERT, UPDATE, DELETE через PHP, отображение данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Пагинация и лимитирование выборки	Ограничение количества записей на странице, OFFSET, кнопки «Показать ещё».	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Работа с изображениями	Хранение путей к файлам, загрузка, изменение размеров.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
XML в веб-приложениях	Структура XML, разбор XML в PHP, XSLT-преобразования.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
AJAX и динамическое обновление данных	Принцип асинхронных запросов, XMLHttpRequest, Fetch API, работа с JSON и XML.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
AJAX и БД: CRUD-	Получение, добавление, изменение, удаление данных через AJAX.	2	ОК 01- ОК 05,

операции			ОК 09, ПК 9.1-9.6
Обработка ошибок и отладка в веб-приложениях	try/catch в реальных сценариях, журналирование ошибок.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Промежуточный итог: архитектура простого веб-приложения	Обзор структуры: фронтенд, серверная часть, БД.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Темы	**Содержание практических занятий и лабораторных работ**	**72**	
Создание серверных сценариев на PHP	Написание простых скриптов, вывод текста, работа с переменными и базовыми операторами.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Переменные и операторы в PHP	Работа с типами данных, арифметическими и логическими операторами, присвоение значений, конкатенация строк.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Управляющие конструкции	Использование if/else, switch, циклов (for, while, foreach), break, continue, а также обработка исключений try/catch.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Массивы и циклы	Создание индексных и ассоциативных массивов, перебор элементов, сортировка, многомерные массивы.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Подключение модулей (include/require)	Разделение проекта на модули, подключение внешних файлов с помощью include, require, include_once, require_once.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Обработка данных форм (POST, GET)	Получение данных из форм, проверка и фильтрация, работа с суперглобальными массивами $_POST и $_GET.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Регулярные выражения для валидации	Проверка корректности e-mail, телефона, пароля, текста через регулярные выражения.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Загрузка файлов на сервер	Создание формы загрузки файлов, проверка типа и размера, сохранение на сервере ($_FILES).	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6

Работа с изображениями	Изменение размера, конвертация форматов, генерация миниатюр с помощью PHP.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Проектирование БД под мини-проект	Создание схемы базы данных, проектирование таблиц, связей, первичных и внешних ключей.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Подключение PHP к БД (PDO)	Подключение к базе данных через PDO, обработка ошибок, безопасное выполнение SQL-запросов.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Вывод данных из БД	Получение и отображение информации из таблиц в виде списка или таблицы.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Добавление новых записей (Create)	Создание формы добавления данных, сохранение в базу данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Редактирование записей (Update)	Обновление информации в базе через форму редактирования.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Удаление записей (Delete)	Удаление выбранных записей из базы данных через интерфейс.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Организация CRUD	Объединение операций Create, Read, Update, Delete в единый функционал для одной сущности.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Регистрация пользователя	Создание формы регистрации с проверкой данных, запись пользователя в БД, обработка ошибок.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Авторизация (cookie, session)	Реализация входа пользователя, хранение состояния через сессии и cookie.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Администрирование (добавление/удаление данных)	Создание панели администратора для управления контентом сайта.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Создание каталога	Формирование страницы каталога с динамическим выводом товаров из БД.	2	ОК 01- ОК 05,

товаров			ОК 09, ПК 9.1-9.6
Пагинация («Показать больше»)	Ограничение числа товаров на странице, динамическая подгрузка следующей порции данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Фасетное меню и фильтрация	Создание фильтров по категориям, цене, характеристикам, динамическое обновление списка товаров.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Карточка продукта (основные данные)	Вывод информации о товаре: название, описание, цена, характеристики.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Карточка продукта (похожие товары)	Отображение рекомендаций на основе категории или тегов.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Добавление изображений продукта	Загрузка и отображение изображений, обработка файлов через PHP.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Комментарии (добавление, вывод)	Создание формы для комментариев, хранение в БД, вывод списка комментариев.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
AJAX-запросы (обновление данных без перезагрузки)	Подгрузка данных с сервера асинхронно, обновление контента на странице без перезагрузки.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
AJAX-CRUD с БД	Реализация операций Create, Read, Update, Delete через AJAX-запросы к базе данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
XML-документы (создание и отображение)	Создание XML-файлов, вывод данных на странице через PHP.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Работа с XML RSS	Получение и обработка RSS-каналов, отображение на сайте.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Web-сервисы (SOAP, WSDL)	Вызов SOAP-сервисов, разбор WSDL, взаимодействие с удаленными веб-сервисами.	2	ОК 01- ОК 05, ОК 09, ПК

			9.1-9.6
JSON-обмен между PHP и JS	Обмен данными между клиентом и сервером через JSON, обработка на стороне PHP и JS.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Bootstrap — оформление форм и каталога	Применение Bootstrap для стилизации форм, таблиц и каталога товаров.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Мини-блог с комментариями	Создание блога с возможностью добавления и отображения комментариев, работа с БД и сессиями.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Подготовка приложения к публикации	Оптимизация кода, настройка конфигурации, проверка работы всех функций.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Публикация сайта на бесплатном хостинге	Размещение проекта на внешнем сервере, настройка домена, проверка работы сайта онлайн.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
	Экзамен	**6**	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Внеаудиторная (самостоятельная) учебная работа при изучении МДК 09.01 **Изучение PHP-функций и документации (4 ч)** Студенты самостоятельно изучают встроенные функции PHP, работу с массивами, строками, датой и временем. Формируется навык поиска информации в официальной документации и использования стандартных функций при разработке веб-приложений. **Практика работы с формами и суперглобальными массивами (4 ч)** Создание мини-приложений с формами, например, регистрация или обратная связь. Обработка данных через `$_POST` и `$_GET`. Закрепляется работа с параметрами URL и отправкой форм на сервер. **Работа с файлами и изображениями (4 ч)** Реализация загрузки, хранения и вывода файлов и изображений. Практика проверки типов и размеров загружаемых файлов, использование функций PHP для работы с файловой системой.		**28**	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6

Создание мини-проектов с CRUD на PHP (6 ч) Разработка простой системы заметок или мини-каталога товаров с операциями Create, Read, Update, Delete через PHP и MySQL. Формирование навыков взаимодействия с базой данных и проектирования логики приложения. **Валидация данных и регулярные выражения (2 ч)** Создание форм с проверкой корректности e-mail, телефонов, паролей и других данных. Использование регулярных выражений для фильтрации и обработки данных. **Работа с cookie и сессиями (2 ч)** Создание страниц с авторизацией, сохранение данных о пользователе через cookie и сессии, управление временем жизни сессий, защита данных пользователя. **AJAX-запросы (4 ч)** Настройка динамического обновления данных на странице без полной перезагрузки, например, лайки, комментарии или фильтрация данных. Практика работы с XMLHttpRequest и JSON. **Подготовка проекта к публикации (2 ч)** Настройка структуры файлов, тестирование функционала проекта, создание инструкций по запуску приложения на сервере или локальной машине.			
7 семестр			
Темы	**Содержание**	**22**	
MVC и фреймворки PHP	Разбор паттерна Model–View–Controller. Роль модели, вида и контроллера. Обзор популярных PHP-фреймворков, их преимущества и сферы применения.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Установка и структура фреймворка	Настройка окружения с помощью Composer, установка фреймворка. Структура проекта: каталоги моделей, контроллеров, видов, конфигураций.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Контроллеры и маршрутизация	Создание контроллеров, маршрутизация запросов, работа с URL. Обработка GET/POST-запросов, передача параметров в контроллеры.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Модели и работа с БД	Использование моделей для работы с данными. Active Record, миграции, создание связей между таблицами, работа с реляционными базами данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Шаблоны и	Подключение CSS/JS. Шаблонизаторы (Blade, Twig, встроенные механизмы). Разделение	2	ОК 01- ОК 05,

представления	логики и представления.		ОК 09, ПК 9.1-9.6
Формы и встроенная валидация	Создание форм, проверка данных на сервере и клиенте. Автоматическая валидация, защита от некорректных данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Аутентификация и авторизация (RBAC)	Реализация входа и регистрации пользователей. Управление правами доступа (Role-Based Access Control).	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
CRUD-приложения во фреймворке	Генерация базовых операций Create, Read, Update, Delete. Расширение функционала через контроллеры и модели.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Разработка API во фреймворке	Создание REST API, маршруты для API, работа с JSON. Документирование API через Swagger или Postman.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Оптимизация и безопасность веб-приложений	Кэширование данных, защита от XSS, SQL-инъекций, CSRF. Улучшение производительности и безопасности приложения.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Тестирование и публикация приложений	Создание unit-тестов, проверка работы основных функций. Деплой приложения на сервер, подготовка к публикации.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Темы	**Содержание практических занятий и лабораторных работ**	**40**	
Установка фреймворка (Yii/Laravel) и настройка окружения	Установить выбранный фреймворк с помощью Composer, настроить локальный сервер, подключить базу данных и проверить работу стартового проекта.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Создание первого контроллера и маршрутов	Создать контроллер для основной страницы, настроить маршруты (URL) и отобразить простое сообщение в браузере.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Работа с моделями и миграциями	Создать модель для работы с таблицей БД, использовать миграции для создания таблицы и начальных данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Создание связей между таблицами	Настроить связи «один ко многим» и «многие ко многим» между моделями, проверить выборку связанных данных.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6

Отображение данных в представлениях	Создать шаблон для отображения списка записей из базы данных, подключить стили и JS при необходимости.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Создание форм и проверка данных	Реализовать форму добавления/редактирования данных, встроенную валидацию и обработку ошибок.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Реализация аутентификации (логин, регистрация)	Создать страницы регистрации и входа, шифрование пароля, проверку данных и хранение сессий.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Организация личного кабинета пользователя	Разработать страницу профиля с выводом персональных данных, возможностью редактирования.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
RBAC: настройка ролей и прав	Настроить роли и права пользователей, ограничить доступ к определённым страницам и функциям.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Генерация CRUD (например, модуль «Товары»)	С помощью генераторов фреймворка создать модуль CRUD, доработать контроллеры и формы.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Добавление изображений и работа с файловой системой	Реализовать загрузку изображений, проверку формата и размера, сохранение на сервере и отображение в приложении.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Создание ЧПУ-ссылок и маршрутов	Настроить человеко-понятные URL для страниц и категорий, изменить маршрутизацию для удобства.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Создание виджетов/компонентов интерфейса	Создать повторно используемые компоненты (виджеты) для меню, карточек товаров, форм и других элементов.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Поиск и фильтрация данных во фреймворке	Реализовать поиск и фильтры для данных, например по цене, категории, дате добавления.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Создание REST API (вывод данных в JSON)	Разработать API для выбранной сущности, выводить данные в формате JSON.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Добавление	Реализовать операции создания, редактирования и удаления данных через API-запросы.	2	ОК 01- ОК 05,

POST/PUT/DELETE-запросов к API (CRUD через API)			ОК 09, ПК 9.1-9.6
Документация API (Swagger/Postman)	Подготовить документацию API, протестировать эндпоинты через Postman или Swagger.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Оптимизация запросов и кэширование	Оптимизировать выборки данных, добавить кэширование запросов для повышения производительности.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Тестирование API и веб-приложения	Написать unit-тесты и функциональные тесты, проверить корректность работы всех модулей.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Итоговый проект: полноценное приложение (например, «Интернет-магазин» или «Каталог услуг») с API	Создать готовое веб-приложение с модулями CRUD, авторизацией, API, фильтрацией и загрузкой файлов.	2	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
	Экзамен	**8**	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6
Внеаудиторная (самостоятельная) учебная работа при изучении МДК 09.01 **Изучение фреймворка (Yii/Laravel) (4 ч)** Установка фреймворка, изучение структуры проекта, документации и инструментов генерации кода. Формируется понимание архитектуры MVC и стандартов работы с современными PHP-фреймворками. **Создание и настройка модулей (4 ч)** Практическое создание CRUD-модулей, работа с моделями, миграциями и связями таблиц. Студенты закрепляют навыки разработки логики приложения через готовые инструменты фреймворка.		**30**	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6

Формы и встроенная валидация (2 ч) Создание форм с автоматической валидацией через фреймворк, обработка ошибок ввода и защита от некорректных данных. **Аутентификация и RBAC (4 ч)** Реализация системы логина, регистрации пользователей, настройка ролей и прав доступа. Практика управления правами пользователей и разграничения функционала. **Работа с REST API (4 ч)** Разработка собственного API для приложения, вывод данных в формате JSON, тестирование запросов через Postman. Формируются навыки интеграции фронтенда и внешних приложений. **CRUD через API (4 ч)** Реализация операций создания, редактирования и удаления данных через API-запросы. Практика построения полноценного взаимодействия клиент-сервер через API. **Компоненты и виджеты (2 ч)** Создание повторно используемых элементов интерфейса, например, карточки товаров или меню. Использование шаблонов и встроенных механизмов фреймворка для удобного переиспользования кода. **Оптимизация и кэширование (2 ч)** Улучшение производительности запросов к базе данных и оптимизация работы приложения через кэширование данных. Практика анализа узких мест и ускорения работы веб-приложения. **Итоговый мини-проект (4 ч)** Разработка полноценного веб-приложения, например, интернет-магазина или каталога услуг, с использованием фреймворка, интегрированного API, форм, CRUD и компонентов интерфейса. Итоговый проект закрепляет все изученные навыки.		
Курсовой проект по МДК 09.01 **Глава 1. Аналитическая часть** • **Цель и задачи проекта**: описание назначения веб-приложения, какие проблемы решает, что должно быть реализовано с помощью фреймворка. • **Обзор аналогов**: сравнение существующих веб-приложений, выявление преимуществ и недостатков. • **Требования к приложению**:	**30**	ОК 01- ОК 05, ОК 09, ПК 9.1-9.6

- Функциональные: CRUD для основных сущностей, регистрация и авторизация пользователей, работа с файлами и изображениями, пагинация, фильтры, AJAX-запросы, REST API.
- Нефункциональные: безопасность, производительность, кроссбраузерность, удобство интерфейса.

- **Роли пользователей**: администратор, зарегистрированный пользователь, гость; описание прав доступа.
- **Выбор технологий**: PHP-фреймворк (Yii, Laravel или другой), база данных MySQL/PostgreSQL, JS, AJAX, XML/JSON, Docker/Nginx (по необходимости).

Глава 2. Проектная часть

- **Проектирование базы данных**: ER-диаграмма, таблицы, связи, ограничения; миграции фреймворка.
- **Архитектура приложения**:
  - Структура проекта по MVC (Model–View–Controller).
  - Описание директорий фреймворка: контроллеры, модели, представления, конфигурации.
  - Организация маршрутизации (routes).
- **Реализация функционала**:
  - Контроллеры: обработка запросов, формирование ответов.
  - Модели: работа с базой данных через Active Record или ORM.
  - Представления: отображение данных, подключение стилей и скриптов, шаблонизаторы (Blade/Twig или встроенные механизмы).
  - Формы и валидация: автоматическая проверка данных, защита от ошибок.
  - Аутентификация и авторизация (RBAC): регистрация, логин, управление ролями и правами.
  - CRUD-операции: создание, чтение, обновление, удаление данных.
  - Работа с файлами и изображениями: загрузка, хранение, отображение.
  - Пагинация и фильтры: динамическое отображение и сортировка данных.
  - AJAX-запросы и API: обмен данными с сервером без перезагрузки страницы, реализация REST API с JSON.
  - Оптимизация и безопасность: кэширование, защита от XSS, CSRF, SQL-инъекций.

Глава 3. Тестирование и отладка

- **Функциональное тестирование**: проверка корректной работы всех модулей, CRUD, форм, фильтров, API.
- **Безопасность**: тестирование на уязвимости (XSS, CSRF, SQL-инъекции), проверка прав доступа.
- **Отладка**: выявленные ошибки и способы их исправления.
- **Производительность**: оптимизация запросов к БД, кэширование данных, работа с большим количеством записей.

Заключение

- Выводы о реализованном веб-приложении.
- Достигнутые цели проекта.

