

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ:

Проректор по учебной работе,  
качеству образования – первый  
проректор

\_\_\_\_\_ Хагуров Т.А.

*подпись*  
« 29 » августа 2025 г.

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**Б1. О.22 WEB-разработка**

Направление подготовки 02.03.02 Фундаментальная информатика и  
информационные технологии

Профиль Современные методы машинного обучения и компьютерного зрения

Форма обучения очная

Квалификация бакалавр

Краснодар 2025

Рабочая программа дисциплины «Web-разработка» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.02 Фундаментальная информатика и информационные технологии.

Программу составил(и):

С. Г. Сеница, доцент КИТ, к.т.н.

И.О. Фамилия, должность, ученая степень, ученое звание

  
\_\_\_\_\_

подпись

Рабочая программа дисциплины «Web-разработка» утверждена на заседании кафедры информационных технологий протокол № 1 от «26» августа 2025г.

Заведующий кафедрой Подколзин В.В.



Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол №1 от «28» августа 2025 г.

Председатель УМК факультета

А. В. Коваленко



подпись

Рецензенты:

Мостовой Евгений Викторович, генеральный директор ООО «Портал-Юг»,  
e-mail: mostovoy@portal-yug.ru

Луценко Евгений Вениаминович, доктор экономических наук, кандидат технических наук, профессор кафедры компьютерных технологий и систем Федерального государственного бюджетное образовательное учреждение высшего образования «Кубанский государственный аграрный университет имени И.Т. Трубилина», e-mail: prof.lutsenko@gmail.com

## **1 Цели и задачи изучения дисциплины (модуля)**

### **1.1 Цель освоения дисциплины**

Цель дисциплины – изучение веб-технологий.

### **1.2 Задачи дисциплины**

- Изучение веб-архитектуры, протокола HTTP.
- Изучение основ фронтенд разработки, HTML, CSS, JavaScript, React.
- Изучение основ бэкенд разработки на Python.

### **1.3 Место дисциплины (модуля) в структуре образовательной программы**

Дисциплина «Web-разработка» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» учебного плана, код Б1.О.22.

Дисциплина в значительной степени **взаимодействует для формирования компетенций** с дисциплинами:

- Операционные системы.

Требованием к «входным» знаниям является понимание основ программирования на Python.

### **1.4 Профессиональные роли в структуре образовательной программы**

**Роль 1: Data Engineer (Инженер по данным)**

Задачи:

- 1 Проектирование и построение ETL-процессов
- 2 Создание и оптимизация хранилищ данных
- 3 Обеспечение качества и доступности данных
- 4 Настройка инфраструктуры для обработки больших данных
- 5 Интеграция разрозненных источников данных
- 6 Работа с данными в области природопользования, медицины, связи и телекоммуникаций

**Роль 2: ML Engineer (Инженер МО)**

Задачи:

- 1 Реализация ML-моделей в продуктивных системах
- 2 Оптимизация производительности и масштабирование моделей
- 3 Разработка ML-пайплайнов и автоматизация процессов
- 4 Мониторинг качества моделей в продуктиве
- 5 Интеграция ML-решений с бизнес-приложениями

**Роль 3: MLOps (Специалист по эксплуатации ИИ)**

Задачи:

- 1 Автоматизация процессов обучения и развертывания моделей
- 2 Мониторинг производительности ML-систем
- 3 Управление версиями моделей и данных
- 4 Обеспечение CI/CD для ML-проектов
- 5 Оптимизация вычислительных ресурсов

**1.5 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы**

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

<b>УК-1</b>	<b><i>Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач</i></b>
УК-1.1	Осуществляет поиск необходимой информации, опираясь на результаты анализа поставленной задачи
	Умеет осуществлять поиск спецификаций веб-стандартов, библиотек, компонентов и примеров для решения поставленной задачи в области веб-разработки
УК-1.2	Выбирает оптимальный вариант решения задачи, аргументируя свой выбор
	Умеет выбирать оптимальный вариант решения задачи с использованием веб-технологий, аргументируя свой выбор
<b>УК-2</b>	<b><i>Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений</i></b>
УК-2.3	Использует принципы проектной методологии для решения профессиональных задач
	Умеет декомпозировать веб-проект на задачи для работы в команде Знает инструменты командной работы над кодом и документацией проекта, такие как GitLab
УК-2.4	Выбирает оптимальный способ решения задач, имеющихся ресурсов и ограничений, оценки рисков на основе проектного инструментария
	Умеет оценивать трудоемкость задач веб-разработки, выбирать оптимальный способ решения задачи
<b>ОПК-2</b>	<b>Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности</b>
ОПК-2.1	Знает основные положения и концепции в области программирования, архитектуру языков программирования, теории коммуникации, знает основную терминологию, знаком с содержанием Единого Реестра Российских программ
	Знает основные положения и концепции в области программирования веб-приложений, архитектуру языков программирования, теории коммуникации, Знает основную терминологию, знаком с содержанием Единого Реестра Российских программ в сфере веб-разработки
ОПК-2.3	Имеет практический опыт решения задач анализа, интеграции различных типов программного обеспечения, анализа типов коммуникаций
	Знает технологии веб-сервисов Умеет разрабатывать веб-приложения Владеет методами интеграции различных типов программного обеспечения с помощью веб-технологий
<b>ОПК-3</b>	<b>Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз</b>

	данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям
ОПК-3.1	Аргументировано применяет методы проектирования, разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности
	Аргументировано применяет веб-технологии при создании программных продуктов и программных комплексов различного назначения
ОПК-3.2	Использует инструментальные, программные и аппаратные средства измерений для оценки качества программного обеспечения
	Владеет технологиями тестирования веб-приложений
<b>ОПК-6</b>	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности
ОПК-6.1	Аргументировано применяет современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения
	Умеет аргументировано применять веб-технологии при разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности
ОПК-6.2	Ориентируется в современных положениях и концепциях прикладного и системного программного обеспечения, архитектуры компьютеров и сетей (в том числе и глобальных), технологии создания и сопровождения программных продуктов и программных комплексов
	Применяет технологии разработки веб-интерфейсов и веб-сервисов
<b>LLM-5</b>	<b><i>Организует взаимодействие с генеративными моделями через проектирование, анализ и применение промптов</i></b>
LLM-5.3	Настраивает API для работы с LLM
	Интегрирует модель через API
<b>PL-1</b>	<b><i>Способен применять язык программирования Python для решения задач в области ИИ</i></b>
PL-1.1	Разрабатывает и отлаживает прикладные решения разной сложности и для разного круга конечных пользователей с использованием языка программирования Python, тестирует, испытывает и оценивает качество таких решений
	Знает основы синтаксиса языка, пишет небольшие скрипты для автоматизации ручной работы по обработке небольших объемов данных с помощью встроенных модулей и внешних библиотек (csv, json, requests). Владеет основными библиотеками для выполнения большинства рутинных задач: ввод-вывод, серверное программирование (FastAPI). Самостоятельно участвует в разработке серверных приложений и их поддержке
<b>PL-3</b>	<b><i>Способен применять языки программирования C/C++ для решения задач в области ИИ</i></b>
PL-3.2	Разрабатывает и отлаживает эффективные многопоточные решения на C++, тестирует, испытывает и оценивает качество таких решений
	Знает основы синтаксиса языка. Знает общие принципы параллельных

вычислений и понимает проблемы, возникающие при распараллеливании алгоритмов. Проводит распараллеливание обработки HTTP-запросов с применением потоков Posix, стандартных библиотек C/C++ или др.

## 2. Структура и содержание дисциплины

### 2.1 Распределение трудоёмкости дисциплины по видам работ

Общая трудоёмкость дисциплины составляет 2 зач. ед. (72 часов), их распределение по видам работ представлено в таблице

Вид учебной работы	Всего часов	Семестры (часы)					
		4					
<b>Контактная работа, в том числе:</b>	<b>52,2</b>	<b>52,2</b>					
<b>Аудиторные занятия (всего):</b>	<b>48</b>	<b>48</b>					
Занятия лекционного типа	16	16					
Лабораторные занятия	32	32					
Занятия семинарского типа (семинары, практические занятия)							
<b>Иная контактная работа:</b>	<b>2,2</b>	<b>2,2</b>					
Контроль самостоятельной работы (КСР)	2	2					
Промежуточная аттестация (ИКР)	0,2	0,2					
<b>Самостоятельная работа, в том числе:</b>	<b>21,8</b>	<b>21,8</b>					
Выполнение индивидуальных заданий	20	20					
Реферат							
Подготовка к текущему контролю	1,8	1,8					
<b>Контроль:</b>							
Подготовка к зачету							
<b>Общая трудоемкость</b>	<b>час.</b>	<b>72</b>	<b>72</b>				
	<b>в том числе контактная работа</b>	<b>52,2</b>	<b>72,2</b>				
	<b>зач. ед</b>	<b>2</b>	<b>2</b>				

### 2.2 Структура дисциплины

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы (темы) дисциплины, изучаемые в 4 семестре

№	Наименование разделов (тем)	Всего	Количество часов			
			Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	
1	2	3	4	5	6	7
1.	Фронтенд разработка	18	8		6	10
2.	Бекэнд разработка	40	8		26	10
<b>ИТОГО по разделам дисциплины</b>		<b>68</b>	<b>16</b>		<b>32</b>	<b>20</b>
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		0,2				
Подготовка к текущему контролю		1,8				
<b>Общая трудоемкость по дисциплине</b>		<b>72</b>				

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

## 2.3 Содержание разделов (тем) дисциплины

### 2.3.1 Занятия лекционного типа

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
1.	Фронтэнд разработка	HTML, гиперссылки, формы, таблицы. Расширенные возможности современных браузеров и HTML5. PWA. CMS.	ЛР
2.	Фронтэнд разработка	CSS: боксовая модель, float, flexbox, адаптивная верстка, анимации CSS. Фреймворк Bootstrap.	ЛР
3.	Фронтэнд разработка	Javascript: синтаксис, работа с DOM, события, отладка. setTimeout/setInterval, исключения, область видимости, замыкания. JavaScript: массивы, объекты, словари, JSON, ООП. JavaScript: ES6 и функциональное программирование, Promises, fetch, CORS, async/await, History API, RequestAnimationFrame.	ЛР
4.	Фронтэнд разработка	Реактивное программирование и компонентный подход, фреймворк React. Роутер. Управление состоянием веб-приложения, Redux. Библиотеки компонентов Material UI и Ant Design.	ЛР
5.	Бекэнд разработка	Основы веб-архитектуры, DNS, домены, регистраторы доменов, виды хостинга, взаимодействие браузера и веб-сервера. Веб-серверы Apache и Nginx. История развития веб-приложений и сравнение основных используемых технологий. Работа с веб-сервером по SSH, ключи, GitHub.	ЛР
6.	Бекэнд разработка	HTTP. Основные определения и принцип работы HTTP, версии протокола, общий вид запроса и ответа, примеры. Коды статусов ответа, основные заголовки запросов и ответов HTTP. Cookies, сессии, JWT, Basic аутентификация HTTP. Кеширование в HTTP. Архитектура и методы масштабирования и повышения производительности веб-приложений. Примеры на Python. Использование СУБД PostgreSQL для разработки динамических веб-приложений, подготовленные запросы. Примеры на Python	ЛР
7.	Бекэнд разработка	Понятие веб-сервиса, архитектурные стили и технологии веб-сервисов.	ЛР

