



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И.
Носова»



УТВЕРЖДАЮ
Директор ИЭиАС
В.Р. Храмшин

03.02.2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

РАЗРАБОТКА ВЕБ-ПРИЛОЖЕНИЙ

Направление подготовки (специальность)
09.03.03 Прикладная информатика

Направленность (профиль/специализация) программы
Управление проектами разработки бизнес-приложений для цифровой экономики

Уровень высшего образования - бакалавриат

Форма обучения
очная

Институт/ факультет	Институт энергетики и автоматизированных систем
Кафедра	Бизнес-информатики и информационных технологий
Курс	1, 2
Семестр	2, 3

Магнитогорск
2026 год

Рабочая программа составлена на основе ФГОС ВО - бакалавриат по направлению подготовки 09.03.03 Прикладная информатика (приказ Минобрнауки России от 19.09.2017 г. № 922)

Рабочая программа рассмотрена и одобрена на заседании кафедры Бизнес-информатики и информационных технологий

22.01.2026, протокол № 5

Зав. кафедрой

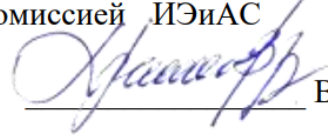


Г.Н. Чусавитина

Рабочая программа одобрена методической комиссией ИЭиАС

03.02.2026 г. протокол № 5

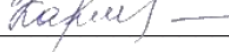
Председатель



В.Р. Храмшин

Рабочая программа составлена:

доцент кафедры БИиИТ, канд. пед. наук

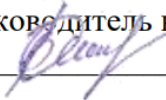


Е.В. Карманова

Рецензент:

руководитель направления консалтинга, КОНСОМ ГРУПП, канд. тех.

наук



В.А. Ошурков

Лист актуализации рабочей программы

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2027 - 2028 учебном году на заседании кафедры Бизнес-информатики и информационных

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____ Г.Н. Чусавитина

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2028 - 2029 учебном году на заседании кафедры Бизнес-информатики и информационных

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____ Г.Н. Чусавитина

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2029 - 2030 учебном году на заседании кафедры Бизнес-информатики и информационных

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____ Г.Н. Чусавитина

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2030 - 2031 учебном году на заседании кафедры Бизнес-информатики и информационных

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____ Г.Н. Чусавитина

1 Цели освоения дисциплины (модуля)

Целями освоения дисциплины «Разработка веб-приложений» являются: формирование профессиональных компетенций в области веб-разработки - освоение основных языков веб-разработки, изучение фреймворков и библиотек для реализации интернет ресурсов, приобретение навыков работы с системами контроля версий, а также развитие практических навыков создания адаптивных интерфейсов, интеграция с API и сторонними сервисами.

2 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина Разработка веб-приложений входит в обязательную часть учебного плана образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Информатика

Программирование

Информационные системы и технологии

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Базы данных

Технологии искусственного интеллекта

Большие данные в цифровой экономике

Основы управления качеством и рисками в ИТ-проектах

Оценка эффективности ИТ-проектов

Выполнение и защита выпускной квалификационной работы

Подготовка к сдаче и сдача государственного экзамена

3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «Разработка веб-приложений» обучающийся должен обладать следующими компетенциями:

Код индикатора	Индикатор достижения компетенции
ОПК-7	Способен разрабатывать алгоритмы и программы, пригодные для практического применения;
ОПК-7.1	Определяет средства разработки программных средств для решения практических задач профессиональной деятельности
ОПК-7.2	Разрабатывает алгоритмы и программы для решения прикладных задач различных классов

4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 6 зачетных единиц 216 академических часов, в том числе:

- контактная работа – 138,7 академических часов;
- аудиторная – 136 академических часов;
- внеаудиторная – 2,7 академических часов;
- самостоятельная работа – 77,3 академических часов;
- в форме практической подготовки – 0 академических часов;

Форма аттестации - зачет, зачет с оценкой

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в академических часах)			Самостоятельная работа студента	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код компетенции
		Лек.	лаб. зан.	практ. зан.				
1. Разработка Интернет приложений на клиентской стороне								
1.1 Основы Web-технологий	2	2	2			Подготовка к лекции и выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.1
1.2 Архитектура web-приложений		2	2			Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.1
1.3 Основы семантического веба. HTML5		2	2			Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2
1.4 Каскадные таблицы стилей. Подходы к верстке.		3	4			Выполнение лабораторной работы	Отчет по выполнению лабораторной работы	ОПК-7.2
1.5 Программный интерфейс для доступа и манипулирования содержимым веб-страниц DOM API		2	4			Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2
1.6 Фреймворки и библиотеки JavaScript. jQuery. Vue. Angular. React.		7	6			Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2
1.7 Языки реализации клиентских сценариев. JavaScript.		6	4		2,3	Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2
Итого по разделу		24	24		2,3			
2. Основы веб-дизайна								
2.1 Юзабилити веб-приложения. UX/UI	2	4	4		1	Выполнение лабораторной	Отчет по лабораторной	ОПК-7.2, ОПК-7.1

						работы	работе	
2.2 Дизайн в цифровой среде.	2	4	4		3			ОПК-7.2
Итого по разделу		8	8		4			
Итого за семестр		32	32		6,3		зачёт	
3. Разработка Интернет приложений на серверной стороне								
3.1 Политики информационной безопасности при разработке Интернет приложений	3	2	4		4	Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2, ОПК-7.1
3.2 Python – язык разработки веб приложений		2	8		14	Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2
3.3 Фреймворки Python		4	18		20	Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2
3.4 Разработка интернет-приложения с использованием Python фреймворка		6	10		14	Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.2
3.5 Особенности разработки RestFull API		4	14		19	Выполнение лабораторной работы	Отчет по лабораторной работе	ОПК-7.1, ОПК-7.2
Итого по разделу		18	54		71			
Итого за семестр		18	54		71		зао	
Итого по дисциплине		50	86		77,3		зачет, зачет с оценкой	

5 Образовательные технологии

Основными образовательными технологиями, положенными в основу преподавания дисциплины являются:

- активные технологии обучения:

о метод ролевых игр - это разыгрывание участниками группы сценки с заранее распределенными ролями в интересах овладения определенной поведенческой или эмоциональной стороной жизненных ситуаций.