• Возможные направления развития и улучшения проекта. **Список использованных источников** • Официальная документация фреймворка. • Книги и статьи по PHP, JavaScript, AJAX, XML/JSON, Docker/Nginx. • Примеры кода и учебные материалы. **Приложения (по необходимости)** • Код ключевых модулей или скриптов. • Скриншоты работы веб-приложения. • ER-диаграммы и схемы структуры проекта. • Примеры API-запросов и ответов.			
Учебная практика «Разработка игр» 1. Создание логики игры. Игровой цикл. От идеи до концепции. Узлы игрового дизайна. Определяем цели игры. Управляющие конструкции. Функции, массивы Основы ООП. Классы и объекты. Наследование. Исключения. 2. Базовые классы проекта. Создание настроек проекта, структуры проекта и движка. Разработка интерфейса, создание сцен. Дизайн уровней. Обозначаем пространство и неигровых персонажей. 3. Создание объектов игры. Прописываем главного героя. Описываем возможные действия героя и окружения. Даём игрокувыбор. 4. Создание интерактивных предметов и объектов (инвентарь персонажа) 5. Реализация логики поведения объектов. Работа с анимацией. Движение на основе физики. Вращение и направление 6. Обработка нажатий клавиш. . Обработка ввода, перемещение персонажа 7. Разработка игрового движка. Создание скриптов 8. Добавление аудио (звуки и музыка). 9. Работа с базой данных. Создание авторизации. Формирование рейтинга игрока. Сохранение игрового процесса 10. Тестирование игры 11. Работа с Git		**108**	ОК 01- ОК 09, ПК 9.3, ПК 9.5
Раздел 2 Оптимизация веб-приложений		**150**	
МДК. 09.02 Оптимизация веб-приложений			
Темы	**Содержание**	**38**	
Методы оптимизации сайта и определение	Обзор методов оптимизации веб-сайтов и роли целевой аудитории в формировании стратегии оптимизации.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 -

целевой аудитории			10
Внутренняя оптимизация сайта и разработка структуры сайта и навигации	Основы внутренней SEO-оптимизации, создание логичной структуры и удобной навигации.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Виды контента и проверка уникальности текста	Типы контента для сайта и методы проверки уникальности текста для улучшения SEO.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Настройка тега title, description и биржи копирайтинга	Значение мета-тегов для SEO и использование бирж копирайтинга для качественного наполнения сайта.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Поиск маркерных запросов и сбор семантического ядра	Методики поиска ключевых запросов, составление семантического ядра для сайта.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Мета-теги, HTML и настройка ссылок, внутренних якорей	Работа с мета-тегами в HTML, организация внутренних ссылок и якорей для улучшения навигации и SEO.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Подключение инструментов аналитики и оптимизация изображений	Использование аналитических инструментов (Google Analytics, Яндекс.Метрика) и оптимизация изображений для ускорения загрузки.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Каталогизация семантического ядра и SEO-анализ текста	Организация ключевых слов в каталоги и анализ текстового контента с точки зрения SEO.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Анализ сайта с помощью PageSpeed Insights и оптимизация сайта	Использование PageSpeed Insights для оценки производительности и техническая оптимизация сайта.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Создание карты сайта и файла Robots.txt	Значение карты сайта и файла Robots.txt для SEO, настройка и создание этих файлов.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Настройка страницы 404 и работа с Яндекс.Вебмастер	Правильное оформление страницы 404, регистрация и работа с Яндекс.Вебмастер для мониторинга сайта.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10

Техническая оптимизация и дополнительные настройки Улучшение поведенческих факторов сайта	Углубленная техническая оптимизация, работа с кэшированием, компрессией, HTTP-заголовками. Методы улучшения взаимодействия пользователей с сайтом, повышение времени на сайте, снижение отказов. Практическое внедрение изменений.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Влияние хостинга и доменного имени на SEO	Как выбор хостинга и доменного имени влияет на продвижение сайта, ограничения и фильтры поисковых систем. Лекция с анализом примеров.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Основные метрики веб-приложений и аналитика пользовательского поведения	Понятия DAU, Average Session Length, монетизация, анализ пользовательских данных.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Создание и оптимизация страницы 404	Детальный разбор важности и дизайна страницы 404, особенности SEO.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Юзабилити и мобильная адаптивность сайта	Введение в юзабилити, адаптивный дизайн, влияние мобильной версии на SEO и поведенческие факторы.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Технические ошибки, редиректы и канонические теги	Обзор типичных ошибок, работа с 301 и 302 редиректами, использование канонических тегов для устранения дублирования.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Микроразметка (schema.org) и SEO для динамических сайтов	Использование микроразметки для улучшения сниппетов и SEO, особенности оптимизации SPA и динамических страниц.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Продвижение через социальные сети (SMM) и стратегия продвижения. Анализ поведенческих факторов и внешних ссылок конкурентов	Роль социальных сетей в продвижении сайта, основы создания стратегии продвижения и контент-маркетинга. Практическое планирование кампании. Анализ вовлеченности пользователей, сбор и оценка внешних ссылок конкурентов для построения собственной стратегии продвижения. Практическое исследование.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Темы	**Содержание практических занятий и лабораторных работ**	**42**	

Анализ целевой аудитории и определение целей сайта	Определение ЦА, целей сайта и задач для продвижения. Знакомство с аналитикой поведения аудитории.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Разработка структуры сайта и создание навигации	Составление структуры сайта, проектирование логики переходов, удобство пользовательского пути.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Проверка уникальности текста и подбор контента	Проверка и доработка текстов, работа с инструментами уникальности, анализ контента конкурентов.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Настройка мета-тегов title и description	Практическая настройка мета-информации для главной и внутренних страниц сайта, с анализом сниппетов.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Сбор и группировка семантического ядра	Поиск ключевых слов, работа с Wordstat, подбор по частотности и целевым страницам.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
HTML-мета-теги и внутренняя перелинковка	Добавление мета-тегов в HTML, настройка внутренних ссылок и якорей, улучшение навигации.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Подключение аналитики: Яндекс.Метрика, Google Analytics	Создание аккаунтов, подключение трекеров, отслеживание целей, событий, поведения пользователей.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
SEO-анализ текста и каталогизация ядра	Анализ текстов с SEO-позиций (вода, тошнота, частота ключей), логическая группировка ключей по темам.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
PageSpeed Insights: технический аудит	Использование инструментов оценки скорости сайта, устранение основных технических проблем.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Создание sitemap.xml и robots.txt	Генерация карты сайта, настройка файла robots.txt, ограничение индексации служебных страниц.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Настройка и оформление страницы 404	Создание полезной страницы 404, добавление элементов дизайна, проверка редиректов.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10

Работа в Яндекс.Вебмастере и Google Search Console	Регистрация сайта, загрузка карты сайта, проверка индексации, отслеживание ошибок и запросов.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Углублённая техническая оптимизация	Кэширование, сжатие CSS/JS, устранение дублирующего контента, настройка переадресаций.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Поведенческие факторы: анализ и улучшение	Анализ глубины просмотров, времени на сайте, проработка CTA и структуры страниц.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Влияние домена и хостинга. Работа с фильтрами	Оценка домена и хостинга на скорость и безопасность. Разбор Яндекс-фильтров и причин попадания.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Метрики: DAU, удержание, длина сессии, конверсии	Работа с основными метриками эффективности веб-приложений. Построение отчётов по вовлечённости.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Мобильная адаптивность и юзабилити	Анализ сайта на различных устройствах, выявление проблем, внесение улучшений по UX/UI.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Редиректы и канонические ссылки	Настройка 301/302 редиректов, устранение дублей через canonical, устранение битых ссылок.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Микроразметка и SEO для динамических страниц	Внедрение schema.org, Open Graph и JSON-LD. Работа с индексированием SPA-сайтов.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
SMM и стратегия контентного продвижения	Планирование публикаций, интеграция SMM с сайтом, добавление кнопок шеринга и соц. блоков.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Сбор статистики и подготовка финального SEO-отчёта	Сбор статистики по всем каналам, составление отчёта об эффективности: посещаемость, запросы, цели.	2	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Курсовой проект **Введение** Актуальность темы Цель и задачи проекта		**30**	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10

Обоснование выбора методов оптимизации ***Аналитическая часть*** **1. Анализ конкурентов и определение целевой аудитории** 1.1. Исследование конкурентов: анализ сильных и слабых сторон 1.2. Определение целевой аудитории: портрет пользователя и сценарии использования 1.3. Постановка задач на основе анализа **2. Проектирование структуры и компонентов программного продукта** 2.1. Общая информация о проекте *3.1. Подбор ключевых запросов: анализ поискового спроса* *3.2. Формирование, кластеризация семантического ядра* 2.2. Разработка структуры сайта и навигации 2.3. Сбор ссылок на страницы веб-приложения 2.4. Оформление и проверка уникальности (*SEO*) текстового контента 2.5. Разработка макетов: разметка страниц и дизайн веб-приложения ***Проектная часть*** **3. Настройка SEO-оптимизации проекта** 3.1. Заполнение *заголовков* и мета-тегов: Title, Description, Keywords, *Robots* 3.2. Настройка анкорных ссылок и внутренних якорей *3.3. Офрмление и настройка страницы 404, прописывание ошибки 404 в файле .htaccess* *3.4. Создание, наполнение файла Robots.txt* *3.5. Создание, наполнение файла sitemap.xml* *3.6. Разработка сниппета с применением разметки OpenGraph* 3.7. Подключение инструмента "Вебмастер" и обоснование его необходимости 3.8. Подключение аналитического инструмента: настройка Google Analytics / Яндекс.Метрика **4. Тестирование и анализ производительности** 4.1. Инструменты для анализа производительности сайта 4.2. Проверка скорости загрузки страниц (PageSpeed Insights, GTmetrix) 4.3. Оптимизация изображений и кода (CSS, JS) 4.4. Проверка на адаптивность и кроссбраузерность **5. Результаты оптимизации и рекомендации** 5.1. Анализ улучшений после внедрения оптимизаций 5.2. Рекомендации по поддержанию и дальнейшему развитию проекта 5.3. Выводы о достигнутых результатах **Заключение**		

Обобщение выполненной работы и описание полученных навыков **Список использованных источников** Перечень литературы, статей и ресурсов, использованных в работе **Приложения (если есть)** - Скриншоты страниц сайта до и после оптимизации - Пример снипета до и после использования разметки opengraph - Фрагменты кода, мета-тегов или структуры семантического ядра		
Внеаудиторная (самостоятельная) учебная работа при изучении МДК. 09.02 Оптимизация веб-приложений 1 Изучение дополнительной литературы и Интернет-ресурсов; 2 Подготовка презентационных материалов по темам занятий 3 Подготовительные работы: общий аудит сайта, анализ конкурентной среды, создание резервной копии сайта 4 Внутренняя SEO оптимизация сайта – Составление семантического ядра – Кластеризация ключевых слов и составление карты релевантности – Поиск и устранение дублей сайта, оптимизация основных тегов, поиск и удаление битых ссылок – Разработка CSS для корректного отображения сайта – Очистка и оптимизация кода для ускорения загрузки сайта – Оптимизация картинок Alt, Title для рисунков 5 Контент-маркетинг 6 Получение естественных ссылок, улучшение поведенческих факторов 7 Работа с соц.сетями 8 Поддержка и отчетность 9 Рекомендации по улучшению сайта 10 Реферат на тему: Преимущества и недостатки конструкторов сайта. 11 Кластеризация ключевых слов и составление карты релевантности 12 Подготовка презентации на тему: «Google web – master, преимущества и недостатки» 13 Запись в блокнот «оптимизатора» ключевых ссылок сайта 14 Создание таблицы классификацию факторов в зависимости от вида и назначения сайта 15 Получение естественных ссылок, улучшение поведенческих факторов 16 Составление алгоритма «Очистка и оптимизация кода для ускорения загрузки сайта» 17 Составить рекомендации по улучшению сайта. Подготовить план работа с соц. Сетями	**14**	ПК 9.7, ПК 9.9 - ПК 9.10, ОК.01 - 10
Раздел 3. Обеспечение безопасности веб-приложений	**140**	

МДК. 09.03 Обеспечение безопасности веб-приложений			
Темы	**Содержание**	**44**	
Введение в безопасность веб-приложений	Основные угрозы и уязвимости, их влияние на разработку и эксплуатацию приложений	2	ПК 9.8, ОК.01 - 10
Модель угроз и архитектура безопасности	Методы STRIDE и другие подходы, определение доверенных границ и проектирование защищённой архитектуры	2	ПК 9.8, ОК.01 - 10
Аутентификация и управление сессиями	Современные методы входа, безопасное хранение паролей, токены и защита от угона сессий	2	ПК 9.8, ОК.01 - 10
Авторизация и контроль доступа	Сравнение ролевых и атрибутных моделей, распространённые ошибки и реализация принципа наименьших привилегий	2	ПК 9.8, ОК.01 - 10
Owasp top 10	Разбор наиболее критичных уязвимостей, примеры их эксплуатации и рекомендации по предотвращению	2	ПК 9.8, ОК.01 - 10
Инъекции (SQL, nosql, Command Injection)	Механизмы атак, последствия для системы и защита через валидацию ввода и безопасные запросы	2	ПК 9.8, ОК.01 - 10
Межсайтовый скриптинг (XSS)	Отражённые, хранимые и DOM-атаки, методы их предотвращения с помощью экранирования и CSP	2	ПК 9.8, ОК.01 - 10
Межсайтовая подделка запроса (CSRF)	Техника обмана пользователя, примеры атак и защита с использованием токенов и samesite cookies		
Уязвимости в аутентификации	Brute force, утечка учётных данных, ошибки при сбросе пароля и роль многофакторной аутентификации	2	ПК 9.8, ОК.01 - 10
Небезопасная десериализация	Примеры атак в разных языках, возможность удалённого выполнения кода и переход на безопасные форматы обмена	2	ПК 9.8, ОК.01 - 10
Ошибки в настройках безопасности	Неверные конфигурации серверов и сервисов, роль HTTP-заголовков и минимизация поверхности атаки	2	ПК 9.8, ОК.01 - 10
Утечки чувствительных данных	Шифрование при хранении и передаче, TLS и HTTPS, а также защита персональной информации и ключей	2	ПК 9.8, ОК.01 - 10
Использование компонентов с уязвимостями	Риски старых библиотек и фреймворков, мониторинг зависимостей и своевременное обновление	2	ПК 9.8, ОК.01 - 10

Веб-API безопасность (REST, graphql)	Особенности защиты интерфейсов, управление ключами и токенами, лимиты запросов и предотвращение dos	2	ПК 9.8, ОК.01 - 10
Современные стандарты безопасности (CORS, CSP, SRI)	Защита от междоменных атак, ограничение источников данных и контроль целостности ресурсов	2	ПК 9.8, ОК.01 - 10
Безопасность клиентской части (SPA, javascript)	Риски работы с localstorage, особенности защиты приложений на React и Angular	2	ПК 9.8, ОК.01 - 10
Безопасность мобильных веб-приложений	Язвимости мобильных браузеров, угрозы MITM-атак и специфика OWASP Mobile Top 10	2	ПК 9.8, ОК.01 - 10
Тестирование безопасности веб-приложений	Подходы SAST, DAST, IAST, использование Burp Suite и OWASP ZAP для поиска уязвимостей	2	ПК 9.8, ОК.01 - 10
Devsecops и безопасность в CI/CD	Интеграция анализа безопасности в pipeline, управление секретами и автоматизация проверок	2	ПК 9.8, ОК.01 - 10
Социальная инженерия и фишинг	Методы психологического воздействия, типичные атаки и инструменты защиты пользователей	2	ПК 9.8, ОК.01 - 10
Логирование и мониторинг атак	Организация журналирования, использование SIEM и практика реагирования на инциденты	2	ПК 9.8, ОК.01 - 10
Будущее веб-безопасности	Концепция Zero Trust, роль искусственного интеллекта в атаках и защите, перспективные стандарты и практики	2	ПК 9.8, ОК.01 - 10
Темы	**Содержание практических занятий и лабораторных работ**	**70**	
Установка и настройка тестовой среды	развёртывание уязвимого веб-приложения (DVWA, Juice Shop) для отработки атак	2	ПК 9.8, ОК.01 - 10
Работа с Burp Suite	перехват и анализ HTTP-запросов и ответов, тестирование простых уязвимостей	2	ПК 9.8, ОК.01 - 10
SQL-инъекции в практике	поиск уязвимостей и эксплуатация с помощью ручных запросов и автоматизированных инструментов	2	ПК 9.8, ОК.01 - 10
NoSQL-инъекции	выявление уязвимостей в MongoDB и защита с использованием фильтрации и параметризации	2	ПК 9.8, ОК.01 - 10
Command Injection	выполнение системных команд через веб-формы и предотвращение подобных атак	2	ПК 9.8, ОК.01 - 10

XSS Reflected	проведение отражённой атаки XSS и применение контекстного экранирования	2	ПК 9.8, ОК.01 - 10
XSS Stored	внедрение вредоносного кода в поля ввода и защита с помощью фильтрации и CSP	2	ПК 9.8, ОК.01 - 10
DOM-based XSS	эксплуатация уязвимостей в клиентском JavaScript-коде и способы защиты	2	ПК 9.8, ОК.01 - 10
CSRF-атака	создание поддельной формы запроса и защита через токены и SameSite cookies	2	ПК 9.8, ОК.01 - 10
Brute Force атака	подбор паролей с использованием Hydra или Burp Intruder и защита с помощью капчи и rate limiting	2	ПК 9.8, ОК.01 - 10
Атака на сброс пароля	анализ небезопасных механизмов восстановления доступа и внедрение многофакторной защиты	2	ПК 9.8, ОК.01 - 10
Небезопасная десериализация	эксплуатация на примере Java/PHP и защита через проверку формата данных	2	ПК 9.8, ОК.01 - 10
Ошибки в конфигурации сервера	сканирование открытых портов, лишних сервисов и настройка безопасных заголовков HTTP	2	ПК 9.8, ОК.01 - 10
Перехват cookies и угон сессии	эксплуатация слабых cookies и настройка HttpOnly, Secure и SameSite	2	ПК 9.8, ОК.01 - 10
Инъекции в XML (XXE)	проведение атаки XXE и предотвращение с помощью безопасных парсеров	2	ПК 9.8, ОК.01 - 10
Тестирование безопасности REST API	работа с Postman и Burp Suite, проверка токенов и лимитов запросов	2	ПК 9.8, ОК.01 - 10
Атаки на GraphQL API	перебор полей, использование introspection и методы защиты	2	ПК 9.8, ОК.01 - 10
CORS-атака	эксплуатация небезопасных настроек и правильная настройка заголовков	2	ПК 9.8, ОК.01 - 10
CSP-политика	внедрение политики CSP и проверка её эффективности против XSS	2	ПК 9.8, ОК.01 - 10
Сканирование уязвимостей зависимостей	использование OWASP Dependency-Check и Snyk для анализа библиотек	2	ПК 9.8, ОК.01 - 10
Использование HTTPS и TLS	настройка сертификата, проверка устойчивости к downgrade-атакам	2	ПК 9.8, ОК.01 - 10
Защита паролей	хеширование с помощью bcrypt/Argon2 и проверка устойчивости к атакам перебора	2	ПК 9.8, ОК.01 - 10