№	Наименование раздела (темы)	Содержание раздела (темы)	Форма текущего контроля
1	2	3	4
		XML/JSON по HTTP, XMLRPC, SOAP. Принципы и ограничения REST. Пример REST веб-сервиса на Python/FastAPI	
8.	Бекэнд разработка	Типы уязвимостей веб-приложений. Характеристика и защита от уязвимостей XSS, SQL Injection, CSRF, Upload, Include. Защита клиента и сервера веб-приложения. OWASP, WebAppSec, CVE, Exploit DB. HTTPS, PKI, основные понятия и принцип работы, ГОСТы и стандарты криптографии, ЭЦП, ЭДО.	ЛР

Примечание: ЛР – отчет/защита лабораторной работы, КП - выполнение курсового проекта, КР - курсовой работы, РГЗ - расчетно-графического задания, Р - написание реферата, Э - эссе, К - коллоквиум, Т – тестирование, РЗ – решение задач.

### 2.3.2 Занятия семинарского типа

Не предусмотрены

### 2.3.3 Лабораторные занятия

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
1.	Фронтэнд разработка	Сверстать HTML-страницу с нумерованным списком гиперссылок и формой	ЛР
2.	Фронтэнд разработка	Сверстать адаптивно шапку, футер, меню по схематичному макету с помощью FlexBox. В контенте сверстать таблицу	ЛР
3.	Фронтэнд разработка	Форма с обработчиком событий JavaScript. Отправка на сервер с помощью fetch/async	ЛР
4.	Фронтэнд разработка	Vite + React, создание проекта	ЛР
5.	Фронтэнд разработка	Библиотеки компонентов React	ЛР
6.	Фронтэнд разработка	Приложение на React, роутер	ЛР
7.	Фронтэнд разработка	Приложение на React, форма, валидация	ЛР
8.	Фронтэнд разработка	Приложение на React, хранение состояния	ЛР
9.	Бекэнд разработка	Настройка веб-сервера и GitLab по SSH, ping, whois, nslookup, FTP, GIT. HTTP-запросы.	ЛР
10.	Бекэнд разработка	Разработка динамического веб-приложения на Python. Создание базы данных PostgreSQL, выборка и вставка данных.	ЛР
11.	Бекэнд разработка	Бекэнд на FastAPI, валидация данных на сервере	ЛР
12.	Бекэнд разработка	Авторизация на сайте с использованием JWT и Cookies	ЛР
13.	Бекэнд разработка	Отчеты и запросами к БД с группировкой	ЛР

№	Наименование раздела (темы)	Наименование лабораторных работ	Форма текущего контроля
1	2	3	4
14.	Бекэнд разработка	Интеграция фронтэнд части. Аудит безопасности веб-приложения	ЛР
15.	Бекэнд разработка	Веб-сервер на C/C++	ЛР
16.	Бекэнд разработка	Gradio для быстрого создания веб-интерфейса к LLM	ЛР

Примечание: ЛР – отчет/защита лабораторной работы, КП – выполнение курсового проекта, КР – курсовой работы, РГЗ – расчетно-графического задания, Р – написание реферата, Э – эссе, К – коллоквиум, Т – тестирование, РЗ – решение задач.

### 2.3.4 Примерная тематика курсовых работ (проектов)

Курсовая работа не предусмотрена.

### 2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Изучение теоретического материала	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019
2	Решение задач	Методические указания по организации самостоятельной работы студентов, утвержденные кафедрой информационных технологий, протокол №1 от 30.08.2019

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

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

## 3. Образовательные технологии

В соответствии с требованиями ФГОС в программа дисциплины предусматривает использование в учебном процессе следующих образовательные технологии: чтение

лекций с использованием мультимедийных технологий; метод малых групп, разбор практических задач и кейсов.