Ролевая игра проводится в небольших группах (3-5 участников);

о технологии кейс-стади - техника обучения, использующая описание ре-альных ситуаций. Обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Кейсы базируются на реальном фактическом материале, или же приближены к реальной ситуации;

о разработка проекта - это способ достижения дидактической цели через детальную разработку проблемы (технология), которая должна завершиться вполне реальным, осязаемым практическим результатом, оформленным тем или иным образом;

о работа в малых группах - это одна из самых популярных стратегий, так как она дает всем обучающимся (в том числе и стеснительным) возможность участвовать в работе, практиковать навыки сотрудничества, меж-личностного общения (в частности, умение активно слушать, вырабатывать общее мнение, разрешать возникающие разногласия);

- интерактивные лекции:

о лекций-дискуссий - преподаватель приводит отдельные примеры в виде ситуаций или кратко сформулированных проблем и предлагает студентам коротко обсудить, затем краткий анализ, выводы и лекция продолжается. Положительным в дискуссии является, то, что обучаемые согласятся с точкой зрения преподавателя с большой охотой, скорее в ходе дискуссии, нежели во время беседы, когда преподаватель лишь указывает на необходимость принять его позицию по обсуждаемому вопросу. Данный метод позволяет преподавателю видеть, насколько эффективно слушатели используют полученные знания в ходе дискуссии.

Активные технологии обучения преимущественно используются в рамках практических занятий, интерактивные лекции - в процессе изучения и закрепления нового учебного материала.

В качестве практико-ориентированного средства обучения выбран образовательный портал ФГБОУ ВО «МГТУ им. Г.И. Носова».

6 Учебно-методическое обеспечение самостоятельной работы обучающихся

Представлено в приложении 1.

7 Оценочные средства для проведения промежуточной аттестации

Представлены в приложении 2.

8 Учебно-методическое и информационное обеспечение дисциплины

а) Основная литература:

1. Янцев, В. В. Web-программирование на Python : учебное пособие для вузов / В.В. Янцев. — 3-е изд., перераб. — Санкт-Петербург : Лань, 2024. — 180 с. — ISBN 978-5-507-48364-8. — Текст : электронный // Лань : электронно-библиотечная система. —URL: <https://e.lanbook.com/book/392993> (дата обращения: 17.01.2026). — Режим доступа: для авториз. пользователей.

2. Калиберда, Е. А. Разработка web-приложений : учебное пособие / Е. А. Калиберда, К. В. Кравченко. — Омск : ОмГТУ, 2023. — 100 с. — ISBN 978-5-8149-3679-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/421766> (дата обращения: 17.01.2026). — Режим доступа: для авториз. пользователей.

Дополнительная литература:

Янцев, В. В. JavaScript. Создание анимации и разработка игр : учебное пособие для вузов / В. В. Янцев. — Санкт-Петербург : Лань, 2025. — 116 с. — ISBN 978-5-507-50164-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/438941> (дата обращения: 11.01.2026). — Режим доступа: для авториз. пользователей.

Web-программирование на языке Python : учебно-методическое пособие / составители Т. Г. Бурдыко, А. В. Гавриленко. — Сургут : СурГУ, 2025. — 56 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/494693> (дата обращения: 11.01.2026). — Режим доступа: для авториз. пользователей.

Журнал «Программные продукты и системы»[Электронный ресурс]. Научно-исследовательский институт «Центрпрограммсистем» — Режим доступа: https://e.lanbook.com/journal/2276#journal_name — Загл. с экрана.

в) Методические указания:

Карманова, Е. В. Основы веб-дизайна : Учебно-методическое пособие / Е. В. Карманова, О. В. Безбородова ; . — Магнитогорск : Магнитогорский государственный технический университет им. Г.И. Носова, 2022. — 112 с. — ISBN 978-5-9967-2522-9. — EDN VVXQRP.

г) Программное обеспечение и Интернет-ресурсы:

Программное обеспечение

Наименование ПО	№ договора	Срок действия лицензии
LibreOffice	свободно распространяемое ПО	бессрочно
Браузер Yandex	свободно распространяемое ПО	бессрочно
PostgreSQL	свободно распространяемое ПО	бессрочно
Anaconda Python	свободно распространяемое ПО	бессрочно
MS Visual Studio Code	свободно распространяемое ПО	бессрочно

Профессиональные базы данных и информационные справочные системы

Название курса	Ссылка
Национальная информационно-аналитическая система – Российский индекс научного цитирования (РИНЦ)	URL: https://elibrary.ru/project_risc.asp
Электронные ресурсы библиотеки МГТУ им. Г.И. Носова	https://host.megaprolib.net/MP0109/Web

9 Материально-техническое обеспечение дисциплины (модуля)

Материально-техническое обеспечение дисциплины включает:

Учебные аудитории для проведения занятий лекционного типа Персональный компьютер (или ноутбук) с пакетом Office, с выходом в Интернет и с доступом в электронную информационно-образовательную среду университета. Доска, мультимедийный проектор, экран. Мультимедийные презентации к лекциям, учебно-наглядные пособия

Учебные аудитории для проведения лабораторных занятий Персональные компьютеры с пакетом Office; с выходом в Интернет и с доступом в электронную информационно-образовательную среду университета. Браузер Yandex.