OAuth 2.0 в действии	реализация авторизации через стороннего провайдера и анализ типичных ошибок	2	ПК 9.8, ОК.01 - 10
JWT-атаки	подделка токенов, ошибки в алгоритмах и защита через подписи и правильную валидацию	2	ПК 9.8, ОК.01 - 10
Хранение данных на клиенте	тестирование рисков localStorage/sessionStorage и переход на защищённые механизмы	2	ПК 9.8, ОК.01 - 10
MITM-атака в Wi-Fi	перехват трафика в небезопасной сети и защита через HTTPS и VPN	2	ПК 9.8, ОК.01 - 10
Логирование безопасности	организация записи событий входа, ошибок и подозрительных действий в приложении	2	ПК 9.8, ОК.01 - 10
Мониторинг атак	настройка оповещений и анализ журналов в SIEM-системе	2	ПК 9.8, ОК.01 - 10
DevSecOps-практика	интеграция проверки кода на уязвимости в CI/CD pipeline	2	ПК 9.8, ОК.01 - 10
Secrets Management	безопасное хранение ключей и паролей в системах Vault или Kubernetes secrets	2	ПК 9.8, ОК.01 - 10
DDoS-тестирование	имитация перегрузки сервера и защита через rate limiting и фильтрацию	2	ПК 9.8, ОК.01 - 10
Атака методом социальной инженерии	моделирование фишинговой рассылки и обучение методам защиты	2	ПК 9.8, ОК.01 - 10
Реверс-инжиниринг уязвимостей	анализ эксплойта и поиск причин уязвимости в коде	2	ПК 9.8, ОК.01 - 10
Bug Bounty симуляция	выполнение тестирования по правилам ответственного раскрытия уязвимостей	2	ПК 9.8, ОК.01 - 10
Инцидент-менеджмент	моделирование атаки и отработка плана реагирования и устранения последствий	2	ПК 9.8, ОК.01 - 10
Внеаудиторная (самостоятельная) учебная работа при изучении МДК 09.03 – Изучение дополнительной литературы и Интернет-ресурсов; – Подготовка презентационных материалов по темам занятий – Оформление отчетов по практическим и лабораторным работам – Аудит безопасности веб-приложения в соответствии с регламентами по безопасности: – Поиск уязвимостей серверных компонентов; – Поиск уязвимостей в веб-окружении сервера; – Проверка на удаленное выполнение произвольного кода; – Проверка на наличие инъекций (внедрение кода);		26	ПК 9.8, ОК.01 - 10

– Попытки обхода системы аутентификации веб-ресурса; – Проверка веб-ресурса на наличие «XSS» / «CSRF» уязвимостей; – Попытки перехватить привилегированные аккаунты (или сессии таких аккаунтов); – Попытки произвести Remote File Inclusion / Local File Inclusion; – Поиск компонентов с известными уязвимостями; – Проверка на перенаправление на другие сайты и открытые редиректы; – Сканирование директорий и файлов, используя перебор и «google hack»; – Анализ поисковых форм, форм регистраций, форм авторизации и т.д.; – Проверки ресурса на возможность открытого получения конфиденциальной и секретной информации; – Атаки класса «race condition»; – Внедрение XML-сущностей; – Подбор паролей.		
Производственная практика 1. Выполнение разработки и проектирования информационных систем 2. Модернизация веб-приложений с учетом правил и норм подготовки информации для поисковых систем 3. Реализации мероприятий по продвижению веб-приложений в сети Интернет 4. Разработка и внедрение алгоритмов защиты информационных систем	**324**	ОК 01- ОК 05, ОК 09, ПК 9.1-9.10
Всего:	**996**	

1. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

4.1. Требования к минимальному материально-техническому обеспечению

Реализация программы предполагает наличие **Лаборатории разработки веб-приложений.**

Оборудование лаборатории:

- Автоматизированные рабочие места на 10 обучающихся с конфигурацией: Core i3 или аналог, не менее 4GB ОЗУ, мышь, клавиатура;
- Автоматизированное рабочее место преподавателя: ноутбук с конфигурацией: Pentium® Dual-Core CPU 2.00GHz, оперативная память 4 Гб;
- Специализированная эргономичная мебель для работы за компьютером;
- Мультимедийное устройство вывода;
- Многофункциональное устройство (МФУ) формата А4;
- Программное обеспечение общего и профессионального назначения.

4.2. Информационное обеспечение обучения

Перечень используемых учебных изданий, Интернет-ресурсов, дополнительной литературы

Основные источники

1. Бенкен, Е.С. PHP, MySQL, XML: программирование для интернета / Е.С. Бенкен. – СПб: БХВ–Петербург, 2016. – 336 с.: ил.+CD–ROM
2. Ляпин, Д.А. PHP это просто. Начинаем с видеоуроков / Д.А. Ляпин, А.В. Никитин. – СПб: БХВ–Петербург, 2017. – 176 с.: ил.+CD–ROM.
3. Маркин, А.В. Основы web-программирования на PHP: учебное пособие / А.В. Маркин, С.С. Шкарин. – М: Диалог-МИФИ, 2016. – 252с.
4. Храмцов, П.Б. Основы Web-технологий: учебное пособие / П.Б. Храмцов, С.А. Брик, А.М. Русак, А.И. Сурин. – 3–е изд., испр. – М: Интернет–Университет Информационных технологий; БИНОМ. Лаборатория знаний, 2016. – 512с.
5. Федорова, Г.Н.., Рудаков, А.В. Технология разработки программных продуктов. Практикум: учебное пособие / Г.Н. Федорова, А.В. Рудаков. – М: .Academia, 2016. – 192 с.

Дополнительная литература

1. Дунаев, В. Самоучитель JavaScript / В. Дунаев. 2–е изд. – СПб: Питер, 2016. – 400с.
2. Кузнецов, М.В. PHP 5. Практика разработки Web-сайтов / М.В. Кузнецов, И.В. Симдянов, С. В. Голышев. – СПб: БХВ-Петербург, 2015. – 960 с.: ил.
3. Котеров, Д. PHP 5 в подлиннике / Д. Котеров, А. Костарев. – СПб: Символ – Плюс, 2015. – 1120 с., ил.
4. Немцова Т.И., Назарова Ю.В. Компьютерная графика и web–дизайн практикум по информатике: учеб. Пособие. Под ред. Л.Г. Гагариной – М: ИД «ФОРУМ»: ИНФРА–М, 2015. – 287с.
5. Рудаков, А.В. Технология разработки программных продуктов: учебник. / А.В. Рудаков. – М. : .Academia, 2015. – 208с.
6. Савельева, Н.В. Основы программирования на PHP: курс лекций./ Н.В. Савельева. – М.: ИНТУИТ.РУ «Интернет–университет информационных технологий», 2016. – 264 с.

Интернет-ресурсы

1. Журнал веб-дизайн – уголок профессионала. [Электронный ресурс] – режим доступа: http://www.webmagazine.biz.
2. Система федеральных образовательных порталов. Информационно – коммуникационные технологии в образовании. [Электронный ресурс] – режим доступа: http://www.ict.edu.ru.
3. Методы и средства инженерии программного обеспечения: Учебник. Автор/создатель Лавришева Е.М., Петрухин В.А. Единое окно доступа к образовательным ресурсам. http://window.edu.ru/catalog/pdf2txt/699/41699/18857
4. http://ru.wikipedia.org
5. http://www.pstut.ru/
6. htpp://www.php.spb.ru
7. http://www.javaportal.ru
8. От модели объектов - к модели классов. Единое окно доступа к образовательнымресурсам. http://real.tepkom.ru/Real_OM-CM_A.asp

4.3. Общие требования к организации образовательного процесса

Профессиональный модуль предназначен для реализации требований к минимуму содержания и уровню подготовки выпускников по специальности среднего профессионального образования **09.02.07 Информационные системы и программирование; квалификация выпускника – Разработчик веб и мультимедийных приложений**. Учебные занятия должны проводиться в специализированной лаборатории, которая оснащается современным оборудованием и инструментарием.

Теоретические положения должны подкрепляться практическими занятиями. Учебная практика проводится образовательным учреждением при освоении обучающимися профессиональных компетенций в рамках профессиональных модулей и могут реализовываться как концентрированно в несколько периодов, так и рассредоточено, чередуясь с теоретическими занятиями в рамках профессиональных модулей.

Консультационные мероприятия проходят в обязательном порядке, согласно учебного плана и дополнительно по желанию учащихся.

Освоению данного модуля предшествует изучение следующих учебных дисциплин и профессионального модуля:

- Информационные технологии
- Основы алгоритмизации и программирования
- Информационная безопасность
- ПМ.08 Разработка дизайна веб-приложений

4.4. Кадровое обеспечение образовательного процесса

Реализация образовательной программы обеспечивается дипломированными педагогическими кадрами, имеющими высшее профессиональное образование, **с**оответствующее профилю преподаваемого модуля **“**Разработка дизайна веб-приложений**”** и по специальности **09.02.07 Информационные системы и программирование; квалификация выпускника – Разработчик веб и мультимедийных приложений**. Опыт деятельности в организациях соответствующей профессиональной сферы является обязательным для преподавателей, отвечающих за освоение обучающимся профессионального модуля.

5. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ (ПО РАЗДЕЛАМ)

Код и наименование профессиональных и общих компетенций, формируемых в рамках модуля	Критерии оценки	Методы оценки
Раздел модуля 1 Проектирование и разработка веб-приложений		
ПК 9.1. Разрабатывать техническое задание на веб-приложение в соответствии с требованиями заказчика.	Оценка «**отлично**» - изучены требования заказчика по результатам анкет и интервью; изучены типовые решения, обосновано, выбрано и согласовано с заказчиком оптимальное решение; разработано и оформлено техническое задание в полном соответствии с рекомендациями стандартов; разделы технического задания изложены логично и технически грамотно. Оценка «**хорошо**» - изучены требования заказчика по результатам анкет и интервью; изучены типовые решения, выбрано и согласовано с заказчиком оптимальное решение; разработано и оформлено техническое задание в соответствии с рекомендациями стандартов; разделы технического задания изложены логично играмотно. Оценка «**удовлетворительно**» - изучены требования заказчика по результатам анкет и интервью; изучены типовые решения, выбранои согласовано с заказчиком одно решение; разработано и оформлено техническое заданиев соответствии с рекомендациями стандартов; разделы технического задания изложены грамотно.	Экзамен в форме собеседования: практическое задание по разработке технического задания на проектирование веб-приложения Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
ПК 9.2. Разрабатывать веб-приложение в соответствии с техническим заданием.	Оценка «**отлично**» - веб приложение разработано и корректно функционирует в полном соответствии с техническим заданиемв среде программирования с использованием открытых библиотек; приложение предварительно смоделировано (применены объектные модели); код оформлен в соответствии со стандартами кодирования. Оценка «**хорошо**» - веб приложение разработано и работоспособно в соответствиис техническим заданием в среде программирования с использованием открытых библиотек; приложение предварительно смоделировано; код оформлен	Экзамен в форме собеседования: практическое задание по разработке веб-приложения по предложенному техническому заданию. Защита отчетов по практическим и лабораторным работам

	в соответствии со стандартами кодирования. Оценка «**удовлетворительно**» - веб приложение разработано и работоспособно в соответствии с техническим заданием в среде программирования с использованием открытых библиотек; код оформлен с незначительными отклонениями от стандартовкодирования.	Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
ПК 9.3. Разрабатывать интерфейс пользователя веб-приложений в соответствии с техническим заданием.	Оценка «**отлично**» - интерфейс пользователя разработан и корректно функционирует в полном соответствии с техническим заданием; приложение предварительно смоделировано (применены объектные модели); использованы анимационные эффекты; код оформлен в соответствии со стандартами кодирования. Оценка «**хорошо**» - интерфейс пользователя разработан и функционирует в соответствии с техническим заданием; приложение предварительно смоделировано; использованы анимационные эффекты; код оформлен в соответствии со стандартами кодирования. Оценка «**удовлетворительно**» - интерфейс пользователя разработан и функционирует; приложение предварительно смоделировано; использованы анимационные эффекты; код оформлен с незначительными отклонениями от стандартов кодирования.	Экзамен в форме собеседования: практическое задание по разработке интерфейса пользователя веб - приложения Защита отчетов по практическим и лабораторнымработам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
ПК 9.4 Осуществлять техническое сопровождение и восстановление веб-приложений в соответствии с техническим заданием	Оценка «**отлично**» - установлено программное обеспечение для создания резервной копии веб – приложения, создана копия веб приложения, серверные данные зарезервированы, веб – приложение восстановлено из резервной копии (развернуто), веб-сервер настроен; работоспособность проверена, вывод о качестве сделан. Оценка «**хорошо**» - установлено программное обеспечение для создания резервной копии веб – приложения, создана копия веб приложения, серверные данные зарезервированы, веб – приложение восстановлено из резервной копии (развернуто), веб-сервер настроен без существенных замечаний; работоспособность проверена. Оценка «**удовлетворительно**» - создана копия веб приложения, серверные данные зарезервированы, веб – приложение восстановлено из резервной копии (развернуто), веб-сервер настроен без	Экзамен в форме собеседования: практическое задание по настройке веб-серверов, резервному копированию и восстановлению работы веб-приложений. Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видов

	существенных замечаний.	работ во время учебной/ производственной
ПК 9.5. Производить тестирование разработанного веб приложения	Оценка «**отлично**» - выполнено тестирование веб – приложения в соответствии с тест– планом; результаты тестирования сохранены в системе контроля версий; по результатам тестирования сделаны выводы и внесены предложения по рефакторингу кода; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий; сделаны выводы по результатам отладки. Оценка «**хорошо**» - выполнено тестирование веб – приложения в соответствии с тест– планом; результаты тестирования сохранены в системе контроля версий; по результатам тестирования сделаны выводы; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий; сделаны выводы по результатам отладки. Оценка «**удовлетворительно**» - выполнено тестирование веб – приложения в соответствиис тест– планом; результаты тестирования сохранены в системе контроля версий; выполнена отладка приложения; результаты отладки сохранены в системе контроля версий.	Экзамен в форме собеседования: практическое задание по тестированию и отладке веб – приложения по предложенному тест-плану. Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
ПК 9.6. Размещать веб приложения в сети в соответствиис техническим заданием	Оценка «**отлично**» - выполнен анализ характеристик доступных хостингов; проанализированы параметры размещаемоговеб – приложения выбран и обоснован оптимальный хостинг для размещения предложенного веб – приложения; предложенное веб – приложение опубликовано на выбранном хостинге, проверено качество функционирования, сделан вывод по результатам проверки. Оценка «**хорошо**» - выполнен анализ характеристик хостингов; проанализированы параметры размещаемого веб – приложения; выбран и обоснован оптимальный хостинг для размещения предложенного веб – приложения; предложенное веб – приложение опубликовано, проверено качество функционирования, сделан вывод по результатам проверки. Оценка «**удовлетворительно**» - перечислены возможные хостинги; указаны параметры размещаемого веб – приложения; выбран и хостинг для размещения предложенного веб – приложения; предложенное веб – приложение опубликовано, проверено качество	Экзамен в форме собеседования: практическое задание по размещению веб-приложения в сети Интернет Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной

	функционирования.	
Раздел модуля 2 Оптимизация веб-приложений		
ПК 9.7. Осуществлять сбор статистической информации о работе веб- приложений для анализа эффективности его работы.	Оценка «**отлично**» - приведены основные показатели работы веб-приложения и обоснованы способы их анализа; подключена и настроена система мониторинга работы веб-приложения и получены конкретные характеристики; полученные характеристики проанализированы, сделаны выводы о работе веб-приложения и внесены в отчет. Оценка «**хорошо**» - приведены основные показатели работы веб-приложения; подключена и настроена система мониторинга работы веб-приложения и получены конкретные характеристики; полученные характеристики проанализированы, сделаны выводы о работе веб-приложения и внесены в отчет. Оценка «**удовлетворительно**» - приведены основные показатели работы веб-приложения; подключена и настроена система мониторинга работы веб-приложения и получены конкретные характеристики; сделаны выводы о работе веб-приложения и внесены в отчет.	Экзамен в форме собеседования: практическое задание по анализу эффективности работы веб-приложения Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
ПК 9.9. Модернизировать веб-приложение с учетом правил и норм подготовки информации для поисковых систем.	Оценка «**отлично**» - проанализирован и модифицирован код веб-приложения с помощью системы администрирования; получен работоспособный вариант; проверена работоспособность кода и сделан вывод о результатах оптимизации. Оценка «**хорошо**» - проанализирован и модифицирован код веб-приложения с помощью системы администрирования; получен практически работоспособный вариант; проверена работоспособность кода и сделан вывод о результатах оптимизации. Оценка «**удовлетворительно**» - модифицирован код веб-приложения с помощью системы администрирования; получен работоспособный вариант с некоторыми недостатками; проверена работоспособность кода и сделан вывод орезультатах оптимизации.	Экзамен в форме собеседования: практическое задание по оптимизации веб-приложения с целью адаптации к новым версиям поисковых систем Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
ПК 9.10. Реализовывать мероприятия по продвижению веб-	Оценка «**отлично**» - выбрана с обоснованием выбора система мониторинга работы сайта; система подключена и настроена; настройки обоснованы; выполнен сбор статистики и пояснены его результаты; составлены	Экзамен в форме собеседования: практическое задание по подключению, настройке и