При обучении используются следующие образовательные технологии:

- Технология коммуникативного обучения – направлена на формирование коммуникативной компетентности студентов, которая является базовой, необходимой для адаптации к современным условиям межкультурной коммуникации.
- Технология разноуровневого (дифференцированного) обучения – предполагает осуществление познавательной деятельности студентов с учётом их индивидуальных способностей, возможностей и интересов, поощряя их реализовывать свой творческий потенциал. Создание и использование диагностических тестов является неотъемлемой частью данной технологии.
- Технология модульного обучения – предусматривает деление содержания дисциплины на достаточно автономные разделы (модули), интегрированные в общий курс.
- Информационно-коммуникационные технологии (ИКТ) - расширяют рамки образовательного процесса, повышая его практическую направленность, способствуют интенсификации самостоятельной работы учащихся и повышению познавательной активности. В рамках ИКТ выделяются 2 вида технологий:
- Технология использования компьютерных программ – позволяет эффективно дополнить процесс обучения языку на всех уровнях.
- Интернет-технологии – предоставляют широкие возможности для поиска информации, разработки научных проектов, ведения научных исследований.
- Технология индивидуализации обучения – помогает реализовывать личностно-ориентированный подход, учитывая индивидуальные особенности и потребности учащихся.
- Проектная технология – ориентирована на моделирование социального взаимодействия учащихся с целью решения задачи, которая определяется в рамках профессиональной подготовки, выделяя ту или иную предметную область.
- Технология обучения в сотрудничестве – реализует идею взаимного обучения, осуществляя как индивидуальную, так и коллективную ответственность за решение учебных задач.
- Игровая технология – позволяет развивать навыки рассмотрения ряда возможных способов решения проблем, активизируя мышление студентов и раскрывая личностный потенциал каждого учащегося.
- Технология развития критического мышления – способствует формированию разносторонней личности, способной критически относиться к информации, умению отбирать информацию для решения поставленной задачи.

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

Основные виды интерактивных образовательных технологий включают в себя:

- работа в малых группах (команде) - совместная деятельность студентов в группе под руководством лидера, направленная на решение общей задачи путём творческого сложения результатов индивидуальной работы членов команды с делением полномочий и ответственности;
- проектная технология - индивидуальная или коллективная деятельность по отбору, распределению и систематизации материала по определенной теме, в результате которой составляется проект;

- анализ конкретных ситуаций - анализ реальных проблемных ситуаций, имевших место в соответствующей области профессиональной деятельности, и поиск вариантов лучших решений;
- развитие критического мышления – образовательная деятельность, направленная на развитие у студентов разумного, рефлексивного мышления, способного выдвинуть новые идеи и увидеть новые возможности.

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

Семестр	Вид занятия	Используемые интерактивные образовательные технологии	количество интерактивных часов
4	ЛР	Практические занятия в режимах взаимодействия «преподаватель – студент» и «студент – студент»	32
<b>Итого</b>			<b>32</b>

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

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

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

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

Для лиц с нарушениями зрения:

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

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

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

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

## **4. Оценочные и методические материалы**

### **4.1 Оценочные средства для текущего контроля успеваемости и промежуточной аттестации**

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «Web-разработка».

Оценочные средства включает контрольные материалы для проведения **текущего контроля** в форме оценки лабораторных работ и зачета.

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

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

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

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

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

Для лиц с нарушениями зрения:

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

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

### Структура оценочных средств для текущей и промежуточной аттестации

№ п/п	Контролируемые разделы (темы) дисциплины*	Код контролируемой компетенции (или ее части)	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
1	Фронтэнд разработка	УК-1; УК-2; ОПК-2; ОПК-3; ОПК-6	<i>ЛР 1-8</i>	<i>Зачет</i>
2	Бекэнд разработка	LLM-5, PL-1; PL-3	<i>ЛР 9-16</i>	<i>Зачет</i>

### Показатели, критерии и шкала оценки сформированных компетенций

Соответствие **базовому уровню** освоения компетенций планируемым результатам обучения и критериям их оценивания (оценка: **зачтено**):

<b>УК-1</b>	<b><i>Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач</i></b>
УК-1.1	Осуществляет поиск необходимой информации, опираясь на результаты анализа поставленной задачи
	Умеет осуществлять поиск спецификаций веб-стандартов, библиотек, компонентов и примеров для решения поставленной задачи в области веб-

	разработки
УК-1.2	Выбирает оптимальный вариант решения задачи, аргументируя свой выбор
	Умеет выбирать оптимальный вариант решения задачи с использованием веб-технологий, аргументируя свой выбор
<b>УК-2</b>	<b><i>Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений</i></b>
УК-2.3	Использует принципы проектной методологии для решения профессиональных задач
	Умеет декомпозировать веб-проект на задачи для работы в команде Знает инструменты командной работы над кодом и документацией проекта, такие как GitLab
УК-2.4	Выбирает оптимальный способ решения задач, имеющихся ресурсов и ограничений, оценки рисков на основе проектного инструментария
	Умеет оценивать трудоемкость задач веб-разработки, выбирать оптимальный способ решения задачи
<b>ОПК-2</b>	<b>Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности</b>
ОПК-2.1	Знает основные положения и концепции в области программирования, архитектуру языков программирования, теории коммуникации, знает основную терминологию, знаком с содержанием Единого Реестра Российских программ
	Знает основные положения и концепции в области программирования веб-приложений, архитектуру языков программирования, теории коммуникации, Знает основную терминологию, знаком с содержанием Единого Реестра Российских программ в сфере веб-разработки
ОПК-2.3	Имеет практический опыт решения задач анализа, интеграции различных типов программного обеспечения, анализа типов коммуникаций
	Знает технологии веб-сервисов Умеет разрабатывать веб-приложения Владеет методами интеграции различных типов программного обеспечения с помощью веб-технологий
<b>ОПК-3</b>	<b>Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям</b>
ОПК-3.1	Аргументировано применяет методы проектирования, разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности
	Аргументировано применяет веб-технологии при создании программных продуктов и программных комплексов различного назначения
ОПК-3.2	Использует инструментальные, программные и аппаратные средства