Аудитории для самостоятельной работы: компьютерные классы; читальные залы библиотеки Персональные компьютеры с пакетом Office; с выходом в Интернет и с доступом в электронную информационно-образовательную среду университета. Браузер Yandex.

Аудитории для групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации Персональные компьютеры с пакетом Office; с выходом в Интернет и с доступом в электронную информационно-образовательную среду университета. Браузер Yandex.

Аудитория для хранения и профилактического обслуживания учебного оборудования Мебель для хранения и обслуживания оборудования (шкафы, столы), учебно-методические материалы, компьютеры, ноутбуки, принтеры.

Учебно-методическое обеспечение самостоятельной работы обучающихся

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
<p>Понятие Интернет приложения. Архитектура Интернет-приложений. Классификация. Подходы к разработке Интернет приложения. Среды программирования, IDE</p>	<p>Подготовка к лекции и семинару. Перечень тем для подготовки к семинарским занятиям:</p> <ol style="list-style-type: none"> 1. Определение Интернет приложения. 2. Классификация Интернет приложений. 3. Основные понятия, используемые при реализации Интернет приложений. 4. Архитектурные шаблоны Интернет приложений. 5. Механизмы взаимодействия web-сервера и клиента.
<p>Семантическая верстка</p>	<p>Подготовка к лекции и лабораторной работе. Лабораторная работа. Обработка формы</p> <ol style="list-style-type: none"> 1. Создать 3 html страницы "Моя визитная карточка". Все файлы разместить в папке Mysite_ВашаФамилия. 2. Первый файл назвать index.html. Это главная страница. Разместить в нем следующую информацию: ФИО, направление подготовки, профиль, дата рождения, фотография, название школы / колледжа, город рождения, телефон, адрес почтового ящика, ссылки на профили соц. сетей. Ссылки на страницы Главная, Увлечения, Контакты. Требования к использованию html тегов: <ul style="list-style-type: none"> • Метатеги с описанием и ключевыми словами. • Заголовки 1-3 уровня. • Изображения. • Ссылки. 3. Второй файл назвать Interests.html. Разместить в нем следующую информацию: ФИО. Заголовок "Мои увлечения". Описать свои достижения, хобби, привычки, интересы и т.д. Моя любимая музыка, фильм - вставить видео, аудио. Ссылки на страницы Главная, Увлечения, Контакты. Ссылки на страницы Главная, Увлечения, Контакты. Требования к использованию html тегов: <ul style="list-style-type: none"> • Метатеги с описанием и ключевыми словами. • Заголовки 1-2 уровня. • Изображения. • Ссылки. • Списки. <p>Таблицу с объединением строки и столбцов (обязательно отработать оба способа объединения). Текст, оформленный курсивом. Видео, аудио</p> <ol style="list-style-type: none"> 4. Третий файл назвать Contacts.html. Разместить в нем следующую информацию: ФИО. Заголовок "Написать мне". Создать форму, содержащую следующие элементы: поле Имя отправителя, название почтового ящика отправителя, номер телефона (указать шаблон ввода), переключатели (Как обращаться к пользователю в дальнейшем: Уважаемый / Уважаемая), выпадающий список с тематикой письма (Жалоба, Предложения, Дружба, Совместный проект и т.д.), текстовое поле для ввода сообщения, поле для прикрепления файла, флажок для согласия на обработку персональных данных, кнопка Отправить письмо. Ссылки на страницы Главная, Увлечения, Контакты. Требования к использованию html тегов: <ul style="list-style-type: none"> • Метатеги с описанием и ключевыми словами. • Заголовки 1-2 уровня. • Ссылки. • Поля формы.
<p>Валидный CSS.</p>	<p>Подготовка к лекции и лабораторной работе. Лабораторная работа. Flexbox</p>