приложений в сети Интернет	оригинальные и грамотные тексты для ссылок для размещения на сайтах партнеров и в справочниках. Оценка «**хорошо**» - выбрана система мониторинга работы сайта; система подключена и настроена; настройки обоснованы; выполнен сбор статистики и пояснены его результаты; применен инструментарий для подбора ключевых словосочетаний; составлены грамотные тексты для ссылок для размещения на сайтах партнеров и в справочниках. Оценка «**удовлетворительно**» - система мониторинга работы сайта подключена и настроена; выполнен сбор статистики; составлены грамотные тексты для ссылок для размещения на сайтах партнеров и в справочниках.	применению системы мониторинга работы сайта. Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
Раздел модуля 3 Обеспечение безопасности веб-приложений		
ПК 9.8. Осуществлять аудит безопасности веб-приложения в соответствии с регламентами по безопасности	Оценка «**отлично**» - проанализированы источники угроз безопасности; проанализированы методы защиты доступа к данным и защиты кода; предложены и реализованы меры защиты; код сайта и папки проанализированы на предмет наличия вредоносных программ; сделаны выводы о безопасности. Оценка «**хорошо**» - проанализированы источники угроз безопасности; предложены и реализованы меры защиты; код сайта и папки проанализированы на предмет наличия вредоносных программ; сделаны выводы о безопасности. Оценка «**удовлетворительно**» - проанализированы источники угроз безопасности; предложены и реализованы меры защиты; код сайта и папки проанализированы на предмет наличия вредоносных программ.	Экзамен собеседования: практическое задание по обеспечению безопасности функционирования веб- приложения. Защита отчетов по практическим и лабораторным работам Экспертное наблюдение за выполнением различных видовработ во время учебной/ производственной
ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.	– обоснованность постановки цели, выбора и применения методов и способов решения профессиональных задач; - адекватная оценка и самооценка эффективности и качества выполнения профессиональных задач	Экспертное наблюдение за выполнением работ
ОК 02. Использовать современные средства поиска,	- использование различных источников, включая электронные ресурсы, медиаресурсы, Интернет-ресурсы, периодические издания по	

анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности;	специальности для решения профессиональных задач	

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях	- демонстрация ответственности за принятые решения - обоснованность самоанализа и коррекция результатов собственной работы;	
ОК 04. Эффективно взаимодействовать и работать в коллективе и команде	- взаимодействовать с обучающимися, преподавателями и мастерами в ходе обучения, с руководителями учебной и производственной практик; - обоснованность анализа работы членов команды (подчиненных)	
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	Демонстрировать грамотность устной и письменной речи, - ясность формулирования и изложения мыслей	
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения	- соблюдение норм поведения во время учебных занятий и прохождения учебной и производственной практик,	

ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях	- эффективное выполнение правил ТБ во время учебных занятий, при прохождении учебной и производственной практик; - демонстрация знаний и использование ресурсосберегающих технологий в профессиональной деятельности	

ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности	- эффективность использовать средств физической культуры для сохранения и укрепления здоровья при выполнении профессиональной деятельности.	
ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках. (п. 3.2 в ред. Приказа Минпросвещения России от 01.09.2022 N 796)	Понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), понимать тексты на базовые профессиональные темы Участвовать в диалогах на знакомые общие и профессиональные темы строить простые высказывания о себе и о своей профессиональной деятельности Кратко обосновывать и объяснить свои действия (текущие и планируемые) писать простые связные сообщения на знакомые или интересующие профессиональные темы Применять в профессиональной деятельности инструкций на государственном и иностранном языке. Вести общение на профессиональные темы	

6. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОФЕССИОНАЛЬНОГО МОДУЛЯ

Контроль и оценка результатов усвоения профессионального модуля осуществляется преподавателем в процессе проведения практических занятий, на основании оценочных листов учебной практики, а также выполнения обучающимися индивидуальных заданий в результате сдачи демонстрационного экзамена.

Результаты (освоенные профессиональные компетенции)	**Основные показатели оценки результата**	**Формы и методы контроля и оценки**
ПК 9.1. Разрабатывать техническое задание на веб-приложение в соответствии с требованиями заказчика.	– Проводить анкетирование. – Проводить интервьюирование. – Оформлять техническую документацию. – Осуществлять выбор одного из типовых решений. – Работать со специализированным программным обеспечением для планирования времени и организации работы с клиентами. – Осуществлять сбор предварительных данных длявыявления требований к веб-приложению. – Определять первоначальных требований заказчика к веб-приложению и возможности их реализации. – Подбирать оптимальных вариантов реализации задач и согласование их с заказчиком. – Оформлять техническое задание	Дифференцированный зачет
ПК 9.2. Разрабатывать веб-приложение в соответствии с техническим заданием.	– Разрабатывать программный код клиентской и серверной части веб-приложений. – Использовать язык разметки страниц веб-приложения. – Оформлять код программы в соответствии со стандартом кодирования. – Использовать объектные модели Веб-приложений и браузера. – Использовать открытые библиотеки (framework). – Использовать выбранную среду программирования и средства системы управления базами данных. – Осуществлять взаимодействие клиентской и серверной частей Веб-приложений. – Разрабатывать программный код приложений. – Кодировать на языках веб-программирования. – Разрабатывать базы данных.	
ПК 9.3. Разрабатывать интерфейс пользователя веб- приложений в соответствии с техническим заданием.	– Разрабатывать программный код – клиентской части Веб-приложений. – Оформлять код программы в соответствии со стандартом кодирования. – Использовать объектные модели – Веб-приложений и браузера. – Разрабатывать анимацию для Веб-приложений для повышения его доступности и визуальной привлекательности (Canvas).	

	– Действия – Разрабатывать интерфейс пользователя. – Разрабатывать анимационные эффекты	
ПК 9.4. Осуществлять техническое сопровождение и восстановление веб- приложений в соответствии с техническим заданием	– Подключать и настраивать системы мониторинга работы Веб-приложений и сбора статистики его использования. – Устанавливать и настраивать веб сервера, СУБД для организации работы веб-приложений. Работать с системами Helpdesk. – Выяснять из беседы с заказчиком и понимать причины возникших аварийных ситуаций с информационным ресурсом. – Анализировать и решать типовые запросы заказчиков. – Выполнять регламентные процедуры по резервированию данных. – Устанавливать прикладное программное обеспечение для резервирования веб-приложений – Устанавливать и настраивать веб-серверы, СУБД для организации работы веб-приложений. – Использовать инструментальные средства контроля версийи баз данных. – Проводить работы по резервному копированию веб приложений. – Выполнять регистрацию и обработку запросов Заказчика в службе технической поддержки.	
ПК 9.5. Производить тестирование разработанного веб приложения	– Выполнять отладку и тестирование программного кода (втом числе с использованием инструментальных средств). – Выполнять оптимизацию и рефакторинг программногокода. – Кодировать на скриптовых языках программирования; – Тестировать веб-приложения с использованием тест-планов. – Применять инструменты подготовки тестовых данных. – Выбирать и комбинировать техники тестирования веб приложений. – Работать с системами контроля версий в соответствии с регламентом использования системы контроля версий. – Выполнять проверку веб-приложения по техническому заданию. – Использовать инструментальные средства контроля версийи баз данных, учета дефектов. – Тестировать веб приложения с точки зрения логической целостности. – Тестировать интеграцию веб-приложения с внешними сервисами и учетными системами	

ПК 9.6. Размещать веб приложения в сети в соответствии с техническим заданием	– Выбирать хостинг в соответствии с параметрами веб-приложения. – Составлять сравнительную характеристику хостингов – Публиковать веб-приложения на базе хостинга в сети Интернет.	
ПК 9.7. Осуществлять сбор статистической информации о работе веб-приложений для анализа эффективности его работы.	– Подключать и настраивать системы мониторинга работы Веб-приложений и сбора статистики его использования. – Составлять отчет по основным показателям использования Веб-приложений (рейтинг, источники и поведение пользователей, конверсия и др.). – Реализовывать мероприятия по продвижению веб-приложений в сети Интернет. – Собирать и предварительно анализировать статистическую информацию о работе веб-приложений	
ПК 9.8. Осуществлять аудит безопасности веб-приложения в соответствии с регламентами по безопасности	– Осуществлять аудит безопасности веб-приложений. – Модифицировать веб-приложение с целью внедрения программного кода по обеспечению безопасности егоработы – Обеспечивать безопасную и бесперебойную работу	
ПК 9.9. Модернизировать веб- приложение с учетом правил и норм подготовки информации для поисковых систем.	– Модифицировать код веб-приложения в соответствии с требованиями и регламентами поисковых систем. – Размещать текстовую и графическую информацию на страницах веб-приложения. – Редактировать HTML-код с использованием систем администрирования. – Проверять HTML код на соответствие отраслевым стандартам. – Модернизировать веб-приложения для обеспечениядоступа к ним поисковых систем.	
ПК 9.10. Реализовывать мероприятия по продвижению веб-приложений в сети Интернет	– Подключать и настраивать системы мониторинга работы Веб-приложений и сбора статистики его использования. – Работать с системами продвижения веб приложений. – Публиковать информации о веб приложении вспециальных справочниках и каталогах. – Осуществлять подбор и анализ ключевых слов и фраз для соответствующей предметной области с использованием специализированных программных средств. – Составлять тексты, включающие ссылки на продвигаемый сайт, для размещения на сайтах партнеров – Реализовывать мероприятия по продвижению веб-приложений в сети Интернет. – Собирать и предварительно анализировать статистическую информацию о работе веб-приложений.	

Формы и методы контроля и оценки результатов обучения должны позволять проверять у обучающихся не только сформированность профессиональных компетенций, но и развитие общих компетенций и обеспечивающих их умений.

Результаты (освоенные общие компетенции)	**Основные показатели оценки результата**	**Формы и методы контроля и оценки**
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	– Распознавать задачу и/или проблему в профессиональном и/или социальном контексте; – Анализировать задачу и/или проблему и выделять её составные части; – Правильно определить и найти информацию, необходимую для решения задачи и/или проблемы; – Составить план действия, – Определить необходимые ресурсы; – Владеть актуальными методами работы в профессиональной и смежных сферах; – Реализовать составленный план; – Оценить результат и последствия своих действий (самостоятельно или с помощью наставника). – Распознавать сложные проблемы в знакомых ситуациях. Выделять сложные составные части проблемы и описывать её причины и ресурсы, необходимые для её решения в целом. Определять потребность в информации и предпринимать усилия для её поиска. – Выделять главные и альтернативные источники нужных ресурсов. Разрабатывать детальный план действий и придерживаться его. Качество результата, в целом, соответствует требованиям. – Оценивать результат своей работы, выделять в нём сильные и слабые стороны.	Демонстрационный экзамен
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	– Определять задачи поиска информации – Определять необходимые источники информации – Планировать процесс поиска – Структурировать получаемую информацию – Выделять наиболее значимое в перечнеинформации – Оценивать практическую значимость результатов поиска – Оформлять результаты поиска – Планировать информационный поиск из широкого набора источников, необходимого для выполнения профессиональных задач – Проводить анализ полученной информации, выделять в ней главные аспекты – Структурировать отобранную информацию в соответствии с параметрами поиска	

	– Интерпретировать полученную информацию в контексте профессиональной деятельности	
ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по финансовой грамотности в различных жизненных ситуациях	– Определять актуальность нормативно-правовой документации в профессиональной деятельности – Использовать актуальную нормативно-правовую документацию по профессии (специальности) – Применять современную научно профессиональную терминологию – Определять траекторию профессионального развития и самообразования	
ОК 04. Эффективно взаимодействовать и работать в коллективе и команде	– Уметь работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями. – Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения ими заданий.	
ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	– Излагать свои мысли на государственном языке – Оформлять документы – Грамотно устно и письменно излагать свои мыслипо профессиональной тематике на государственном языке – Проявлять толерантность в рабочем коллективе	
ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, в том числе с учетом гармонизации межнациональных и межрелигиозных отношений, применять стандарты антикоррупционного поведения	Понимать значимость своей профессии (специальности)в обществе.	
ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого	Соблюдает правила экологической безопасности при ведении профессиональной деятельности.	

производства, эффективно действовать в чрезвычайных ситуациях		
ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности	Использовать средства физической культуры для сохранения и укрепления своего здоровья.	

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках. (п. 3.2 в ред. Приказа Минпросвещения России от 01.09.2022 N 796)	– Понимать общий смысл четко произнесенных высказываний на известные темы (профессиональные и бытовые), – понимать тексты на базовые профессиональные темы – участвовать в диалогах на знакомые общие и профессиональные темы – строить простые высказывания о себе и о своей профессиональной деятельности – кратко обосновывать и объяснить свои действия (текущие и планируемые) – писать простые связные сообщения на знакомыеили интересующие профессиональные темы – Применять в профессиональной деятельности инструкций на государственном и иностранном языке. – Вести общение на профессиональные темы	

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
МДК 09.01 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЙ
для специальности
09.02.07 «Информационные системы и программирование»

Екатеринбург
2025

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Фонд оценочных средств (ФОС) предназначен для организации и контроля учебного процесса по МДК 09.01 «Проектирование и разработка веб-приложений», входящему в профессиональный цикл подготовки студентов по направлению разработки и поддержки веб-приложений.

ФОС включает перечень лекционных и практических тем, а также краткое содержание занятий, раскрывающее основное содержание и направленность работы обучающихся на каждом этапе.

Фонд разработан с целью:

- обеспечения прозрачности и системности оценки результатов обучения;
- помощи преподавателям при планировании и проведении занятий;
- информирования студентов о тематике, задачах и ожидаемых результатах по каждой теме курса;
- формирования базы заданий для текущего контроля знаний (создание CRUD-модулей, работа с базами данных, реализация аутентификации, создание REST API и др.);
- подготовки к промежуточной аттестации в форме экзамена и выполнению курсового проекта.

Материалы ФОС могут использоваться:

- при проведении текущего, рубежного и итогового контроля;
- для формирования заданий на курсовой проект;
- как основа для составления оценочных листов, чек-листов и других форм диагностики;
- при организации самостоятельной и проектной деятельности студентов.

Завершается изучение дисциплины экзаменом, а также выполнением курсового проекта, где обучающиеся демонстрируют способность применять полученные знания на практике — от проектирования базы данных и реализации MVC-структуры до разработки полноценного веб-приложения с функционалом CRUD, аутентификацией, фильтрацией данных и взаимодействием через API.

Фонд является важным инструментом обеспечения качества подготовки специалистов и соответствует требованиям ФГОС СПО.

6 семестр — Теоретический материал

1. Схема клиент-серверного взаимодействия. Веб-окружение (Apache+PHP+MySQL)

Изучается архитектура клиент-сервер, роль браузера и сервера, структура HTTP-запроса и HTTP-ответа, особенности URL. Рассматриваются возможности локального веб-окружения: Apache как сервер, PHP как язык серверной логики и MySQL как система управления базами данных. Также объясняется настройка связки Apache+PHP+MySQL для работы динамических веб-приложений.

2. Основы PHP: синтаксис, переменные и типы данных

Разбираются правила написания кода на PHP, область видимости переменных, правила именования. Изучаются основные типы данных: строки, числа, булевы значения, массивы. Рассматривается приведение типов, базовые арифметические и логические операции, функции вывода данных на страницу и особенности интерполяции строк.

3. Управляющие конструкции и обработка ошибок

Изучаются конструкции ветвления: `if/else`, `switch` и новые возможности `match`. Рассматриваются циклы: `for`, `while`, `foreach` и управляющие команды `break`, `continue`, `declare`, `goto`. Вводится обработка ошибок через `try/catch/finally`, генерация исключений с помощью `throw` и принципы обработки ошибок в реальных сценариях.

4. Подключение модулей: include, require

Объясняется принцип модульности кода и необходимость разделения программы на отдельные файлы. Рассматривается отличие `include` и `require`, использование `include_once` и `require_once` для предотвращения повторного подключения. Подчеркивается значение модульной структуры для масштабируемости проектов.

5. Массивы в PHP

Рассматриваются различные виды массивов: индексные, ассоциативные и многомерные. Изучаются методы перебора массивов (`foreach`, `for`) и встроенные функции для сортировки, фильтрации и объединения массивов. Объясняется, как правильно хранить и обрабатывать структурированные данные.

6. Формы и методы передачи данных

Изучаются HTTP-методы GET и POST, структура передачи данных через URL и тело запроса. Рассматриваются суперглобальные массивы `$_GET`, `$_POST`, `$_REQUEST` и их применение для получения данных от пользователя. Объясняется принцип безопасности передачи данных через формы.

7. Валидация и регулярные выражения

Объясняется необходимость проверки данных, полученных от пользователя. Рассматриваются встроенные функции PHP для фильтрации и валидации данных (`filter_var`). Изучаются регулярные выражения для проверки email, телефона, пароля и текста.

8. Cookie и сессии

Рассматриваются механизмы хранения состояния пользователя на сервере и клиенте: `$_COOKIE` и `$_SESSION`. Изучаются принципы управления сессиями, установка и удаление cookie, безопасное хранение данных о пользователе.

9. Работа с файлами в PHP

Изучаются операции чтения и записи файлов, функции для работы с текстовыми файлами и изображениями. Рассматривается загрузка файлов на сервер через `$_FILES`, проверка их типа и размера, безопасное сохранение.

10. Проектирование базы данных для веб-приложения

Разбираются принципы проектирования базы данных: нормализация, проектирование сущностей и связей, определение первичных и внешних ключей. Приводится пример проектирования БД для интернет-магазина или каталога услуг.

11. Подключение PHP к MySQL

Изучаются способы подключения к базе данных с помощью `mysqli` и `PDO`. Рассматривается обработка ошибок подключения и базовые принципы безопасного выполнения SQL-запросов.

12. Выборка и модификация данных из БД

Изучаются SQL-запросы: `SELECT`, `INSERT`, `UPDATE`, `DELETE`. Рассматривается отображение данных в веб-приложении и работа с результатами запросов через PHP.

13. Пагинация и лимитирование выборки

Объясняется необходимость ограничения количества записей на одной странице, принципы использования `LIMIT` и `OFFSET`. Рассматриваются методы реализации кнопок «Показать ещё» и постраничной навигации.

14. Работа с изображениями

Разбираются подходы к хранению путей к изображениям, загрузка и обработка файлов, изменение размеров и форматов изображений с помощью PHP.

15. XML в веб-приложениях

Изучается структура XML-документов, разбор XML с помощью PHP, применение XSLT-преобразований. Рассматривается использование XML для обмена данными между приложениями.