	измерений для оценки качества программного обеспечения
	Владеет технологиями тестирования веб-приложений
<b>ОПК-6</b>	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности
ОПК-6.1	Аргументировано применяет современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения
	Умеет аргументировано применять веб-технологии при разработки и реализации программных продуктов и программных комплексов в различных областях человеческой деятельности
ОПК-6.2	Ориентируется в современных положениях и концепциях прикладного и системного программного обеспечения, архитектуры компьютеров и сетей (в том числе и глобальных), технологии создания и сопровождения программных продуктов и программных комплексов
	Применяет технологии разработки веб-интерфейсов и веб-сервисов
<b>LLM-5</b>	<b><i>Организует взаимодействие с генеративными моделями через проектирование, анализ и применение промптов</i></b>
LLM-5.3	Настраивает API для работы с LLM
	Интегрирует модель через API
<b>PL-1</b>	<b><i>Способен применять язык программирования Python для решения задач в области ИИ</i></b>
PL-1.1	Разрабатывает и отлаживает прикладные решения разной сложности и для разного круга конечных пользователей с использованием языка программирования Python, тестирует, испытывает и оценивает качество таких решений
	Знает основы синтаксиса языка, пишет небольшие скрипты для автоматизации ручной работы по обработке небольших объемов данных с помощью встроенных модулей и внешних библиотек (csv, json, requests). Владеет основными библиотеками для выполнения большинства рутинных задач: ввод-вывод, серверное программирование (FastAPI). Самостоятельно участвует в разработке серверных приложений и их поддержке
<b>PL-3</b>	<b><i>Способен применять языки программирования C/C++ для решения задач в области ИИ</i></b>
PL-3.2	Разрабатывает и отлаживает эффективные многопоточные решения на C++, тестирует, испытывает и оценивает качество таких решений
	Знает основы синтаксиса языка. Знает общие принципы параллельных вычислений и понимает проблемы, возникающие при распараллеливании алгоритмов. Проводит распараллеливание обработки HTTP-запросов с применением потоков Posix, стандартных библиотек C/C++ или др.

**Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы**

### **Практические задания для лабораторных работ**

## ЗАДАНИЕ 1

Сверстать HTML 5 страницу с нумерованным списком гиперссылок и формой. Список должен содержать следующие гиперссылки: 1) абсолютную гиперссылку на главную страницу сайта example.com; 2) абсолютную на главную сайта example.com в протоколе https; 3) ссылку на файл на сервере FTP без авторизации; 4) ссылку на файл на сервере FTP с авторизацией; 5) ссылку на фрагмент страницы некоторого сайта; 6) ссылку на фрагмент текущей страницы; 7) ссылку с двумя параметрами в URL; 8) список ссылок основной навигации сайта с подписями title; 9) ссылку без href; 10) ссылку с пустым href; 11) ссылку, по которой запрещен переход поисковикам; 12) запрещенную для индексации поисковиками; 13) контекстную в тексте абзаца; 14) ссылку-изображение; 15) ссылки из прямоугольных и круглых областей картинки (HTML-тег map); 16) относительную на страницу в текущем каталоге; 17) относительную на страницу в каталоге about; 18) относительную на страницу в каталоге уровнем выше текущего; 19) относительную на страницу в каталоге двумя уровнями выше; 20) сокращенную на главную; 21) сокращенную ссылку на внутреннюю. Форма должна содержать следующие поля: 1) имя (текстовое поле); 2) e-mail (поле email); 3) дата рождения (поле даты); 4) пол (радиокнопки); 5) количество конечностей (радиокнопки); 6) сверхспособности: бессмертие, прохождение сквозь стены, левитация (множественный выбор из списка); 7) биография (многострочное текстовое поле); 8) с контрактом ознакомлен (а) (чекбокс); 9) кнопка «Отправить».

## ЗАДАНИЕ 2

Добавить к странице в задании 1 таблицу, шапку, логотип, название сайта, подвал и меню со ссылками по схематичному макету: Таблица содержит строку-заголовок с названиями колонок, не менее 3 колонок и 5 строк, использует объединенные ячейки. Фон четных и нечетных строк различается. Шапка и подвал тянутся на всю ширину окна. Максимальная ширина контента и меню 960px с выравниванием по центру. Логотип обтекает слева название сайта, по горизонтали слева выравнивается по границе контента. CSS вынести в отдельный файл и подключить стили через id и классы. Использовать теги header, footer, nav. Сверстать страницу адаптивно, Mobile First, с применением Flex Box и Media Query. 1. На смартфонах в портретном режиме ссылки в главном меню выводятся вертикальным списком. Шапка, навигация, контент и футер занимают 100% ширины экрана: На планшетах и десктопах страница выглядит как в предыдущем задании. Использовать Flex Box и Media Query. 2. Первыми грузятся правила для мобильных телефонов. Затем, с помощью Media Query, подключаются правила для остальных размеров экрана (Mobile First). 3. В потоке HTML таблица предшествует форме. На смартфоне необходимо отобразить форму визуально выше таблицы (использовать Flex Box order).

## ЗАДАНИЕ 3

Реализовать форму обратной связи с полями Имя, Email, сообщение, чекбокс согласия с политикой обработки персональных данных. На смартфоне форма должна полностью помещаться в экран без прокрутки. Внешний вид формы произвольный. Данные из формы необходимо отправлять по нажатию кнопки Отправить без перезагрузки страницы с помощью fetch и сохранять на сервер. Использовать сервер готового бекэнда для сохранения форм, например formcarry.com или slapform.com. После успешной отправки или ошибки отображать пользователю сообщение. Последние введенные в форму значения сохранять и восстанавливать при повторном открытии страницы с помощью LocalStorage API. При отправке формы очищать данные.