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
	<ol style="list-style-type: none"> 1. Реализовать для созданных в предыдущем задании страницах все способы <u>объявления</u> CSS (3 способа). Штраф за отсутствие одного из способов - 2 балла. 2. Создать стили для таблицы, списков, меню сайта, элементов формы. Штраф за отсутствие одного из стилей - 3 балла. 3. Реализовать разметку сайта на основе flexbox. Обязательные разделы: шапка, меню, контент, подвал (футтер). Все страницы сайта должны быть оформлены в одном стиле, все разделы должны присутствовать. Штраф за отсутствие одного из разделов - 3 балла. Штраф за различие стилей на страницах - 3 балла. 4. Реализовать не менее 2 блоков с флекс-контейнерами (на любой странице), внутри которых все флекс-элементы размещены по горизонтали. Применить свойства выравнивания и переноса для флекс-элементов. Штраф - 5 баллов. 5. Добавить на каждую страницу фавикон. Штраф - 3 балла. 6. Реализовать объявление следующих типов селекторов: класс, идентификатор, тег, атрибутов; потомков (вложенные селекторы). Штраф за отдельный тип селектора - 2 балла. 7. На второй странице добавить кнопку Наверх. Кнопка должна быть зафиксирована в нижней части, справа. При нажатии на кнопку страница пролистывается вверх. Штраф за отсутствие кнопки - 5 баллов. 8. Реализовать анимацию на главной странице сайта, используя свойство animation. Штраф за отсутствие - 5 баллов. 9. Проверить файл CSS через валидатор CSS. Штраф за каждый тип ошибки - 2 балла. <p>Лабораторная работа. CSS. Grid Сверстать в соответствии с предложенным мокапом лендинг-страницу сайта. Верстку каркаса страницы реализовать с использованием сеток (grid). Внутри разрешается использовать флекс-боксы. Верстка должна быть "резиновой". Все блоки обязательно должны быть сверстаны. Необходимо полностью повторить концепцию мокапа: все разделы, блоки, ориентацию блоков, размещение блоков на странице (отступы, выравнивание). Разрешается использовать другие изображения, иную палитру цветов, шрифты. Если на изображении размещен текст, то текст верстается отдельным блоком. Варианты с мокапами можно скачать по ссылке - https://drive.google.com/file/d/1gk_WYIIIRaaYJjQ6lRGlps7V90T96btJk/view?usp=share_link</p> <p>Лабораторная работа. Фреймворки CSS. Bootstrap Создайте сайт, используя следующие элементы Bootstrap 5:</p> <ul style="list-style-type: none"> • Хлебные крошки/крошки • Кнопки • Карточки или Аккордеоны (разворачиваемые панели) • Формы (типы элементов - любые) • Списки • Навигационная панель • Пагинация (нумерация страниц) • <u>Слайдер</u>. Требования к <u>слайдеру</u>: располагается на главной странице сайта, обязательно имеет средства навигации (возможность ручного пролистывания слайдов), слайды сопровождаются дополнительными текстовыми блоками с информацией, ссылками или таблицами. <p>Тематика сайтов по вариантам.</p> <ol style="list-style-type: none"> 1. Сайт туристического агенства. 2. Сайт волонтерского движения (привлечение новых волонтеров, организация мероприятий, информирование).

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
	<p>3. Сайт/Интернет-магазин для продажи кофе.</p> <p>4. Сайт для отеля 5*.</p> <p>5. Сайт для строительной компании (строительство как для физических лиц (коттеджи, таун-хаусы и др), так и для юридических лиц (многоэтажные здания).</p> <p>6. Сайт /Интернет-магазин для продажи компьютерной техники.</p> <p>7. Сайт кредитного банка (кредиты, вклады).</p> <p>8. Сайт книжного магазина.</p> <p>9. Сайт/Интернет-магазин продажи домашних животных.</p> <p>10. Сайт для компании, занимающейся изготовлением продажей кровельных материалов (от металлочерепицы, профлиста, металлосайдинга до фасадных систем и сэндвич-панелей)</p> <p>Лабораторная работа. Адаптивность сайтов Создайте сайт по одной из нижеприведенных тематик:</p> <ol style="list-style-type: none"> 1. Человеко-машинный интерфейс 2. WEB-аналитика 3. Реклама и PR в Интернет 4. Юзабилити интернет-проектов 5. Разновидности поисковых систем в Интернете 6. Сетевые сервисы Web 2.0 и 3.0. 7. Облачные технологии 8. Чат-боты 9. UI и UX 10. Iot и INDUSTRY 4.0 11. Семантическая паутина (англ. SemanticWeb) <p>Требование к сайту:</p> <ol style="list-style-type: none"> 1. Наличие не менее 5 HTML страниц. (название HTML файлов ТОЛЬКО латинскими буквами, рекомендуется - index, home, pageN, news). Штраф за отсутствие страницы - 5 баллов. 2. Наличие главной страницы, с которой можно обратиться на любую страницу. Штраф - 2 балла. 3. Наличие файла css с внешним представлением сайта. Штраф - 2 балла. 4. Структура страниц сайта: шапка, меню, контент, футер. Структура на всех страницах одинаковая. Штраф - 5 баллов. 5. В блоке head - наличие фавикона, мета данных (ключевые слова, описание). Штраф за отсутствие отдельного элемента - 3 балла. 6. Реализация эффектов с помощью CSS (штраф за отсутствие отдельного эффекта - 3 балла): <ul style="list-style-type: none"> • Градиент • Тень • Скругленные уголки элементов (блоков/изображений) • Анимация (свойства: transition, animation) • Декоративные рамки (свойство border-image) • Полупрозрачный фон с картинкой • Текст в несколько колонок 7. Реализация адаптивности сайта для планшетов и смартфонов. Верстка должна быть адаптивна и корректно открываться со следующих устройств: <ul style="list-style-type: none"> • Мобильные телефоны с разрешением 375x812px (iPhone X); • Планшеты с разрешением 1366x1024px (iPad Pro); • Компьютеры с шириной экрана от 1600px. <p>Штраф за отсутствие реализации адаптивности под конкретный вид устройства - 5 баллов.</p> <p>Штраф за появление горизонтальной прокрутки - 10 баллов.</p> <p>Немного об адаптивности: https://go-job.ru/seo/tehnicheskaya-optimizaciya-sajta/adaptivnost.htm</p> <p>8. Импортирование и применение нестандартного шрифта. Штраф за отсутствие импортирования шрифтов - 3 балла.</p> <p>10. Все графические изображения должны лежать в отдельной папке Images.</p>