16. AJAX и динамическое обновление данных

Объясняется принцип асинхронных запросов, работа с `XMLHttpRequest` и `Fetch API`. Рассматривается обновление данных на странице без перезагрузки, взаимодействие с JSON и XML.

17. AJAX и БД: CRUD-операции

Изучается реализация операций Create, Read, Update и Delete через AJAX-запросы к серверу, обработка данных на стороне PHP и динамическое обновление интерфейса.

18. Обработка ошибок и отладка в веб-приложениях

Разбираются методы отладки и логирования ошибок, применение `try/catch` в реальных сценариях, настройка журналирования и мониторинга ошибок.

19. Промежуточный итог: архитектура простого веб-приложения

Обобщается пройденный материал: структура приложения, взаимодействие фронтенда с серверной частью, подключение к БД, организация CRUD-операций и динамических элементов.

6 семестр — Практические занятия

1. Создание серверных сценариев на PHP

Студенты пишут первые скрипты на PHP: вывод текста на страницу, работа с переменными и базовыми операторами. Осваивается запуск кода на локальном сервере, тестирование и отладка скриптов.

2. Переменные и операторы в PHP

Практическая работа с типами данных: строки, числа, булевы значения. Выполнение арифметических и логических операций, присвоение значений переменным, конкатенация строк. Формирование базовых вычислений и выражений.

3. Управляющие конструкции

Использование условных операторов `if/else`, `switch`. Работа с циклами: `for`, `while`, `foreach`, а также применение `break` и `continue`. Реализация обработки исключений через `try/catch`, тестирование поведения кода при ошибках.

4. Массивы и циклы

Создание и работа с индексными и ассоциативными массивами. Перебор элементов массивов через циклы, сортировка массивов, работа с многомерными массивами. Практика обработки больших наборов данных.

5. Подключение модулей (include/require)

Разделение проекта на отдельные модули. Подключение внешних файлов с помощью `include`, `require`, `include_once`, `require_once`. Проверка работы модульного кода и предотвращение повторного подключения.

6. Обработка данных форм (POST, GET)

Создание HTML-форм и получение данных на сервере через `$_POST` и `$_GET`. Проверка и фильтрация данных, предотвращение ошибок при вводе, освоение безопасной обработки пользовательского ввода.

7. Регулярные выражения для валидации

Применение регулярных выражений для проверки корректности email, телефона, пароля и текста. Практика написания и тестирования шаблонов для валидации данных.

8. Загрузка файлов на сервер

Создание формы загрузки файлов. Проверка типа и размера файлов, сохранение на сервере через массив `$_FILES`. Практика безопасной обработки пользовательских файлов.

9. Работа с изображениями

Изменение размера изображений, конвертация форматов, создание миниатюр. Практика обработки изображений средствами PHP и отображения их на страницах сайта.

10. Проектирование БД под мини-проект

Создание схемы базы данных, проектирование таблиц и связей, определение первичных и внешних ключей. Практика создания проекта с учетом структуры данных.

11. Подключение PHP к БД (PDO)

Настройка соединения с базой данных через PDO, обработка ошибок подключения. Безопасное выполнение SQL-запросов, защита от SQL-инъекций.

12. Вывод данных из БД

Получение данных из таблиц базы данных и вывод в виде списка или таблицы. Работа с результатами запросов, отображение информации на странице.

13. Добавление новых записей (Create)

Создание формы для добавления данных в базу. Обработка формы на сервере, сохранение новых записей и проверка корректности данных.

14. Редактирование записей (Update)

Разработка функционала редактирования данных. Создание форм редактирования, обновление информации в базе данных, тестирование изменений.

15. Удаление записей (Delete)

Реализация удаления выбранных записей через интерфейс. Обеспечение подтверждения удаления и безопасного удаления данных.

16. Организация CRUD

Объединение операций Create, Read, Update, Delete для одной сущности. Разработка полноценного интерфейса для управления данными.

17. Регистрация пользователя

Создание формы регистрации с проверкой введенных данных. Запись новых пользователей в базу, обработка ошибок и проверка уникальности данных.

18. Авторизация (cookie, session)

Реализация входа пользователя на сайт. Хранение состояния через сессии и cookie, создание функционала “Запомнить меня”.

19. Администрирование (добавление/удаление данных)

Создание панели администратора для управления контентом. Добавление, редактирование и удаление данных через интерфейс.

20. Создание каталога товаров

Формирование страницы каталога с динамическим выводом товаров из базы данных. Реализация структуры каталога и базовой навигации.

21. Пагинация («Показать больше»)

Ограничение числа товаров на странице. Реализация кнопки “Показать ещё” для подгрузки следующей порции данных.

22. Фасетное меню и фильтрация

Создание фильтров по категориям, цене, характеристикам. Динамическое обновление списка товаров при выборе фильтров.

23. Карточка продукта (основные данные)

Вывод основной информации о товаре: название, описание, цена, характеристики. Работа с динамическим контентом.

24. Карточка продукта (похожие товары)

Отображение рекомендаций на основе категории или тегов. Создание блока “Похожие товары” на странице продукта.

25. Добавление изображений продукта

Загрузка и отображение изображений товаров. Практика работы с `$_FILES` и отображением медиафайлов на странице.

26. Комментарии (добавление, вывод)

Создание формы для комментариев, сохранение их в базе данных. Отображение списка комментариев для каждого продукта.

27. AJAX-запросы (обновление данных без перезагрузки)

Реализация подгрузки данных с сервера асинхронно. Обновление контента на странице без перезагрузки, взаимодействие с PHP через AJAX.

28. AJAX-CRUD с БД

Создание функционала Create, Read, Update, Delete через AJAX-запросы к базе данных. Практика динамического управления данными на клиентской стороне.

29. XML-документы (создание и отображение)

Создание XML-файлов, вывод их содержимого на страницу через PHP. Работа с форматом XML для хранения и обмена данными.

30. Работа с XML RSS

Получение и обработка RSS-каналов, вывод новостей или статей на сайте. Практика парсинга и отображения внешних данных.

31. Web-сервисы (SOAP, WSDL)

Вызов SOAP-сервисов, разбор WSDL-файлов. Взаимодействие с удаленными веб-сервисами для получения и отправки данных.

32. JSON-обмен между PHP и JS

Обмен данными между клиентом и сервером через JSON. Работа с массивами и объектами на PHP и их обработка на JavaScript.

33. Bootstrap — оформление форм и каталога

Применение Bootstrap для стилизации форм, таблиц и каталога товаров. Практика создания красивого и удобного интерфейса.

34. Мини-блог с комментариями

Создание мини-блога с возможностью добавления и отображения комментариев. Работа с базой данных, сессиями и формами.

35. Подготовка приложения к публикации

Оптимизация кода, настройка конфигурации проекта. Проверка работы всех функций перед переносом на сервер.

36. Публикация сайта на бесплатном хостинге

Размещение проекта на внешнем сервере. Настройка домена, проверка корректной работы сайта онлайн.

Примерные вопросы к экзамену – 6 семестр

Теория

1. Что такое PHP и где он используется?
2. Как подключить PHP-скрипт к HTML-странице?
3. Какие типы данных поддерживает PHP?
4. Чем отличается переменная от константы в PHP?
5. Какие арифметические и логические операторы есть в PHP?
6. Как работают условные конструкции `if`, `else` и `switch`?
7. В чем разница между циклами `for`, `while` и `foreach`?
8. Что такое массивы в PHP? Чем отличаются индексные и ассоциативные массивы?
9. Как сортировать массивы в PHP?
10. Для чего используются функции `include`, `require`, `include_once`, `require_once`?
11. Какие существуют суперглобальные массивы PHP и для чего они нужны?
12. Как получать данные из форм методом POST и GET?
13. Как проверять корректность данных с помощью регулярных выражений?
14. Как реализовать загрузку файлов на сервер через PHP?
15. Какие функции PHP позволяют работать с изображениями?
16. Какие основные принципы проектирования базы данных?
17. Что такое PDO и зачем он используется в PHP?
18. Как выполнять безопасные SQL-запросы с использованием PDO?
19. Какие операции входят в CRUD?
20. Как реализовать регистрацию и авторизацию пользователей в PHP?
21. Что такое сессии и cookie?
22. Как создать административную панель для управления сайтом?
23. Какие методы существуют для организации пагинации?
24. Что такое фасетное меню и как оно реализуется на сайте?
25. Как выводить карточку товара с основной информацией и похожими товарами?
26. Как создавать и отображать комментарии на сайте?
27. В чем заключается работа с AJAX-запросами?
28. Как реализовать AJAX-CRUD с базой данных?
29. Что такое XML и как создавать XML-документы в PHP?
30. Как работать с RSS-каналами через PHP?
31. Что такое SOAP и WSDL?
32. Как обмениваться данными между PHP и JavaScript через JSON?
33. Как применять Bootstrap для оформления форм и каталога товаров?
34. Какие принципы необходимо учитывать при подготовке проекта к публикации?
35. Какие шаги нужно выполнить для публикации сайта на бесплатном хостинге?

Практические задания к экзамену

1. Создать PHP-скрипт, который выводит “Hello, World!”.
2. Объявить несколько переменных разных типов и вывести их значения на страницу.
3. Написать скрипт с арифметическими и логическими операциями.
4. Создать условную конструкцию для проверки возраста пользователя.

5. Написать цикл `for` для вывода чисел от 1 до 10.
6. Создать индексный массив и вывести все его элементы через `foreach`.
7. Создать ассоциативный массив с данными о пользователе и вывести их на экран.
8. Разделить проект на два файла и подключить их через `include` и `require`.
9. Создать HTML-форму и обработать данные методом POST.
10. Реализовать проверку email через регулярное выражение.
11. Создать форму для загрузки файла и сохранить его на сервере.
12. Изменить размер загруженного изображения и сохранить миниатюру.
13. Спроектировать простую базу данных для мини-проекта (3 таблицы).
14. Подключиться к базе данных через PDO и выполнить SELECT-запрос.
15. Создать форму для добавления записи в базу данных.
16. Создать форму для редактирования записи в базе данных.
17. Реализовать удаление записи из таблицы через интерфейс.
18. Организовать полный CRUD для сущности “Товары”.
19. Создать форму регистрации пользователя с проверкой данных.
20. Реализовать вход пользователя с сохранением состояния в сессии.
21. Создать панель администратора для управления товарами.
22. Вывести каталог товаров с динамическим списком из базы данных.
23. Реализовать пагинацию: отображать только 5 товаров на странице.
24. Добавить фильтры по категориям и цене (фасетное меню).
25. Создать карточку товара с основными данными.
26. Реализовать блок с похожими товарами на основе категории.
27. Сделать форму комментариев и вывести все комментарии на странице.
28. Подключить AJAX для обновления комментариев без перезагрузки страницы.
29. Реализовать AJAX-CRUD для добавления и удаления товаров.
30. Создать XML-документ с данными о товарах и вывести его на странице.
31. Получить RSS-канал и вывести новости на сайте.
32. Подключиться к SOAP-сервису и получить данные через WSDL.
33. Реализовать обмен данными между PHP и JS через JSON.
34. Стилизовать форму и каталог товаров с помощью Bootstrap.
35. Подготовить проект к публикации и разместить его на бесплатном хостинге.

7 семестр — Теоретический материал

1. MVC и фреймворки PHP

Теория:

- **MVC (Model–View–Controller)** — архитектурный паттерн, разделяющий приложение на три части:
    - **Model (Модель)** — отвечает за работу с данными, взаимодействие с базой данных, бизнес-логику.
    - **View (Вид)** — отвечает за отображение информации пользователю, визуальную часть.
    - **Controller (Контроллер)** — получает запрос от пользователя, обрабатывает его и вызывает модель и вид.
- **PHP-фреймворки** (Laravel, Symfony, Yii, CodeIgniter) позволяют ускорить разработку, упрощают маршрутизацию, работу с БД, безопасность и тестирование.
- Преимущества: стандартизация кода, повторное использование, встроенные инструменты безопасности, простота масштабирования.

2. Установка и структура фреймворка

Теория:

- **Установка фреймворка** чаще всего через Composer — менеджер зависимостей PHP.
- **Структура проекта** обычно включает:
  - `app/Models` — модели данных, работа с базой.
  - `app/Controllers` — контроллеры для обработки запросов.
  - `app/Views` — шаблоны для отображения.
  - `config/` — настройки приложения.
  - `routes/` — маршрутизация запросов.
- Поддержка автозагрузки классов, единая точка входа (`index.php`) для обработки всех запросов.

3. Контроллеры и маршрутизация

Теория:

- **Контроллеры** обрабатывают HTTP-запросы, получают данные от модели и передают их виду.
- **Маршрутизация** — сопоставление URL-запросов с методами контроллеров.
- Обработка параметров GET/POST, получение данных форм, динамическая маршрутизация.
- Возможность передачи переменных в методы контроллеров для гибкой обработки запросов.

4. Модели и работа с БД

Теория:

- **Модели** представляют данные и бизнес-логику приложения.
- **Active Record** — шаблон, где объект модели соответствует строке таблицы.
- **Миграции** — способ создавать и изменять структуру базы данных через код.
- **Связи между таблицами**:
  - `hasOne`, `hasMany`, `belongsTo`, `manyToMany`.
- Работа с SQL-запросами через ORM или PDO, возможность использования транзакций.

5. Шаблоны и представления

Теория:

- **Вид (View)** отвечает за отображение данных.
- **Шаблонизаторы** (Blade, Twig) позволяют разделять логику и визуальную часть: использование циклов, условий, включение подшаблонов.
- Подключение CSS и JS для стилизации и интерактивности.
- Преимущество шаблонизаторов — чистый код, защита от XSS, повторное использование компонентов.

6. Формы и встроенная валидация

Теория:

- **Создание форм** для ввода данных пользователем (регистрация, добавление товара и др.).
- **Валидация** проверяет корректность данных: обязательные поля, длина, формат email, числа.
- **Встроенные средства фреймворка** позволяют автоматизировать проверку данных и возвращать ошибки пользователю.
- Защита форм от некорректных данных и CSRF-токены для безопасности.

7. Аутентификация и авторизация (RBAC)

Теория:

- **Аутентификация** — процесс идентификации пользователя (вход/регистрация).
- **Авторизация** — управление правами доступа к разделам и функциям приложения.
- **RBAC (Role-Based Access Control)** — роли пользователей (администратор, пользователь, гость), которые определяют, какие действия разрешены.
- Хранение данных в сессиях и cookie для поддержания состояния пользователя.

8. CRUD-приложения во фреймворке

Теория:

- **CRUD** — основные операции с данными:
  - **Create** — создание записи.
  - **Read** — получение данных.
  - **Update** — обновление данных.
  - **Delete** — удаление записи.
- Реализация через модели и контроллеры фреймворка.
- Возможность расширения функционала: фильтры, пагинация, поиск.

9. Разработка API во фреймворке

Теория:

- **REST API** позволяет обмениваться данными между клиентом и сервером через HTTP.
- **Маршруты API** — отдельные URL для работы с ресурсами.
- Используется **JSON** для передачи данных.
- Документирование API через Swagger, Postman или встроенные инструменты.
- Позволяет создавать мобильные приложения и SPA, которые взаимодействуют с сервером.

10. Оптимизация и безопасность веб-приложений

Теория:

- **Оптимизация**: кэширование страниц и запросов, уменьшение количества запросов к базе данных, минимизация CSS/JS.
- **Безопасность**:
  - XSS — защита от внедрения скриптов.
  - SQL-инъекции — защита через подготовленные запросы.

    - CSRF — использование токенов для предотвращения подделки запросов.
- Улучшение производительности и стабильности приложения.

11. Тестирование и публикация приложений

Теория:

- **Unit-тесты** — проверка отдельных методов и функций модели и контроллера.
- **Тестирование функциональности**: проверка маршрутов, форм, API.
- **Деплой приложения**: загрузка проекта на сервер, настройка конфигурации, проверка работы на реальном хостинге.
- Подготовка к публикации включает оптимизацию кода, настройку безопасности, тестирование всех функций.

7 семестр — Практический материал

1. Установка фреймворка (Yii/Laravel) и настройка окружения

Практика:

- Установка выбранного фреймворка через Composer.
- Настройка локального сервера (XAMPP, OpenServer, Valet).
- Подключение базы данных (MySQL, MariaDB).
- Проверка работы стартового проекта через браузер.

2. Создание первого контроллера и маршрутов

Практика:

- Создание контроллера для основной страницы приложения.
- Настройка маршрутов (URL) для страниц.
- Отображение простого сообщения или шаблона на веб-странице.

3. Работа с моделями и миграциями

Практика:

- Создание модели для таблицы базы данных.
- Использование миграций для создания таблицы и начальных данных.
- Проверка корректности структуры таблицы через консоль или phpMyAdmin.

4. Создание связей между таблицами

Практика:

- Настройка связей «один ко многим» и «многие ко многим» между моделями.
- Реализация выборки связанных данных через методы моделей.

- Проверка корректности работы связей на практике.

5. Отображение данных в представлениях

Практика:

- Создание шаблона (View) для отображения списка записей из базы данных.
- Подключение CSS и JS для улучшения визуального оформления.
- Использование циклов и условий для динамического вывода данных.

6. Создание форм и проверка данных

Практика:

- Реализация форм добавления и редактирования данных.
- Встроенная серверная и клиентская валидация данных.
- Обработка ошибок и отображение сообщений пользователю.

7. Реализация аутентификации (логин, регистрация)

Практика:

- Создание страниц регистрации и входа.
- Шифрование паролей (bcrypt, hash).
- Проверка корректности данных и хранение сессий.

8. Организация личного кабинета пользователя

Практика:

- Разработка страницы профиля пользователя.
- Вывод персональных данных и возможность их редактирования.
- Ограничение доступа к профилю только авторизованным пользователям.