## ЗАДАНИЕ 4

Установить NodeJS, Vite + React. Сгенерировать и изучить структуру шаблона проекта.

Перенести верстку из задания 1 в виде отдельных компонентов.

#### ЗАДАНИЕ 5

Сверстать страницу из задания 3 с помощью фреймворка Material UI или Ant Design. Реализовать расположение элементов по сетке (допускается изменение размеров элементов в задании 3 для выравнивания по сетке). При реализации адаптивности применить принцип Mobile First. Для управления отступами margin и padding использовать средства фреймворка. Реализовать форму задания 3 с валидацией на клиенте с помощью React.

#### ЗАДАНИЕ 6

Реализовать галерею из 16 изображений одинакового размера в виде горизонтального слайдера. Всего на экране помещается четыре изображения, на смартфонах два. Галерея прокручивается влево и вправо без перезагрузки страницы кликом на иконки стрелок. Под галереей расположен «пейджер», отображающий количество страниц и текущую страницу.

#### ЗАДАНИЕ 7

Добавить в приложение React роутер, меню, несколько страниц.

#### ЗАДАНИЕ 8

Реализовать сохранение состояния с помощью Redux.

#### ЗАДАНИЕ 9

Взять у преподавателя практики ваш логин и пароль аккаунту в облаке для настройки собственного веб-сервера. Подключиться к серверу по SSH с помощью клиента Putty, команды ssh или онлайн-версии ssh-клиента. 2) С помощью команды ping на учебном сервере узнать IP-адрес веб-сервера kubsu.ru, сделать скриншот вызова команды ping, добавить скриншот в git репозиторий задания. 3) С помощью команды nslookup узнать A-записи и MX-записи домена kubsu.ru и kubsu-dev.ru, добавить скриншоты в git-репозиторий задания. 4) С помощью команды whois узнать дату регистрации домена kubsu.ru и kubsudev.ru, добавить скриншоты в git-репозиторий задания. 5) Сделать веб-страницу index.html со скриншотами, добавить ее в git, с помощью SSH скопировать репозиторий со скриншотами и страницей в каталог /var/www/html/ ваш\_логин/. Веб-страница должна открываться по адресу http://ваш\_логин.kubsudev.ru/каталог\_задания\_1/ 6) С помощью программы FileZilla или любого другого клиента FTP соединиться с учебным сервером с вашим логином и паролем по протоколу FTP и скопировать на локальный компьютер файлы задания из каталога /var/www/html/ваш\_логин/. Сделать скриншот FTP-клиента после скачивания файлов, добавить скриншот в GIT и на страницу. Установить NGINX для отдачи статических файлов. Залить файлы в каталоге files на веб-сервер через GIT. Проверить загрузку файлов в браузере. С помощью программы telnet или Putty выполнить задания отправкой HTTP-запросов к веб-серверу: 1) получить главную страницу методом GET в протоколе HTTP 1.0; 2) получить внутреннюю страницу методом GET в протоколе HTTP 1.1; 3) определить размер файла file.tar.gz, не скачивая его; 4) определить медиатип ресурса /image.png; 5) отправить комментарий на сервер по адресу /; 6) получить первые 100 байт файла /file.tar.gz; 7) определить кодировку ресурса /. Снять 7 скриншотов, отображающих запрос и заголовки ответа веб-сервера. Сверстать веб-страницу со скриншотами, залить на веб-сервер через GIT.

#### ЗАДАНИЕ 10

Для формы из задания 6 напишите веб-сервер на Python, проверяющий все поля и в случае их корректного заполнения записывающий данные в БД. Создайте базу данных

PostgreSQL в 3-й нормальной форме и добавьте вставку данных формы после успешной валидации. Проверьте вставленные данные в консоли psql.

#### ЗАДАНИЕ 11

Для формы из задания 6 напишите веб-сервер на FastAPI, проверяющий все поля и в случае их корректного заполнения записывающий данные в файл на сервере.

#### ЗАДАНИЕ 12

Реализовать возможность входа с паролем и логином с использованием JWT и Cookies для изменения отправленных данных в предыдущей задаче. Пароль и логин генерируются автоматически при первоначальной отправке формы и отображаются пользователю при успешной отправке формы. В базе данных сохранять хеш пароля.

#### ЗАДАНИЕ 13

Реализуйте для авторизованного пользователя отчет и запросом к БД с группировкой для вывода количества пользователей, которые любят каждый из языков программирования.

#### ЗАДАНИЕ 14

Интеграция фронтэнд части задания 6. Проведите аудит безопасности вашего приложения и исправьте уязвимости. В нем должны быть разделы, посвященные уязвимостям XSS, SQL Injection, CSRF, Include, Upload. В отчете укажите по каждой уязвимости примененные методы защиты с примерами вашего кода.

#### ЗАДАНИЕ 15

Разработайте многопоточный веб-сервер на C/C++, отдающий веб-страницу.

#### ЗАДАНИЕ 16

Создайте веб-интерфейс к LLM с помощью Gradio.