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
	<p>Изображения для сайта должны быть хорошего качества. Штраф за отсутствие папки - 3 балла.</p> <p>12. Применить все способы объявления css (3 способа). Штраф за каждый способ - 2 балла.</p>
<p>Программный интерфейс для доступа и манипулирования содержимым веб-страниц DOM API</p>	<p>Подготовка к лекции и лабораторной работе.</p> <p>Лабораторная работа. Динамическое построение html – документа</p> <p>Построить html – страницу, содержащую форму для визуального калькулятора, используя DOM</p>
<p>Языки реализации клиентских сценариев. JavaScript.</p>	<p>Подготовка к лекции и лабораторной работе.</p> <p>Лабораторная работа. Обработка формы</p> <p>Реализовать следующий функционал:</p> <ul style="list-style-type: none"> • Чтение данных с формы и вывод на страницу HTML без перезагрузки. • Каждое следующее сообщение должно добавляться ниже. • Реализовать стили для ленты сообщений (придумать самим) • После обновления страницы лента сообщений пустая • Обработать поля ввода для формы – если пользователь не ввел текст в отдельное поле, то выводится сообщение об ошибке. (Разрешается Alert) • Скрипт js разместить в отдельном файле. <p>Лабораторная работа. Необходимо разработать браузерную игру.</p> <p>Игровое поле представляет собой подводный мир с плавающими рыбками. Игрок должен набрать максимальное количество баллов, кликая на них. После клика по рыбке она исчезает, а игрок получает очки. Каждая рыбка движется со случайными скоростью и траекторией. Рыбка не может находиться на игровом поле постоянно, со временем она должна уплывать, если по ней так и не кликнули. Одновременно на одном игровом поле может находиться не более 10 рыбок. Существуют рыбки 3 размеров. При клике на маленькую начисляется 30 очков, на среднюю – 20 очков, на большую – 10 очков.</p> <p>Участнику необходимо разработать дизайн, реализовать указанный функционал игры, а также разработать понятный, удобный и оригинальный интерфейс игры (менять игровую логику запрещено).</p> <p>HTML5 и CSS3 код должны быть валидными. Создаваемые файлы должны быть структурированы и содержать комментарии. JS код должен выполняться без отображения ошибок, в т.ч. в консоли браузера. Текст программы должен быть стандартизирован и понятен любому стороннему программисту.</p> <p>Браузером для проверки основных функциональных возможностей является MozillaFirefoxDeveloperEdition. Однако работа приложения будет также проверена в браузере GoogleChrome для проверки кроссбраузерности программы.</p>
<p>Фреймворки и библиотеки JavaScript. jQuery. Vue. Angular. React.</p>	<p>Подготовка к лекции и лабораторной работе</p> <p>Лабораторная работа.</p> <p>Разработать клиентское приложение для API социальной сети.</p>
<p>GIT</p>	<p>Подготовка к лекции и лабораторной работе</p> <p>Лабораторная работа.</p> <p>Установить GIT на домашний ПК (ноутбук).</p> <p>Windows</p> <p>Скачайте и установите «Гит» на свой компьютер. Ссылка на скачивание с официального сайта Git.</p> <p>Когда запустите установщик, вам предложат использовать GitBash — это ещё одна программа для полноценной работы с «Гитом». Обязательно поставьте галочку для её установки. Также поставьте галочку на пункте: «Использовать Git из командной строки» (Use Git... from the Command Prompt).</p> <p>macOS</p> <p>1. Скачайте и установите «Гит» на свой компьютер. Ссылка на скачивание с официального сайта Git.</p> <p>Программа-установщик предложит варианты применения «Гита». Выберите</p>

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
	<p>«Использовать Git из командной строки» (Use Git ... from the Command Prompt). Linux</p> <p>У Linux существует множество версий. Чтобы установить «Гит», найдите подходящую версию в разделе установки Git на Linux.</p> <p>!!!Рекомендуем пройти курс по работе с GIT - https://github.com/gu (прохождение данного курса не оценивается, но он вам поможет полностью освоить принципы работы с GIT)</p> <ol style="list-style-type: none"> 2. Подключите GIT к проекту, который вы разработали ранее (проект д.б. связан с разработкой web-ресурса (в случае, если у вас не сохранились проекты, вы можете создать новый - пару файлов (html, css, js)). 3. Сделайте несколько коммитов. Чтобы сделать коммит, нужно добавить в него файлы — без файлов закоммитить не получится. Для тренировки в первом коммите отредактируйте разметку, а во втором добавьте стили. 4. (в случае, если еще нет учетки). Зарегистрируйтесь в Github (либо GitVerse - российский аналог). https://github.com/ 5. (для домашнего ПК.) Сгенерируйте SSH-ключ и привяжите его к вашему аккаунту GitHub. 6. Создайте новый репозиторий в Github и свяжите локальный и удаленный репозиторий с проектом. 7. Выполните один коммит к вашему проекту. Итого в github у проекта должно быть не менее 3х коммитов. 8. Опубликуйте проект с помощью GitHub Pages
<p>Политики информационной безопасности при разработке Интернет приложений</p>	<p>Подготовка к лекции и лабораторной работе Лабораторная работа. Реализовать валидацию данных для формы с регистрацией пользователя.</p>
<p>Python – язык разработки веб приложений</p>	<p>Подготовка к лекции и лабораторной работе Лабораторная работа. Описание задание: Создать новый проект на Django. Создать html-форму в соответствии со своим вариантом. Определить стили страницы через внешнюю страницу стилей. Добавить текстовый и графический контент на страницу под тематику вашего варианта. Реализовать функционал для сохранения пользовательских настроек с использованием cookies, таких как темы, язык интерфейса или последние посещенные страницы. Варианты:</p> <ol style="list-style-type: none"> 1. Веб-приложение для настройки персонализированных новостей. 2. Онлайн-кинотеатр с возможностью выбора жанра фильмов. 3. Веб-сайт для чтения комиксов с настройками внешнего вида. 4. Приложение для создания и хранения заметок с разными темами оформления. 5. Сайт для планирования событий с настройками уведомлений. 6. Онлайн-энциклопедия с возможностью выбора темного или светлого режима. 7. Приложение для изучения языков с сохранением прогресса. 8. Веб-приложение для спортивных фанатов с настройкой любимых команд. 9. Магазин одежды с возможностью выбора любимых стилей и размеров. 10. Платформа для блогов с настройками читательских предпочтений. 11. Веб-приложение для создания плейлистов 12. Онлайн-платформа для курсов и лекций 13. Сайт для планирования путешествий 14. Онлайн-галерея произведений искусств <p>Этапы выполнения задания:</p> <ul style="list-style-type: none"> • Инициализация локального репозитория. • Создайте файл .gitignore для Python (используйте стандартный шаблон). • Создание и активация виртуального окружения (venv) • Установка фреймворка и зависимостей для проекта