9. RBAC: настройка ролей и прав

Практика:

- Создание ролей пользователей (администратор, пользователь).
- Настройка прав доступа к страницам и функциям приложения.
- Ограничение функционала в зависимости от роли пользователя.

10. Генерация CRUD (например, модуль «Товары»)

Практика:

- Создание CRUD-модуля через встроенные генераторы фреймворка.
- Доработка контроллеров, форм и представлений для работы с сущностью.
- Проверка корректного добавления, редактирования, удаления и просмотра записей.

11. Добавление изображений и работа с файловой системой

Практика:

- Реализация загрузки изображений через формы.
- Проверка формата и размера файла.
- Сохранение файлов на сервер и вывод их в приложении.

12. Создание ЧПУ-ссылок и маршрутов

Практика:

- Настройка человеко-понятных URL для страниц и категорий.
- Изменение маршрутизации для удобства навигации.
- Проверка корректного отображения страниц по новым ссылкам.

13. Создание виджетов/компонентов интерфейса

Практика:

- Создание повторно используемых компонентов интерфейса (меню, карточки товаров, формы).
- Подключение виджетов к шаблонам страниц.
- Проверка работы компонентов на разных страницах.

14. Поиск и фильтрация данных во фреймворке

Практика:

- Реализация поиска по ключевым полям (название, категория, цена).
- Настройка фильтров для сортировки данных по параметрам.
- Динамическое обновление списка результатов.

15. Создание REST API (вывод данных в JSON)

Практика:

- Разработка REST API для выбранной сущности.
- Вывод данных в формате JSON для внешнего использования.
- Проверка корректности формата данных через браузер или Postman.

16. Добавление POST/PUT/DELETE-запросов к API (CRUD через API)

Практика:

- Реализация операций создания, редактирования и удаления данных через API-запросы.
- Обработка ошибок и возвращение статусов HTTP.

- Тестирование CRUD через Postman или AJAX-запросы.

17. Документация API (Swagger/Postman)

Практика:

- Подготовка документации API с описанием эндпоинтов, методов и формата данных.
- Тестирование API через Postman или Swagger UI.
- Проверка работы всех маршрутов и параметров.

18. Оптимизация запросов и кэширование

Практика:

- Оптимизация выборки данных с использованием методов ORM.
- Добавление кэширования запросов и страниц для повышения производительности.
- Проверка скорости отклика приложения до и после оптимизации.

19. Тестирование API и веб-приложения

Практика:

- Написание unit-тестов для моделей и контроллеров.
- Функциональное тестирование форм, маршрутов и API.
- Проверка корректности работы всех модулей и компонентов приложения.

20. Итоговый проект: полноценное приложение (например, «Интернет-магазин» или «Каталог услуг») с API

Практика:

- Создание готового веб-приложения с модулями CRUD.
- Реализация авторизации и личного кабинета пользователя.
- Настройка фильтров, загрузки файлов, отображения изображений.
- Разработка REST API для взаимодействия с клиентской частью.
- Проверка корректности работы всех функций и деплой на сервер.

Экзаменационные вопросы – 7 семестр (теория)

MVC и фреймворки PHP

1. Объясните принцип паттерна MVC.
2. Какова роль модели, вида и контроллера в MVC?

3. Назовите популярные PHP-фреймворки и их преимущества.
4. В каких случаях целесообразно использовать фреймворк?

Установка и структура фреймворка

5. Как установить фреймворк с помощью Composer?
6. Какие каталоги обычно присутствуют в проекте на Yii или Laravel?
7. Назначение папок models, controllers, views, config.
8. Как проверить корректность работы стартового проекта после установки?

Контроллеры и маршрутизация

9. Как создаются контроллеры во фреймворке?
10. Что такое маршрутизация (routing) и как она работает?
11. Как обрабатывать GET и POST запросы в контроллере?
12. Как передать параметры в метод контроллера через URL?

Модели и работа с БД

13. Для чего используются модели?
14. Что такое Active Record?
15. Что такое миграции и зачем они нужны?
16. Как создать связи между таблицами через модели?

Шаблоны и представления

17. Как подключить CSS и JS к представлению во фреймворке?
18. Что такое шаблонизаторы (Blade, Twig) и зачем они нужны?
19. Почему важно разделять логику и представление?

Формы и встроенная валидация

20. Как создать форму добавления/редактирования данных?
21. Какие типы валидации данных поддерживаются во фреймворках?
22. Как защитить формы от некорректных данных и XSS?

Аутентификация и авторизация (RBAC)

23. Как реализуется регистрация и вход пользователя?
24. Как хранить и шифровать пароли?
25. Что такое RBAC и как назначаются роли и права?

CRUD-приложения во фреймворке

26. Что такое CRUD и как он реализуется во фреймворках?
27. Как расширить функционал стандартного CRUD-модуля?
28. Как связать CRUD с базой данных через модель?

Разработка API во фреймворке

29. Что такое REST API и зачем оно нужно?

30. Как настроить маршруты для API?
31. В каком формате данные передаются через API и как их обработать?
32. Для чего используется документация API (Swagger, Postman)?

Оптимизация и безопасность веб-приложений

33. Как защитить приложение от SQL-инъекций и CSRF?
34. Какие способы кэширования данных существуют во фреймворках?
35. Как улучшить производительность веб-приложения?

Тестирование и публикация приложений

36. Что такое unit-тесты и функциональные тесты?
37. Как проверить работу всех модулей перед публикацией?
38. Какие шаги включены в деплой веб-приложения на сервер?

Практические задания к экзамену

Установка и настройка

1. Установите Laravel/Yii через Composer и настройте подключение к базе данных.
2. Проверьте работу стартового проекта через браузер.

Контроллеры и маршруты

3. Создайте контроллер для главной страницы с отображением текста.
4. Настройте маршрут для главной страницы и страницы «О проекте».

Модели и миграции

5. Создайте модель «Товар» и таблицу товаров через миграцию.
6. Добавьте начальные данные в таблицу с помощью миграций.

Связи между таблицами

7. Настройте связь «один ко многим» между категориями и товарами.
8. Настройте связь «многие ко многим» между пользователями и ролями.

Представления

9. Выведите список товаров в виде таблицы на странице.
10. Подключите CSS и JS для улучшения отображения данных.

Формы и валидация

11. Создайте форму добавления нового товара с валидацией.
12. Реализуйте обработку ошибок и отображение сообщений пользователю.

Аутентификация и RBAC

13. Создайте страницу регистрации и входа пользователя.
14. Настройте шифрование паролей и хранение сессий.
15. Настройте роли «Администратор» и «Пользователь» и ограничения доступа.

CRUD

16. Сгенерируйте CRUD-модуль для сущности «Товар».
17. Добавьте возможность редактировать и удалять товары через интерфейс.

Работа с файлами

18. Реализуйте загрузку изображения товара и проверку формата и размера.
19. Выведите изображение на странице товара.

ЧПУ-ссылки и маршруты

20. Настройте человеко-понятные URL для страниц категории и товара.

Виджеты/компоненты интерфейса

21. Создайте повторно используемый виджет карточки товара.
22. Подключите виджет на главной странице и в каталоге.

Поиск и фильтрация

23. Реализуйте поиск товаров по названию.
24. Настройте фильтры по цене и категории.

REST API

25. Создайте API для сущности «Товар» с выводом данных в JSON.
26. Добавьте POST, PUT и DELETE запросы для API.
27. Протестируйте API через Postman.

Документация и тестирование

28. Подготовьте документацию API в Swagger или Postman.
29. Напишите unit-тест для метода контроллера.
30. Проверьте корректность работы всех CRUD операций и API.

Оптимизация и кэширование

31. Оптимизируйте выборку данных для страницы каталога.
32. Добавьте кэширование запросов к базе данных.

Итоговый проект

33. Разработайте интернет-магазин или каталог услуг с авторизацией, CRUD и API.
34. Реализуйте загрузку изображений и отображение карточек товаров.
35. Настройте фильтры, поиск и REST API для приложения.

Курсовой проект

Глава 1. Аналитическая часть

- **Цель и задачи проекта**: описание назначения веб-приложения, какие проблемы решает, что должно быть реализовано с помощью фреймворка.
- **Обзор аналогов**: сравнение существующих веб-приложений, выявление преимуществ и недостатков.
- **Требования к приложению**:
  - Функциональные: CRUD для основных сущностей, регистрация и авторизация пользователей, работа с файлами и изображениями, пагинация, фильтры, AJAX-запросы, REST API.
  - Нефункциональные: безопасность, производительность, кроссбраузерность, удобство интерфейса.
- **Роли пользователей**: администратор, зарегистрированный пользователь, гость; описание прав доступа.
- **Выбор технологий**: PHP-фреймворк (Yii, Laravel или другой), база данных MySQL/PostgreSQL, JS, AJAX, XML/JSON, Docker/Nginx (по необходимости).

Глава 2. Проектная часть

- **Проектирование базы данных**: ER-диаграмма, таблицы, связи, ограничения; миграции фреймворка.
- **Архитектура приложения**:
  - Структура проекта по MVC (Model–View–Controller).
  - Описание директорий фреймворка: контроллеры, модели, представления, конфигурации.
  - Организация маршрутизации (routes).
- **Реализация функционала**:
  - Контроллеры: обработка запросов, формирование ответов.
  - Модели: работа с базой данных через Active Record или ORM.
  - Представления: отображение данных, подключение стилей и скриптов, шаблонизаторы (Blade/Twig или встроенные механизмы).
  - Формы и валидация: автоматическая проверка данных, защита от ошибок.
  - Аутентификация и авторизация (RBAC): регистрация, логин, управление ролями и правами.
  - CRUD-операции: создание, чтение, обновление, удаление данных.
  - Работа с файлами и изображениями: загрузка, хранение, отображение.
  - Пагинация и фильтры: динамическое отображение и сортировка данных.
  - AJAX-запросы и API: обмен данными с сервером без перезагрузки страницы, реализация REST API с JSON.
  - Оптимизация и безопасность: кэширование, защита от XSS, CSRF, SQL-инъекций.

Глава 3. Тестирование и отладка

- **Функциональное тестирование**: проверка корректной работы всех модулей, CRUD, форм, фильтров, API.
- **Безопасность**: тестирование на уязвимости (XSS, CSRF, SQL-инъекции), проверка прав доступа.
- **Отладка**: выявленные ошибки и способы их исправления.

- **Производительность**: оптимизация запросов к БД, кэширование данных, работа с большим количеством записей.

Заключение

- Выводы о реализованном веб-приложении.
- Достигнутые цели проекта.
- Возможные направления развития и улучшения проекта.

Список использованных источников

- Официальная документация фреймворка.
- Книги и статьи по PHP, JavaScript, AJAX, XML/JSON, Docker/Nginx.
- Примеры кода и учебные материалы.

Приложения (по необходимости)

- Код ключевых модулей или скриптов.
- Скриншоты работы веб-приложения.
- ER-диаграммы и схемы структуры проекта.
- Примеры API-запросов и ответов.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по курсовому проектированию
МДК 09.01 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЙ
для специальности
09.02.07 «Информационные системы и программирование»

Екатеринбург
2025

Пояснительная записка

Для выполнения курсового проекта по МДК 09.01 «Проектирование и разработка веб-приложений» были разработаны методические указания, которые включают:

1. **Аналитическую часть:** определение целей, задач, требований к веб-приложению, ролей пользователей и выбор технологий.
2. **Проектную часть:** проектирование базы данных (ER-диаграммы, миграции), организация структуры проекта по MVC, настройка маршрутов, реализация функционала (CRUD, формы, аутентификация, работа с файлами, фильтры, AJAX и REST API).
3. **Тестирование и отладку:** проверка корректной работы модулей, тестирование безопасности, оптимизация производительности, исправление ошибок.
4. **Итоговую часть:** подготовка выводов о достигнутых целях, описание возможных направлений развития проекта, оформление списка источников и приложений (код, скриншоты, схемы, примеры API).

Цель курсового проекта:

Создание веб-приложения с использованием современного PHP-фреймворка (Laravel или Yii) с целью формирования практических навыков проектирования, разработки, тестирования и публикации веб-приложений.

Практическая направленность проекта:
В ходе выполнения курсового проекта студенты демонстрируют:

- умение проектировать и реализовывать базы данных и веб-приложения;
- навыки работы с PHP-фреймворками, моделями, контроллерами и представлениями;
- опыт настройки форм, валидации, аутентификации и авторизации пользователей;
- навыки создания CRUD-модулей, работы с файлами и изображениями, реализации пагинации, фильтров и AJAX-запросов;
- умение разрабатывать REST API и обеспечивать безопасность веб-приложения;
- навыки тестирования, оптимизации и подготовки проекта к публикации.

Вывод:
Методические указания обеспечивают структурированное выполнение курсового проекта, позволяют систематизировать знания и практические навыки студентов по разработке современных веб-приложений, формируют компетенции, предусмотренные МДК 09.01.

1. Общие требования

1. Курсовой проект должен быть выполнен в соответствии с принципами **MVC** (Model–View–Controller).
2. Все функциональные возможности веб-приложения должны быть реализованы через выбранный фреймворк.
3. Приложение должно включать CRUD-операции, формы с валидацией, регистрацию/авторизацию пользователей, работу с файлами и изображениями, пагинацию, фильтры, AJAX-запросы и REST API.
4. В работе необходимо соблюдать требования к безопасности: защита от XSS, CSRF, SQL-инъекций.
5. Для базы данных рекомендуется использовать MySQL или PostgreSQL с применением миграций фреймворка.
6. Итоговый проект должен быть готов к запуску локально или на тестовом сервере.

2. Глава 1. Аналитическая часть

Цель и задачи проекта

- Описать назначение веб-приложения, какие задачи оно решает.
- Определить, какие функции должны быть реализованы через фреймворк.

Обзор аналогов

- Сравнить существующие веб-приложения по функционалу и интерфейсу.
- Выявить преимущества и недостатки аналогов.

Требования к приложению

- **Функциональные:** CRUD для основных сущностей, регистрация и авторизация, загрузка файлов, пагинация, фильтры, AJAX-запросы, REST API.
- **Нефункциональные:** безопасность, производительность, кроссбраузерность, удобство интерфейса.

Роли пользователей

- Определить роли (администратор, зарегистрированный пользователь, гость) и права доступа для каждой роли.

Выбор технологий

- PHP-фреймворк (Yii, Laravel или другой)
- База данных: MySQL/PostgreSQL
- Frontend: HTML, CSS, JavaScript, AJAX
- Форматы данных: JSON/XML
- Опционально: Docker/Nginx

3. Глава 2. Проектная часть

Проектирование базы данных

- Разработать ER-диаграмму.
- Описать таблицы, поля, связи, ограничения.
- Создать миграции фреймворка для автоматической генерации таблиц.

Архитектура приложения

- Структура проекта должна соответствовать MVC.
- Описать директории фреймворка: models, controllers, views, config.
- Настроить маршрутизацию (routes).

Реализация функционала

Контроллеры

- Обработка запросов (GET/POST).
- Формирование ответов.

Модели

- Работа с БД через Active Record/ORM.
- Создание связей между таблицами.

Представления

- Отображение данных.
- Подключение CSS/JS и использование шаблонизаторов (Blade/Twig).

Формы и валидация

- Автоматическая проверка данных.
- Обработка ошибок и защита от некорректного ввода.

Аутентификация и авторизация

- Реализовать регистрацию и логин пользователей.
- Настроить RBAC: роли и права.

CRUD-операции

- Создание, чтение, обновление и удаление данных для основных сущностей.

Работа с файлами и изображениями

- Загрузка файлов и изображений, проверка формата и размера.
- Хранение на сервере и отображение в приложении.

Пагинация и фильтры

- Динамическое отображение и сортировка данных.

AJAX и API

- Обмен данными без перезагрузки страницы.
- Реализация REST API с JSON.

Оптимизация и безопасность

- Кэширование данных.
- Защита от XSS, CSRF, SQL-инъекций.

4. Глава 3. Тестирование и отладка

Функциональное тестирование

- Проверить работу всех CRUD-модулей, форм, фильтров, API.

Тестирование безопасности

- Проверка на XSS, CSRF, SQL-инъекции.
- Проверка прав доступа для разных ролей.

Отладка

- Выявление ошибок и методы их устранения.

Производительность

- Оптимизация запросов к БД.
- Использование кэширования.

5. Заключение

- Сформулировать выводы о реализованном веб-приложении.
- Оценить достигнутые цели проекта.
- Указать возможные направления развития и улучшения приложения.

6. Список использованных источников

- Официальная документация фреймворка.
- Книги и статьи по PHP, JavaScript, AJAX, XML/JSON, Docker/Nginx.
- Примеры кода и учебные материалы.

7. Приложения (по необходимости)

- Код ключевых модулей или скриптов.
- Скриншоты работы приложения.
- ER-диаграммы и схемы структуры проекта.
- Примеры API-запросов и ответов.

Пошаговая инструкция по выполнению курсового проекта

Этап 1. Аналитическая часть

1. **Определите цель проекта**
   - Опишите назначение приложения (например: «Интернет-магазин», «Каталог услуг»).
   - Перечислите задачи, которые приложение должно решать.
2. **Изучите аналоги**
   - Найдите 2–3 похожих веб-приложения.
   - Сравните их функционал и интерфейс.
   - Определите сильные и слабые стороны.
3. **Определите требования к приложению**
   - **Функциональные:** CRUD, регистрация/логин, работа с файлами, фильтры, пагинация, AJAX, REST API.
   - **Нефункциональные:** безопасность, скорость, удобство интерфейса, кроссбраузерность.
4. **Определите роли пользователей**
   - Администратор, зарегистрированный пользователь, гость.
   - Опишите права доступа для каждой роли.
5. **Выберите технологии**
   - PHP-фреймворк: Yii или Laravel.
   - База данных: MySQL/PostgreSQL.
   - Frontend: HTML, CSS, JS, AJAX.
   - Дополнительно: Docker/Nginx для локального сервера.