### **Зачетно-экзаменационные материалы для промежуточной аттестации (зачет)**

Проект команды (3-4 человека):

По заданию от индустриального партнера разработайте веб-приложение, использующее вызовы в LLM по API.

*Перечень компетенций (части компетенции), проверяемых оценочным средством УК-1; УК-2; ОПК-2; ОПК-3; ОПК-6; LLM-5, PL-1; PL-3*

### **4.2 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

#### **Методические рекомендации, определяющие процедуры оценивания на экзамене:**

Процедура промежуточной аттестации проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся ФГБОУ ВО «КубГУ».

Итоговой формой контроля сформированности компетенций у обучающихся по дисциплине является зачет. Студенты обязаны сдать зачет в соответствии с расписанием и учебным планом.

ФОС промежуточной аттестации состоит из заданий и результатов текущего контроля.

Форма проведения экзамена: устно (защита проекта).

Преподавателю предоставляется право задавать студентам дополнительные вопросы по всей учебной программе дисциплины.

Результат сдачи зачета заносится преподавателем в экзаменационную ведомость и зачетную книжку.

Оценивание уровня освоения дисциплины основывается на качестве выполнения студентом заданий текущего контроля и проекта.

**Критерии оценки:**

Зачтено – выполнено 60% лабораторных работ и выполнен проект.

Не зачтено – выполнено менее 60% лабораторных работ или не выполнен проект.

***Методические рекомендации, определяющие процедуры оценивания лабораторных работ:***

Процедура оценивания лабораторных работ проходит в соответствии с Положением о текущем контроле и промежуточной аттестации обучающихся ФГБОУ ВО «КубГУ».

По каждой лабораторной работе оформляется отчет. Отчеты сдаются на проверку руководителю в течение курса по мере их выполнения, и защищаются студентами в установленном порядке.

При защите отчета студенту могут быть заданы вопросы и дополнительные задания по сути лабораторной работы, в том числе из списка контрольных вопросов к данной лабораторной работе. При неудовлетворительной оценке знаний студента по теме данного отчета, студент возвращается к повторному изучению соответствующих материалов, после чего допускается к повторной защите. Неудовлетворительно выполненный отчет также возвращается на доработку.

Отчет должен содержать заголовок, тему лабораторной работы, цель, задание, индивидуальную тему, описание хода выполнения работы, необходимые прикладные материалы (схемы, макеты документов и т.п.), в соответствии с требованиями к содержанию, и выводы по работе.

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

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

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

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

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

Для лиц с нарушениями зрения:

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

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

### **4.3. Методические указания по организации вычислительной инфраструктуры**

#### **Условия применения:**

- Курс рассчитан на студентов 2-го года обучения.
- Наличие доступа к вычислительным ресурсам (облако Yandex.Cloud или cloud.ru).
- Инфраструктура включает виртуальные машины для развертывания серверов.

#### **Цели, задачи и ожидаемые результаты:**

##### **Цель:**

Обеспечить студентов инструментами для развертывания виртуальных машин.

##### **Задачи преподавателя:**

- 1 Создание учетных записей в облаке.
- 2 Настройка прав доступа.
- 3 Настройка шаблонного репозитория для лабораторных работ с примерами на HTML/CSS/JS, React и FastAPI.
- 4 Подготовка инструкций по работе с облачными ресурсами.

##### **Ожидаемые результаты студентов:**

1. Умение работать облачной инфраструктурой на уровне создания и настройки виртуальных машин.
2. Умение работать с GIT и GitLab в команде.

##### **Порядок реализации:**

1. Создание учетных записей в облаке.

Каждый студент получает аккаунт в облаке курса.

2. Настройка прав доступа.

Студенты получают доступ к созданию виртуальных машин.

3. Настройка шаблонного репозитория для лабораторных работ.

Репозиторий содержит примеры HTML/CSS/JS, React и FastAPI.

4. Подготовка инструкции по работе с облачными ресурсами.

Инструкция содержит последовательность шагов, необходимых для настройки виртуальной машины.

##### **Порядок проверки корректности:**

- Наличие аккаунтов и доступов в облаке у всех студентов.
- Наличие репозитория для лабораторных работ.
- Инструкция по работе с облаком и примерами в формате README.md.

### **4.4. Методические указания по организации лабораторных работ**

#### **Условия применения:**

- Курс рассчитан на студентов 2-го года обучения.
- Наличие доступа к облачным ресурсам.

## **Цели, задачи и ожидаемые результаты:**

- **Цель:**

Практическое освоение веб-разработки.

- **Задачи преподавателя:**

1. Разработка плана лабораторных работ.
2. Организация Git-инфраструктуры.

- **Ожидаемые результаты студентов:**

1. Умение разрабатывать фронтэнд веб-приложения.
2. Умение разрабатывать бекэнд веб-приложения.

## **Порядок реализации:**

1. **План лабораторных работ:**

Включает:

- Настройку виртуальной машины.
- Настройку проекта GitLab.
- Разработку фронтэнда приложения.
- Разработку бекэнда приложения.

## **Порядок проверки корректности:**

- Наличие выполненных заданий в Git-репозиториях.
- Наличие прототипа веб-приложения в облаке.
- Ссылки для проверки заданий в Moodle.