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
	<ul style="list-style-type: none"> • Зафиксируйте зависимости: requirements.txt • Убедитесь, что папка с виртуальным окружением добавлена в .gitignore. • Создание нового Django-проекта и приложение • Первый коммит (дальше не забываем делать коммиты) • Данные: создайте данные прямо в коде, используя списки и словари. • Реализация представлений • Реализация формы • Реализация шаблонов (Templates) • static/ - стили, изображения • URLs (Маршрутизация): • Тестирование • Создайте README.md с инструкцией по установке и запуску. • Финальный коммит и публикация • Залейте проект на GitHub. • Предоставьте ссылку на репозиторий.
Фреймворки Python	<p>Подготовка к лекции и лабораторной работе Лабораторная работа.</p> <p>Необходимо реализовать указанный функционал: Регистрация - Содержит следующие поля (все поля валидируются на стороне клиента:</p> <ul style="list-style-type: none"> ○ ФИО - содержит только кириллицу без цифр и знаков препинания. ○ E-mail - валидируется на соответствие шаблону e-mail адресов ○ Логин - должен быть уникальным ○ Пароль - должен содержать не менее 6 символов английской раскладки, верхнего и нижнего регистра ○ Подтверждение пароля - должно совпадать с полем Пароль ○ Кнопка Зарегистрироваться. <p>В случае несоответствия требованиям выводится соответствующее сообщение, поля с ошибками выделяются.</p>
Разработка интернет-приложения с использованием Python фреймворка	<p>Подготовка к лекции и лабораторной работе Лабораторная работа</p> <p>Необходимо реализовать указанный функционал для сервиса:</p> <ul style="list-style-type: none"> ● Авторизация - после авторизации простого пользователя должно отправлять в личный кабинет, а администратора в панель управления сайтом. Логин администратора: admin, пароль: wsr2018. При успешной авторизации пользователя перенаправляет на страницу просмотра своих записей на услуги. ● Панель управления сайтом - страница доступная по адресу <сайт>/admin панель управления сайтом доступна только администратору <p>Разделы пользователя:</p> <ul style="list-style-type: none"> ● Просмотр услуг ● Записи на услугу ● Просмотр своих записей на услугу <p>Разделы администратора:</p> <ul style="list-style-type: none"> ● Добавление услуги ● Просмотр записей на услугу ● Удаление записей на услугу
Особенности разработки RestFull API	<p>Создайте RESTful API с использованием Django Rest Framework, которое предоставляет доступ к данным вашего приложения. API должно поддерживать операции CRUD (создание, чтение, обновление, удаление). API должно быть готово к интеграции с фронтендом (React/Vue.js) или мобильными приложениями.</p> <p>Требования:</p> <ul style="list-style-type: none"> • БД должна содержать не менее 3 связанных таблиц. • В API должна быть реализована обязательная аутентификация через JWT-токены. • Должна быть реализована пагинация для всех списковых представлений (List endpoints). • Документация по использованию API должна быть оформлена с помощью Swagger.