Этап 2. Проектирование базы данных

1. Создайте **ER-диаграмму** основных сущностей (например: товары, категории, пользователи, заказы).
2. Определите **таблицы, поля и связи** (один-ко-многим, многие-ко-многим).
3. Создайте **миграции** во фреймворке для автоматического создания таблиц.

Этап 3. Настройка проекта

1. Установите фреймворк через **Composer**.
2. Настройте подключение к базе данных.
3. Проверьте работу стартового проекта (отображение «Hello World» через контроллер и view).
4. Ознакомьтесь со структурой директорий: controllers, models, views, config.

Этап 4. Реализация функционала

4.1 Контроллеры и маршруты

- Создайте контроллеры для основных страниц.

- Настройте маршруты (routes) для URL.
- Обрабатывайте GET/POST-запросы.

4.2 Модели

- Создайте модели для каждой таблицы БД.
- Настройте связи между моделями.
- Используйте Active Record/ORM для работы с БД.

4.3 Представления

- Создайте шаблоны для отображения данных.
- Подключите CSS и JS.
- Используйте шаблонизаторы (Blade, Twig или встроенные).

4.4 Формы и валидация

- Реализуйте формы добавления и редактирования данных.
- Настройте серверную и клиентскую валидацию.
- Обрабатывайте ошибки и некорректные данные.

4.5 Аутентификация и авторизация

- Реализуйте регистрацию и логин.
- Настройте RBAC: роли и права доступа.

4.6 CRUD

- Реализуйте создание, чтение, обновление, удаление данных.

4.7 Работа с файлами

- Реализуйте загрузку изображений.
- Проверяйте формат и размер файлов.
- Отображайте загруженные файлы в приложении.

4.8 Пагинация и фильтры

- Реализуйте динамическую пагинацию и фильтры.

4.9 AJAX и REST API

- Настройте обмен данными без перезагрузки страницы.
- Создайте API для основных сущностей с выводом в JSON.

4.10 Оптимизация и безопасность

- Настройте кэширование.
- Защитите приложение от XSS, CSRF, SQL-инъекций.

Этап 5. Тестирование и отладка

1. **Функциональное тестирование:**
   - Проверьте все CRUD-операции, формы, фильтры, API.
2. **Тестирование безопасности:**
   - Проверьте XSS, CSRF, SQL-инъекции.
   - Убедитесь, что роли пользователей имеют правильные права доступа.
3. **Производительность:**
   - Оптимизируйте запросы к базе данных.
   - Используйте кэширование для ускорения работы.
4. **Отладка:**
   - Исправьте ошибки, выявленные при тестировании.

Этап 6. Итоговый проект

1. Подготовьте **заключение**:
   - Что реализовано.
   - Какие цели достигнуты.
   - Возможные направления улучшения.
2. Добавьте **список использованных источников**.
3. В приложениях:
   - Код ключевых модулей.
   - Скриншоты работы приложения.
   - ER-диаграммы.
   - Примеры API-запросов и ответов.

Шпаргалка для курсового проекта

Этап 1. Аналитика

- Определить цель проекта (например, интернет-магазин).
- Составить список задач (CRUD, регистрация, фильтры, API).
- Изучить аналоги (выявить плюсы и минусы).
- Определить роли пользователей (админ, пользователь, гость).
- Выбрать технологии (Laravel/Yii, MySQL/PostgreSQL, JS, AJAX, Docker/Nginx).

Этап 2. Проектирование базы данных

- Создать ER-диаграмму сущностей.
- Определить таблицы и связи (один-ко-многим, многие-ко-многим).
- Настроить миграции фреймворка для автоматического создания таблиц.
- Добавить начальные данные (seeders).

Этап 3. Настройка проекта

- Установить фреймворк через Composer.
- Настроить подключение к БД в `.env` или конфиге.
- Проверить стартовый проект (отображение «Hello World»).
- Ознакомиться со структурой директорий: controllers, models, views, config.

Этап 4. Реализация функционала

4.1 Контроллеры и маршруты

- Создать основной контроллер (например `SiteController`).
- Настроить маршруты для страниц (`routes/web.php` или `config/routes.php`).
- Реализовать обработку GET/POST-запросов.

4.2 Модели

- Создать модели для каждой таблицы.
- Настроить связи моделей (`hasOne`, `hasMany`, `belongsToMany`).
- Реализовать запросы к БД через Active Record/ORM.

4.3 Представления

- Создать шаблоны для отображения данных.
- Подключить CSS и JS.
- Использовать Blade/Twig или встроенные шаблоны.

4.4 Формы и валидация

- Создать формы добавления и редактирования.
- Настроить серверную и клиентскую валидацию.
- Обрабатывать ошибки.

4.5 Аутентификация и RBAC

- Создать регистрацию и вход.
- Шифровать пароли.
- Настроить роли и права доступа.

4.6 CRUD

- Реализовать создание, чтение, обновление и удаление данных для всех сущностей.

4.7 Работа с файлами

- Реализовать загрузку изображений.
- Проверять формат и размер.
- Отображать загруженные файлы в приложении.

4.8 Пагинация и фильтры

- Добавить пагинацию.
- Настроить фильтры по ключевым параметрам (категория, цена, дата).

4.9 AJAX и REST API

- Настроить AJAX-запросы для динамического обновления страниц.
- Создать API для основных сущностей (вывод JSON).
- Реализовать POST/PUT/DELETE через API.

4.10 Оптимизация и безопасность

- Настроить кэширование запросов.
- Защитить приложение от XSS, CSRF, SQL-инъекций.

Этап 5. Тестирование

- Проверить CRUD и формы.
- Проверить фильтры, пагинацию и API.
- Проверить права доступа пользователей.
- Оптимизировать медленные запросы.
- Исправить ошибки и баги.

Этап 6. Итоговый проект

- Подготовить заключение (цели достигнуты, выводы, возможности развития).
- Добавить список источников (документация, книги, статьи).
- Приложения:
    - Код ключевых модулей
    - Скриншоты работы
    - ER-диаграммы
    - Примеры API-запросов и ответов

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
МДК 09.02 ОПТИМИЗАЦИЯ ВЕБ-ПРИЛОЖЕНИЙ
для специальности
09.02.07 «Информационные системы и программирование»

Екатеринбург
2025

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Фонд оценочных средств (ФОС) предназначен для организации и контроля учебного процесса по модулю **МДК 09.02 «Оптимизация веб-приложений»**, входящему в профессиональный цикл подготовки студентов по направлению разработки и поддержки веб-приложений.

ФОС включает перечень **лекционных и практических тем**, а также **краткое содержание занятий**, раскрывающее основное содержание и направленность работы обучающихся на каждом этапе.

Фонд разработан с целью:

- обеспечения **прозрачности и системности оценки результатов обучения**;
- помощи преподавателям при планировании и проведении занятий;
- информирования студентов о тематике, задачах и ожидаемых результатах по каждой теме курса;
- формирования **базы заданий для текущего контроля знаний** (практические работы, анализ, аудиты, настройка SEO-элементов и пр.);
- подготовки к **промежуточной аттестации в форме экзамена** и **выполнению курсового проекта**.

Материалы ФОС могут использоваться:

- при проведении **текущего, рубежного и итогового контроля**;
- для формирования заданий на **курсовой проект**;
- как основа для составления **оценочных листов, чек-листов** и других форм диагностики;
- при организации **самостоятельной и проектной деятельности** студентов.

Завершается изучение дисциплины **экзаменом**, а также **выполнением курсового проекта**, где обучающиеся демонстрируют способность применять полученные знания на практике — от проведения SEO-аудита до оптимизации производительности и повышения эффективности веб-приложения.

Фонд является важным инструментом обеспечения качества подготовки специалистов и соответствует требованиям ФГОС СПО.

Описание лекционных занятий

1. Методы оптимизации сайта и определение целевой аудитории

В этой теме рассматриваются основные подходы к оптимизации сайтов: SEO, техническая, контентная и UX-оптимизация. Также изучается процесс определения целевой аудитории: как сегментировать пользователей по интересам, демографии, поведению и как это влияет на стратегию продвижения сайта.

2. Внутренняя оптимизация сайта и разработка структуры сайта и навигации

Тема посвящена внутренней SEO-оптимизации: настройке URL, мета-тегов, заголовков, alt-атрибутов, внутренних ссылок. Рассматриваются принципы построения логичной структуры сайта и удобной навигации, которая помогает и пользователям, и поисковым системам.

3. Виды контента и проверка уникальности текста

Разбираются типы контента (текстовый, визуальный, видео, интерактивный) и их влияние на продвижение сайта. Уделяется внимание созданию уникального текста: почему это важно, как искать дубликаты, какими сервисами проверять уникальность.

4. Настройка тега title, description и биржи копирайтинга

Объясняется значение мета-тегов title и description для SEO и кликабельности в выдаче. Учатся формулировать заголовки и описания с учетом ключевых слов. Также рассматриваются биржи копирайтинга как инструмент для получения уникального контента.

5. Поиск маркерных запросов и сбор семантического ядра

Тема знакомит с понятием маркерных (ключевых) запросов, с их ролью в структуре сайта. Изучается процесс сбора семантического ядра: какие сервисы использовать, как группировать запросы, какие запросы приоритетны.

6. Мета-теги, HTML и настройка ссылок, внутренних якорей

Обсуждаются важные элементы HTML, влияющие на SEO: мета-теги, заголовки h1–h6, alt. Поясняется, как правильно использовать внутренние ссылки и якоря для улучшения структуры сайта и навигации.

7. Подключение инструментов аналитики и оптимизация изображений

Изучается подключение и настройка аналитических систем: Яндекс.Метрика, Google Analytics, их возможности и цели. Также рассматривается оптимизация изображений: выбор формата, сжатие, внедрение lazy-loading и CDN.

8. Каталогизация семантического ядра и SEO-анализ текста

Здесь рассматривается распределение семантического ядра по страницам, создание карты ключевых слов. Проводится SEO-анализ текстов: плотность ключей, стоп-слова, тошнота, переспам и другие параметры.

9. Анализ сайта с помощью PageSpeed Insights и оптимизация сайта

Разбирается инструмент Google PageSpeed Insights для анализа производительности сайта. Изучаются рекомендации и способы оптимизации скорости загрузки страниц, снижения CLS, FCP, LCP и других показателей Core Web Vitals.

10. Создание карты сайта и файла Robots.txt

Рассматривается значение карты сайта (sitemap.xml) и файла robots.txt для индексации. Изучается синтаксис, правила создания, подключение к вебмастерам и рекомендации по управлению доступом к страницам.

11. Настройка страницы 404 и работа с Яндекс.Вебмастер

Тема раскрывает роль страницы 404: как она влияет на SEO и пользовательский опыт. Объясняется работа с сервисом Яндекс.Вебмастер: анализ ошибок, управление индексацией, подключение сайта, проверка файлов.

12. Техническая оптимизация и дополнительные настройки

Обсуждаются вопросы ускорения сайта: сжатие CSS и JS, минификация, кэширование, работа с сервером и CDN. Также рассматриваются важные технические настройки: корректные редиректы, SSL-сертификаты, canonical-теги.

13. Улучшение поведенческих факторов сайта

Изучаются ключевые поведенческие метрики: глубина просмотра, время на сайте, показатель отказов. Уделяется внимание улучшению UX, созданию удобного интерфейса, понятной навигации и работе с вовлекающим контентом.

14. Влияние хостинга и доменного имени на SEO

Разбираются факторы хостинга: скорость сервера, стабильность, поддержка HTTP/2, наличие SSL. Также анализируется, как доменное имя (зона, история, возраст) может влиять на доверие поисковых систем и позиции сайта.

15. Основные метрики веб-приложений и аналитика пользовательского поведения

Рассматриваются основные показатели эффективности веб-приложения: DAU, MAU, Session Length, Bounce Rate, Retention. Изучаются методы анализа пользовательских сценариев и построение воронок поведения.

16. Создание и оптимизация страницы 404

Углубленное рассмотрение оформления 404 страницы: как превратить ошибку в полезный опыт. Варианты оформления, юмор, добавление навигации и ссылок, сбор обратной связи через 404.

17. Юзабилити и мобильная адаптивность сайта

Объясняется понятие юзабилити, принципы проектирования интерфейсов. Изучаются особенности мобильной адаптации: responsive-дизайн, тестирование на разных устройствах, Google Mobile-Friendly Test.

18. Технические ошибки, редиректы и канонические теги

Разбираются распространённые ошибки: дубли страниц, битые ссылки, циклические редиректы. Поясняется использование тегов canonical, правила настройки редиректов (301, 302) и их влияние на SEO.

19. Микроразметка (schema.org) и SEO для динамических сайтов

Изучается использование микроразметки для расширенных сниппетов: товары, статьи, рейтинги и др. Рассматриваются сложности индексации динамических сайтов, AJAX, SPA. Методы решения — prerendering, SSR и др.

20. Продвижение через социальные сети (SMM) и стратегия продвижения

Обсуждаются методы привлечения трафика через социальные сети. Создание SMM-стратегии, планирование контента, интеграция соцсетей с сайтом и влияние социальных сигналов на SEO.

21. Анализ поведенческих факторов и внешних ссылок конкурентов

Изучаются инструменты анализа: Serpstat, Ahrefs, SimilarWeb. Разбираются способы анализа поведенческих факторов у конкурентов, изучение их ссылочного профиля и выводы для улучшения собственного сайта.

Описание практических занятий

1. Анализ целевой аудитории и определение целей сайта

Содержание: Студенты определяют целевую аудиторию веб-приложения, анализируют её интересы, поведение и потребности. Формулируют цели проекта: информационные, коммерческие, имиджевые.
Задание: Составить краткое описание целевой аудитории сайта. Определить основные цели сайта и оформить их в виде таблицы.

2. Разработка структуры сайта и создание навигации

Содержание: Разработка логичной структуры сайта на основе семантического ядра и целей. Создание схемы навигации.
Задание: Нарисовать структуру сайта (в виде карты или дерева). Описать логику перемещения между разделами.

3. Проверка уникальности текста и подбор контента

Содержание: Оценка контента на уникальность. Поиск релевантных текстов и изображений. Создание собственного текстового контента.
Задание: Провести проверку уникальности главной страницы. Написать/адаптировать уникальный абзац под тему сайта.

4. Настройка мета-тегов title и description

Содержание: Практика написания мета-тегов с учетом ключевых запросов.
Задание: Составить title и description для 3 страниц сайта. Обосновать выбор ключевых слов.

5. Сбор и группировка семантического ядра

Содержание: Использование сервисов (Яндекс.Вордстат, Serpstat, Key Collector) для сбора ключевых слов. Группировка по страницам.
Задание: Сформировать таблицу семантического ядра. Разбить ключи по тематическим группам.

6. HTML-мета-теги и внутренняя перелинковка

Содержание: Создание разметки страниц с корректным использованием заголовков (h1–h4), мета-тегов и ссылок.
Задание: Разметить одну из страниц проекта: добавить заголовки, meta, alt-атрибуты, внутренние ссылки.

7. Подключение аналитики: Яндекс.Метрика, Google Analytics

Содержание: Подключение и настройка инструментов веб-аналитики. Добавление счётчиков, целей, просмотр отчётов.
Задание: Установить счётчик Яндекс.Метрики на сайт, создать цель «Переход в раздел Контакты».

8. SEO-анализ текста и каталогизация ядра

Содержание: Проверка текста на SEO-параметры: плотность ключевых слов, тошнота, вода. Распределение ключей по страницам.
Задание: Проанализировать текст 1 страницы с помощью онлайн-сервисов. Оформить таблицу распределения ключей.

9. PageSpeed Insights: технический аудит

Содержание: Анализ сайта с помощью Google PageSpeed Insights. Изучение причин медленной загрузки и поиск решений.
Задание: Провести анализ сайта. Составить таблицу рекомендаций по улучшению скорости.

10. Создание sitemap.xml и robots.txt

Содержание: Создание и настройка карты сайта и файла robots.txt. Понимание их структуры и назначения.
Задание: Создать sitemap.xml вручную или через генератор. Написать robots.txt с запретом на индексирование определённых страниц.

11. Настройка и оформление страницы 404

Содержание: Разработка функциональной и эстетически понятной страницы 404 с навигацией и ссылками.
Задание: Создать HTML-страницу 404 с фирменным стилем, текстом и ссылками на главную/меню.

12. Работа в Яндекс.Вебмастере и Google Search Console

Содержание: Подключение сайта, отправка sitemap, проверка индексации и ошибок.
Задание: Подключить проект в Яндекс.Вебмастер. Отправить sitemap. Сфотографировать интерфейс.

13. Углублённая техническая оптимизация

Содержание: Оптимизация кода, минификация JS и CSS, настройка кэширования, lazy-loading изображений.
Задание: Внедрить lazy-loading на странице, провести минификацию CSS.

14. Поведенческие факторы: анализ и улучшение

Содержание: Анализ поведенческих метрик: глубина просмотра, отказ, конверсия. Улучшение CTA, интерфейса, логики переходов.
Задание: Провести аудит одной страницы. Предложить 3 улучшения интерфейса для удержания пользователя.

15. Влияние домена и хостинга. Работа с фильтрами

Содержание: Анализ параметров хостинга, влияния TLD, возраста домена. Изучение фильтров (АГС, Баден-Баден).
Задание: Подобрать и обосновать доменное имя и хостинг для проекта. Проверить сайт на фильтры.

16. Метрики: DAU, удержание, длина сессии, конверсии

Содержание: Работа с метриками поведения. Построение отчетов, интерпретация данных, настройка воронок.
Задание: На основе Google Analytics собрать отчет по DAU, длине сессии. Сделать выводы.