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
	<ul style="list-style-type: none"> • В интерфейсе Swagger должна быть кнопка "Authorize" для ввода JWT-токена и отправки авторизованных запросов прямо из браузера. <p>Варианты:</p> <ol style="list-style-type: none"> 1. API для туристического агентства. 2. API для волонтерского движения (привлечение новых волонтеров, организация мероприятий, информирование). 3. API для Интернет-магазина для продажи кофе. 4. API для отеля 5*. 5. API для строительной компании (строительство как для физических лиц (коттеджи, таун-хаусы и др), так и для юридических лиц (многоэтажные здания)). 6. API для Интернет-магазина для продажи компьютерной техники. 7. API для кредитного банка (кредиты, вклады). 8. API для книжного магазина. 9. API для Интернет-магазина продажи домашних животных. 10. API для биржи услуг. 11. API для фитнес-клуба. 12. API для знакомств. 13. API для частного детского сада. 14. API для автомобильного центра (продажа, ремонт авто) <p>Код опубликовать на GitHub.</p>
Основы веб-дизайна	<p>Лабораторная работа Проектирование веб-дизайна Провести UX- исследование. Данное исследование представить в виде таблиц или текста в Excel или Word.</p> <ul style="list-style-type: none"> • Опишите целевую аудиторию • Проанализируйте конкурентные веб-приложения и их дизайн (минимум 3) • Составьте вопросы для исследования • Подберите подходящие для вас методы и мероприятия для анализа восприятия продукта аудиторией на разных этапах исследования • Опишите бренд, на основе которого будет создан сайт. Разработайте логотип. • Создать дизайн сайта в Figma. Обязательные элементы: <ol style="list-style-type: none"> 1. Логотип 2. Меню 3. Баннер 4. Текстовая область 5. Фото, картинки, изображения, видео, аудио 6. Графические изображения 7. Мотивационный слоган 8. Триггеры доверия (например, наши преимущества) 9. Отзывы пользователей о сайте <p>В качестве ответа отправьте документ с исследованием и описанием брендбука. Ссылку на проект в Figma.</p> <p>Варианты сайтов:</p> <ol style="list-style-type: none"> 1. Кинотеатр 2. Салон красоты 3. Частный учебный центр 4. Знакомства 5. Компьютерная техника 6. Фотосток 7. Репетиторство 8. Ресторан/кафе 9. Караоке 10. Клуб геймеров

Раздел/тема дисциплины	Вид и задания для самостоятельной работы
	<p>*Разрешается предложить свою тематику сайта. Все тематики согласуются с преподавателем. Обратите внимание, в следующем семестре для данного сайта будет создаваться веб-приложение.</p> <p>Итоговое задание. Реализовать веб-приложение в соответствии с заданием, полученным на этапе проектирования.</p>

7 Оценочные средства для проведения промежуточной аттестации по дисциплине «Разработка веб-приложений»

а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации:

<i>Код индикатора</i>	<i>Индикатор достижения компетенции</i>	<i>Оценочные средства</i>
ОПК-7	Способен разрабатывать алгоритмы и программы, пригодные для практического применения;	
ОПК-7.1	Определяет средства разработки программных средств для решения практических задач профессиональной деятельности	<p>Теоретические вопросы (к экзамену, зачету):</p> <ol style="list-style-type: none"> 1. Структура клиент-серверных Интернет-приложений. 2. Классификация Интернет-ресурсов. 3. Подходы к разработке Интернет-приложений. 4. Механизмы взаимодействия web-сервера и клиента. 5. Концепция REST API. 6. Принципы построения Интернет-приложения на базе объектно-ориентированного подхода. <p>Практические задания:</p> <ol style="list-style-type: none"> 1. Определить тип веб-ресурса и предложить технологии, инструменты его разработки для реализации следующего бизнес-процесса: <ul style="list-style-type: none"> • Продажа одежды • Публикация статьи • Написание постов с получение обратной связи в виде лайков и комментариев • Размещение фотографий для распространения своего портфолио • Подсчет стоимости строительных услуг • Распространение информации о конференции <p>Спроектировать логическую модель для веб-приложения по он-лайн заказам продукции в кафе быстрого питания.</p> <p>Комплексное задание:</p> <ol style="list-style-type: none"> 1. Создайте API, который предоставляет информацию о валютах (название, банк, стоимость покупки и продажи (массив)). 1. Список всех валют в БД. 2. Вывод информации об отдельной валюте (по id). <p>Создайте API, который предоставляет данные об отзывах по техническим товарам (название товара, комментарий, оценка, дата комментария, имя пользователя, изображение с аватаром). 1. Список всех отзывов в БД. 2. Вывод информации об отдельном отзыве (по id).</p>
ОПК-7.2	Разрабатывает алгоритмы и программы для решения прикладных задач различных	<p>Теоретические вопросы (к экзамену, зачету):</p> <ol style="list-style-type: none"> 1. HTML5. Семантическая верстка. 2. Технология CSS3. Современные подходы к верстке. 3. Принципы юзабилити. 4. Реализация управляющих конструкций на языке JavaScript. 5. Создание пользовательских функций на языке JavaScript.