17. Мобильная адаптивность и юзабилити

Содержание: Проверка адаптивности сайта на разных устройствах. Оценка юзабилити по чек-листу.
Задание: Провести проверку с помощью Google Mobile Friendly. Сделать скриншоты ошибок и предложить правки.

18. Редиректы и канонические ссылки

Содержание: Настройка редиректов (301, 302), использование rel="canonical", борьба с дублями.
Задание: Настроить 301-редирект с несуществующей страницы на главную. Добавить canonical на одну страницу.

19. Микроразметка и SEO для динамических страниц

Содержание: Применение schema.org для описания сущностей (товары, статьи). Индексация SPA и AJAX.
Задание: Добавить микроразметку к карточке товара или статье. Провести тест на валидаторе.

20. SMM и стратегия контентного продвижения

Содержание: Создание стратегии контентного продвижения, определение каналов распространения, SMM-интеграция.
Задание: Разработать 5 публикаций для соцсетей, интегрированных с темой сайта. Оформить контент-план.

21. Сбор статистики и подготовка финального SEO-отчёта

Содержание: Подведение итогов: анализ посещаемости, видимости в поиске, достижение целей. Подготовка отчёта.
Задание: Составить финальный SEO-отчёт: трафик, позиции, проведённые мероприятия, рекомендации.

Примерные вопросы и практические задания к экзамену

Теоретическая часть

1. Что такое SEO-оптимизация и каковы её цели?
2. Определение и характеристика целевой аудитории сайта.
3. Что такое семантическое ядро и этапы его формирования?
4. Какие бывают типы поисковых запросов? Примеры.
5. Назначение и правила заполнения мета-тегов: Title, Description, Keywords.
6. Как осуществляется внутренняя перелинковка страниц сайта?
7. Что представляет собой файл `robots.txt` и как он работает?
8. Назначение и структура файла `sitemap.xml`.
9. Что такое микроразметка и зачем она нужна в SEO?
10. Как проводится технический аудит сайта (на примере PageSpeed Insights)?
11. Основные причины снижения производительности сайта.
12. Как влияет скорость загрузки сайта на поведенческие факторы?
13. Назовите ключевые инструменты аналитики сайта и их возможности.
14. Что такое поведенческие факторы и как они влияют на ранжирование?
15. Назовите и кратко охарактеризуйте основные метрики веб-приложений (DAU, Retention, CTR и др.).
16. Как влияет хостинг и доменное имя на продвижение сайта?
17. Основные принципы юзабилити сайта.
18. Чем отличаются внешняя и внутренняя SEO-оптимизация?
19. Какие существуют фильтры поисковых систем и за что они применяются?
20. Какие ошибки чаще всего встречаются при оптимизации и как их избежать?

Практические задания

1. **Составить семантическое ядро для сайта на выбранную тематику.**
   – Разделить запросы на группы.
   – Указать частотность и тип запроса.
2. **Проанализировать сайт с помощью PageSpeed Insights**
   – Определить основные технические ошибки.
   – Предложить пути устранения.

3. **Настроить мета-теги для главной страницы сайта.**
   – Title, Description, Keywords.
   – Обосновать выбор текста и ключевых слов.
4. **Создать структуру `sitemap.xml` и `robots.txt` для веб-проекта.**
   – Привести пример правильной структуры.
   – Объяснить, какие страницы индексируются, а какие — нет.
5. **Провести SEO-анализ текста:**
   – Проверка на уникальность, плотность ключевых слов, наличие «воды».
   – Внести правки и оптимизировать фрагмент.
6. **Разработать и описать структуру внутренней перелинковки для сайта.**
   – Использовать анкорные ссылки.
   – Составить схему связей.
7. **Настроить отслеживание посещаемости сайта.**
   – Подключить Яндекс.Метрику или Google Analytics (по скриншотам или шаблону).
   – Пояснить, какие показатели будут использоваться.
8. **Разметить текст с помощью микроразметки schema.org.**
   – Пример разметки организации, события или товара.
9. **Провести юзабилити-аудит сайта.**
   – Выявить проблемные зоны.
   – Дать рекомендации по улучшению навигации и визуального восприятия.
10. **Составить и оформить страницу 404.**
   – Указать, какие элементы должна содержать (поиск, ссылки, дизайн и т.д.).
   – Обосновать её важность для UX и SEO.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по курсовому проектированию
МДК 09.02 ОПИТИМАЗЦИЯ ВЕБ-ПРИЛОЖЕНИЙ
для специальности
09.02.07 «Информационные системы и программирование»

Екатеринбург
2025

Пояснительная записка

Цель курсового проекта

Разработка, реализация и оценка комплекса мероприятий по оптимизации веб-приложения с целью повышения его эффективности, видимости в поисковых системах и удобства использования.

Основные задачи курсового проекта

- Провести анализ веб-приложения и его конкурентов.
- Определить целевую аудиторию и сформировать семантическое ядро.
- Реализовать внутреннюю SEO-оптимизацию сайта.
- Настроить технические параметры: карта сайта, robots.txt, сниппеты и т.д.
- Подключить аналитические инструменты и провести аудит производительности.
- Проанализировать результаты и подготовить рекомендации по дальнейшему развитию.

Структура пояснительной записки

Введение

Краткое описание проекта, его тематики и значимости. Укажите, почему вы выбрали данную тему, в чем её актуальность.

Актуальность темы

Объясните, почему оптимизация веб-приложений важна в современных условиях (например, влияние скорости загрузки на SEO, конкурентоспособность, пользовательский опыт и т.п.).

Цель и задачи проекта

Формулируется одна общая цель и несколько конкретных задач, вытекающих из неё.

Обоснование выбора методов оптимизации

Описываются методы SEO, UX, технической и поведенческой оптимизации, которые планируется использовать. Объясняется, почему были выбраны именно они.

Аналитическая часть

1. Анализ конкурентов и определение целевой аудитории

1.1. Проведение сравнительного анализа 3–5 сайтов-конкурентов.
1.2. Составление портрета целевой аудитории, возможные сценарии взаимодействия с сайтом.
1.3. Постановка задач по результатам анализа.

2. Проектирование структуры и компонентов программного продукта

2.1. Краткое описание проекта: цель, функции, тип сайта.
2.2. Разработка структуры сайта: карта страниц, логика навигации.
2.3. Сбор ссылок на страницы веб-приложения, составление маршрутов переходов.
2.4. Разработка текстового контента с SEO-учётом и проверкой уникальности.
2.5. Создание макетов в Figma или другом инструменте, визуализация элементов интерфейса.

3. Сбор и обработка поисковой информации

3.1. Поиск ключевых слов и формулировка пользовательских запросов.
3.2. Формирование и кластеризация семантического ядра (разбивка по страницам, приоритетность, частотность и пр.).

Проектная часть

4. SEO-оптимизация

4.1. Заполнение и настройка Title, Description, Keywords.
4.2. Внутренняя перелинковка: анкорные ссылки, якоря.
4.3. Создание страницы 404, настройка редиректа через .htaccess.
4.4. Настройка и содержание файла Robots.txt.
4.5. Создание sitemap.xml и его отправка в вебмастеры.
4.6. Разработка сниппета с использованием OpenGraph или Schema.org.
4.7. Подключение Яндекс.Вебмастера или Google Search Console.
4.8. Подключение Яндекс.Метрики или Google Analytics. Настройка целей.

5. Тестирование и анализ производительности

5.1. Использование сервисов PageSpeed Insights, GTmetrix, Web.dev для анализа.
5.2. Оптимизация изображений, CSS и JS.
5.3. Проверка адаптивности на мобильных устройствах.
5.4. Кроссбраузерность: тестирование в нескольких браузерах.

6. Результаты оптимизации и рекомендации

6.1. Сравнительный анализ «до» и «после» оптимизации: позиции, скорость, трафик.
6.2. Выработка рекомендаций по поддержанию SEO и развитию проекта.
6.3. Итоги: достигнутые цели, решение поставленных задач.

Заключение

Краткое обобщение всей работы. Какие навыки были приобретены, какие инструменты использованы, какие проблемы решены.

Список использованных источников

Оформляется по ГОСТ. Включает книги, статьи, официальную документацию, онлайн-сервисы и пр.

Приложения

- Скриншоты интерфейсов до/после оптимизации
- Пример файла robots.txt
- Семантическое ядро (в формате таблицы)
- Сниппеты, элементы микроразметки
- Макеты страниц (в виде изображений)

Инструменты, которые рекомендуется использовать:

- **SEO-аудит:** PageSpeed Insights, GTmetrix, Lighthouse
- **Семантика:** Яндекс.Вордстат, Serpstat, Keyword Planner
- **Аналитика:** Яндекс.Метрика, Google Analytics
- **Вебмастеры:** Google Search Console, Яндекс.Вебмастер
- **Разметка:** OpenGraph, Schema.org
- **Макеты:** Figma, Adobe XD
- **HTML/SEO-проверка:** seolib.ru, sitechecker.pro, pr-cy.ru

Требования к оформлению

- Объём пояснительной записки: от 25 до 40 страниц
- Шрифт: Times New Roman, 14 пт
- Межстрочный интервал: 1,5
- Поля: стандартные (верхнее и нижнее — 2 см, левое — 3 см, правое — 1,5 см)
- Нумерация страниц — сквозная, с 3-й страницы

Критерии оценивания курсового проекта

1. Введение (10 баллов)

- **Актуальность темы (2 балла):**
  - Четко объяснено, почему тема проекта актуальна.
  - Указаны практическая ценность и значимость работы.
- **Цель и задачи проекта (4 балла):**
  - Цель сформулирована конкретно и логично.
  - Задачи соответствуют цели и содержанию работы.
- **Обоснование методов оптимизации (4 балла):**
  - Аргументировано выбраны методы оптимизации.
  - Показана связь методов с целями и задачами.

2. Анализ конкурентов и определение целевой аудитории (20 баллов)

- **Исследование конкурентов (8 баллов):**
  - Проведен качественный анализ конкурентов.
  - Выявлены сильные и слабые стороны.
- **Определение целевой аудитории (8 баллов):**
  - Портрет пользователя описан подробно.
  - Приведены сценарии использования.
- **Постановка задач на основе анализа (4 балла):**
  - Задачи логично вытекают из анализа.

3. Проектирование структуры и компонентов программного продукта (25 баллов)

- **Общая информация о проекте (5 баллов):**
  - Представлен краткий и понятный обзор проекта.
- **Разработка структуры сайта и навигации (8 баллов):**
  - Структура и навигация логичны и удобны для пользователя.
  - Учтена работа с разными устройствами.
- **Оформление текстового контента (4 балла):**
  - Текстовый контент уникален, грамотно оформлен.
- **Разработка макетов (8 баллов):**
  - Макеты соответствуют целям проекта.
  - Дизайн проработан с учетом юзабилити и визуальной эстетики.

4. Настройка SEO-оптимизации (20 баллов)

- **Подбор ключевых запросов и семантического ядра (8 баллов):**
  - Анализ поискового спроса выполнен корректно.
  - Семантическое ядро составлено полно и логично.

- **Заполнение мета-тегов (5 баллов):**
  - Title, Description и Keywords соответствуют требованиям SEO.
- **Настройка ссылок и внутренних якорей (4 балла):**
  - Логичная структура ссылок, корректное использование анкорных текстов.
- **Подключение инструментов аналитики (3 балла):**
  - Осуществлено подключение Google Analytics или Яндекс.Метрики.
  - Описана их необходимость.

5. Тестирование и анализ производительности (20 баллов)

- **Анализ производительности (5 баллов):**
  - Использованы современные инструменты (PageSpeed Insights, GTmetrix).
- **Оптимизация изображений и кода (5 баллов):**
  - Проведена оптимизация изображений, CSS и JS.
- **Проверка адаптивности и кроссбраузерности (6 баллов):**
  - Тестирование выполнено на различных устройствах и браузерах.
- **Выявление и устранение проблем (4 балла):**
  - Найдены и исправлены основные ошибки.

6. Результаты оптимизации и рекомендации (15 баллов)

- **Анализ улучшений (5 баллов):**
  - Четко показаны улучшения (до и после).
  - Использованы метрики для подтверждения результатов.
- **Рекомендации по поддержанию проекта (5 баллов):**
  - Рекомендации логичны и полезны для дальнейшего развития.
- **Выводы (5 баллов):**
  - Подведены итоги работы, показана связь результатов с целями проекта.

7. Заключение (5 баллов)

- **Обобщение работы (3 балла):**
  - Четко описаны результаты проекта.
- **Описание полученных навыков (2 балла):**
  - Показано, чему автор научился в процессе выполнения работы.

8. Оформление и структура проекта (10 баллов)

- **Структура и последовательность изложения (4 балла):**
  - Материал изложен логично и структурировано.
- **Оформление пояснительной записки (4 балла):**
  - Соответствие требованиям к оформлению.
- **Список использованных источников и приложений (2 балла):**
  - Полный и корректно оформленный список литературы и приложений.

Итоговая оценка: 100 баллов

Дополнительные замечания:

- Наличие орфографических и грамматических ошибок может снижать общую оценку.
- Недостаток данных или отсутствие разделов также будет отражаться на итоговой оценке.

Ранжирование оценок по курсовому проекту (100 баллов)

Баллы	Оценка	Уровень выполнения	Комментарий
90 – 100	**Отлично (5)**	Полный уровень	Проект выполнен полностью, разделы проработаны глубоко, задания выполнены качественно. Присутствует аналитика, выводы, оформление соответствует нормам.
75 – 89	**Хорошо (4)**	Продвинутый уровень	Большинство разделов проработано, но имеются незначительные недочеты (неполные выводы, слабое оформление или поверхностный анализ). Нет критических ошибок.
60 – 74	**Удовлетворительно (3)**	Базовый уровень	Проект содержит слабые или не до конца раскрытые разделы. Анализ и обоснования поверхностные. Нарушения в оформлении или логике изложения.
менее 60	**Неудовлетворительно (2)**	Недостаточный уровень	Работа не соответствует требованиям. Пропущены ключевые разделы, отсутствует логика, серьёзные нарушения в структуре и содержании. Требуется доработка.

Дополнительные замечания к оценке:

- Отсутствие хотя бы одного ключевого раздела (например, семантического ядра, мета-тегов или аналитики) может снизить итоговую оценку на один балл.
- Наличие грамматических и орфографических ошибок, плохая структура текста или несоблюдение требований к оформлению — может снизить оценку на 5–10 баллов, особенно при общем балле на границе уровня.
- За оригинальные решения, применение дополнительных инструментов или особенно качественную реализацию можно добавить до 5 бонусных баллов (по усмотрению преподавателя).

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ
ГАПОУ СО «ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

КУРСОВОЙ ПРОЕКТ

по МДК 09.02 Оптимизация веб-приложений

ТЕМА: «формулировка темы берется из приказа»

0	9	0	2	0	7	- КП -	3	5	1	8	–	И	П	2	1	2к	–	2	2

ОЦЕНКА: _____

Руководитель курсового проекта:
Софьина Н.А. _____

Студент группа ИП-212к
Петров И. _____

2025

Приложение Лист задания

МИНИСТЕРСТВО ОБРАЗОВАНИЯ СВЕРДЛОВСКОЙ ОБЛАСТИ

ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СВЕРДЛОВСКОЙ ОБЛАСТИ
«ЕКАТЕРИНБУРГСКИЙ МОНТАЖНЫЙ КОЛЛЕДЖ»

ЗАДАНИЕ №_____

Для курсового проектирования студента группы ИП-212к специальности

09.02.07 «Информационные системы и программирование»

Фамилия, имя и отчество Трошков Дмитрий Александрович

Время курсового проектирования с 06.10.2025 по 12.12.2025

1. Тема курсового проекта Монтаж и настройка системы видеонаблюдения в кабинетах вычислительного центра

Курсовой проект на указанную тему выполняется студентом колледжа в следующем объёме:

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Введение 1. Анализ конкурентов и определение целевой аудитории 1.1. Исследование конкурентов: анализ сильных и слабых сторон 1.2. Определение целевой аудитории: портрет пользователя и сценарии использования 1.3. Постановка задач на основе анализа 2. Проектирование структуры и компонентов программного продукта 2.1. Общая информация о проекте 2.2. Подбор ключевых запросов: анализ поискового спроса 2.3. Формирование семантического ядра 2.4. Разработка структуры сайта и навигации 2.5. Сбор ссылок на страницы веб-приложения 2.6. Оформление и проверка уникальности текстового контента 2.7. Разработка макетов: разметка страниц и дизайн веб-приложения 3. Настройка SEO-оптимизации проекта 3.1. Заполнение мета-тегов: Title, Description и Keywords 3.2. Настройка анкорных ссылок и внутренних якорей 3.3. Создание страницы 404 3.4. Подключение инструмента "Вебмастер" и обоснование его необходимости 3.5. Подключение аналитического инструмента: настройка Google Analytics / Яндекс.Метрика 4. Тестирование и анализ производительности 4.1. Инструменты для анализа производительности сайта 4.2. Проверка скорости загрузки страниц (PageSpeed Insights, GTmetrix) 4.3. Оптимизация изображений и кода (CSS, JS) 4.4. Проверка на адаптивность и кроссбраузерность 5. Результаты оптимизации и рекомендации 5.1. Анализ улучшений после внедрения оптимизаций 5.2. Рекомендации по поддержанию и дальнейшему развитию проекта 5.3. Выводы о достигнутых результатах Заключение

2. РАСЧЕТНАЯ ЧАСТЬ ПРОЕКТА

3. ГРАФИЧЕСКАЯ ЧАСТЬ ПРОЕКТА

Дата выдачи «06» октября 2025 г.

Срок окончания «_____» _____2025 г.

Зав.отделением Софьина Н.А. (_____)

Преподаватель Софьина Н.А. (_____)