Код индикатора	Индикатор достижения компетенции	Оценочные средства																																								
	классов	<p>6. Принципы реализации безопасности Интернет-приложений.</p> <p>7. Технология Ajax в Интернет-приложениях.</p> <p>8. Принципы построения Интернет-приложения на базе объектно-ориентированного подхода.</p> <p>9. Объектная модель документа (DOM).</p> <p>10. Возможности реактивных библиотек JavaScript.</p> <p>11. Реализация управляющих конструкций на языке Python</p> <p>12. Взаимодействие Python и MySQL.</p> <p>13. Фреймворки Python.</p> <p>14. Принципы реализации безопасности Интернет-приложений.</p> <p>15. Механизмы реализации cookies и sessions в Интернет-приложениях.</p> <p>Практические задания:</p> <p>1. Создать html форму с полями: дата новости, название новости, аннотация новости, автор новости, картинка для новости. Написать скрипт, который сохраняет введенную информацию в файл на сервере, файл с картинкой сохраняет в папке Images, всю информацию добавляет в формате xml.</p> <p>2. Написать скрипты, которые реализуют следующий функционал: при авторизации администратора в интернет-приложении ему доступны страница панели администратора и страница новостной ленты; при авторизации обычного пользователя доступна только страница с новостной лентой. Авторизацию реализовать с использованием механизма сессии. Реализовать кнопку Выход (уничтожение сессии).</p> <p>3. Даны 2 связанные таблицы (см. ниже). Составить следующие запросы: 1) выводит перечень пользователей с логином, содержащим подстроку "Us"; 2) подсчитывает количество пользователей, имеющих фотографии.</p> <p>Td_users</p> <table border="1" data-bbox="678 1406 1279 1585"> <thead> <tr> <th>id</th> <th>login</th> <th>password</th> <th>email</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>User1</td> <td>123</td> <td>User1@mail.ru</td> </tr> <tr> <td>2</td> <td>User2</td> <td>1234</td> <td>User2@mail.ru</td> </tr> <tr> <td>3</td> <td>User3</td> <td>12345</td> <td>User3@mail.ru</td> </tr> <tr> <td>4</td> <td>User4</td> <td>123456</td> <td>User4@mail.ru</td> </tr> </tbody> </table> <p>Td_photos</p> <table border="1" data-bbox="678 1624 1388 1780"> <thead> <tr> <th>id</th> <th>Id_user</th> <th>File</th> <th>data</th> <th>description</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>1.png</td> <td>23/07/2019</td> <td>none</td> </tr> <tr> <td>2</td> <td>2</td> <td>foto.png</td> <td>23/11/2019</td> <td>My avatar</td> </tr> <tr> <td>3</td> <td>2</td> <td>2.png</td> <td>24/07/2019</td> <td>none</td> </tr> </tbody> </table> <p>Комплексное задание:</p> <p>Разработать интернет ресурс для главного ресторана отеля 5 *.</p> <p>*. Требования к реализации функционала:</p> <ul style="list-style-type: none"> – Мультиязычность и локализация: реализовать функционал мультиязычности для предоставления информации на нескольких языках (от 3 языков). Весь контент сайта должен отображаться на выбранном языке. 	id	login	password	email	1	User1	123	User1@mail.ru	2	User2	1234	User2@mail.ru	3	User3	12345	User3@mail.ru	4	User4	123456	User4@mail.ru	id	Id_user	File	data	description	1	1	1.png	23/07/2019	none	2	2	foto.png	23/11/2019	My avatar	3	2	2.png	24/07/2019	none
id	login	password	email																																							
1	User1	123	User1@mail.ru																																							
2	User2	1234	User2@mail.ru																																							
3	User3	12345	User3@mail.ru																																							
4	User4	123456	User4@mail.ru																																							
id	Id_user	File	data	description																																						
1	1	1.png	23/07/2019	none																																						
2	2	foto.png	23/11/2019	My avatar																																						
3	2	2.png	24/07/2019	none																																						

<i>Код индикатора</i>	<i>Индикатор достижения компетенции</i>	<i>Оценочные средства</i>
		<ul style="list-style-type: none"> – яЗаказ столика в главном ресторане отеля: пользователь должен иметь возможность выбрать дату, время, свободный столик (выводится интерактивная карта ресторана, где расставлены столики, свободные столики должны быть подсвечены), и нажать кнопку бронировать. При бронировании столика запрашивается ФИО клиента и номер комнаты проживания. – Анимация с использованием SVG - Анимация бронирования столика в ресторане: при нажатии кнопки Забронировать появляется столик к которому пододвигается стул и появляется надпись «Столик забронирован!» – Наличие авторизации пользователей в сервисе. – Разграничение прав доступа для авторизованных и неавторизованных пользователей. – Реализовать поиск и фильтрацию данных в сервисе. – Реализовать валидацию вводимых данных пользователем.

б) Порядок проведения промежуточной аттестации, показатели и критерии оценивания:

Промежуточная аттестация по дисциплине «Разработка веб-приложений» проводится в форме зачета (2 семестр) и зачета с оценкой (3 семестр).

Показатели и критерии оценивания на зачет (в соответствии с формируемыми компетенциями и планируемыми результатами обучения):

– на оценку «зачтено» – обучающийся показывает высокий уровень сформированности компетенций, т.е. выполняет тренировочные, практические и лабораторные работы в установленные сроки; разрабатывает проектные задания по дисциплине с учетом заявленных требований, владеет терминологическим аппаратом, демонстрирует глубокое теоретическое знание вопроса в области программирования, грамотно определяет логико-структурные связи, обосновывает свое решение и формулирует необходимые выводы.

– на оценку «не зачтено» – результат обучения не достигнут, обучающийся не может показать знания на уровне воспроизведения и объяснения информации, не может показать интеллектуальные навыки решения простых задач в области программирования.

Показатели и критерии оценивания зачета с оценкой:

– на оценку «отлично» – полностью выполнен объем работ за семестр, также разработано проектное задание, учащийся чётко и правильно дает определения и раскрывает содержание материала; ответ самостоятельный, при ответе использованы знания, приобретённые ранее;

– на оценку «хорошо» – задания семестра выполнены на 85-90% от всего объема работ за семестр, также разработано проектное задание, учащийся в основном правильно дает определения, понятия; при ответе допускает неточности, практические навыки нетвёрдые;

– на оценку «удовлетворительно» – задания семестра выполнены на 60-80% от всего объема работ за семестр, не разработано проектное задание, усвоено основное содержание

материала, но изложено фрагментарно, не всегда последовательно; определения и понятия даны не чётко; практические навыки слабые;

– на оценку «неудовлетворительно» – задания семестра не выполнены, основное содержание учебного материала не раскрыто; не даны ответы на дополнительные вопросы преподавателя