## **5. Перечень учебной литературы, информационных ресурсов и технологий**

### **5.1. Учебная литература**

- 1 React. Быстрый старт, 2-е изд. Стоян Стефанов. Питер, O'Reilly, 2023.
- 2 FastAPI: веб-разработка на Python. Билл Любанович. SPRINT book, O'Reilly, 2024.
- 3 Документация GitLab.
- 4 Документация Yandex Cloud.
- 5 Документация React.
- 6 Документация Fast API.
- 7 Документация Gradio.
- 8 Web-стандарты W3C.
- 9 Ресурсы <https://htmlbook.ru/> и <https://learn.javascript.ru/>

### **5.2. Периодические издания:**

- 1 Базы данных компании «Ист Вью» <http://dlib.eastview.com>
- 2 Электронная библиотека GREBENNIKON.RU <https://grebennikon.ru/>

### **5.3. Интернет-ресурсы, в том числе современные профессиональные базы данных и информационные справочные системы**

*Электронно-библиотечные системы (ЭБС):*

- 1 ЭБС «ЮРАЙТ» <https://urait.ru/>
- 2 ЭБС «УНИВЕРСИТЕТСКАЯ БИБЛИОТЕКА ОНЛАЙН» <http://www.biblioclub.ru/>
- 3 ЭБС «BOOK.ru» <https://www.book.ru>
- 4 ЭБС «ZNANIUM.COM» [www.znanium.com](http://www.znanium.com)
- 5 ЭБС «ЛАНЬ» <https://e.lanbook.com>

*Профессиональные базы данных*

- 1 Scopus <http://www.scopus.com/>

- 2 ScienceDirect <https://www.sciencedirect.com/>
- 3 Журналы издательства Wiley <https://onlinelibrary.wiley.com/>
- 4 Научная электронная библиотека (НЭБ) <http://www.elibrary.ru/>
- 5 Полнотекстовые архивы ведущих западных научных журналов на Российской платформе научных журналов НЭИКОН <http://archive.neicon.ru>
- 6 Национальная электронная библиотека (доступ к Электронной библиотеке диссертаций Российской государственной библиотеки (РГБ) <https://rusneb.ru/>
- 7 Президентская библиотека им. Б.Н. Ельцина <https://www.prlib.ru/>
- 8 База данных CSD Кембриджского центра кристаллографических данных (CCDC) <https://www.ccdc.cam.ac.uk/structures/>
- 9 Springer Journals: <https://link.springer.com/>
- 10 Springer Journals Archive: <https://link.springer.com/>
- 11 Nature Journals: <https://www.nature.com/>
- 12 Springer Nature Protocols and Methods: <https://experiments.springernature.com/sources/springer-protocols>
- 13 Springer Materials: <http://materials.springer.com/>
- 14 Nano Database: <https://nano.nature.com/>
- 15 Springer eBooks (i.e. 2020 eBook collections): <https://link.springer.com/>
- 16 "Лекториум ТВ" <http://www.lektorium.tv/>
- 17 Университетская информационная система РОССИЯ <http://uisrussia.msu.ru>

#### *Бесплатные образовательные ресурсы*

- 1 Visual Studio Code – редактор кода с поддержкой Python
- 2 Google Scholar/arXiv – доступ к научным публикациям

#### *Ресурсы свободного доступа*

- 1 КиберЛенинка <http://cyberleninka.ru/>;
- 2 Американская патентная база данных <http://www.uspto.gov/patft/>
- 3 Министерство науки и высшего образования Российской Федерации <https://www.minobrnauki.gov.ru/>;
- 4 Федеральный портал "Российское образование" <http://www.edu.ru/>;
- 5 Информационная система "Единое окно доступа к образовательным ресурсам" <http://window.edu.ru/>;
- 6 Единая коллекция цифровых образовательных ресурсов <http://school-collection.edu.ru/> .
- 7 Проект Государственного института русского языка имени А.С. Пушкина "Образование на русском" <https://pushkininstitute.ru/>;
- 8 Справочно-информационный портал "Русский язык" <http://gramota.ru/>;
- 9 Служба тематических толковых словарей <http://www.glossary.ru/>;
- 10 Словари и энциклопедии <http://dic.academic.ru/>;
- 11 Образовательный портал "Учеба" <http://www.ucheba.com/>;
- 12 Законопроект "Об образовании в Российской Федерации". Вопросы и ответы [http://xn--273--84d1f.xn--p1ai/voprosy\\_i\\_otvety](http://xn--273--84d1f.xn--p1ai/voprosy_i_otvety)

#### *Собственные электронные образовательные и информационные ресурсы КубГУ*

- 1 Электронный каталог Научной библиотеки КубГУ <http://megapro.kubsu.ru/MegaPro/Web>
- 2 Электронная библиотека трудов ученых КубГУ <http://megapro.kubsu.ru/MegaPro/UserEntry?Action=ToDb&idb=6>
- 3 Среда модульного динамического обучения <http://moodle.kubsu.ru>
- 4 База учебных планов, учебно-методических комплексов, публикаций и конференций <http://infoneeds.kubsu.ru/>

- 5 Библиотека информационных ресурсов кафедры информационных образовательных технологий <http://mschool.kubsu.ru>;
- 6 Электронный архив документов КубГУ <http://docspace.kubsu.ru/>
- 7 Электронные образовательные ресурсы кафедры информационных систем и технологий в образовании КубГУ и научно-методического журнала "ШКОЛЬНЫЕ ГОДЫ" <http://icdau.kubsu.ru/>

#### **5.4 Перечень информационно-коммуникационных технологий**

1. Облачные платформы и сервисы  
cloud.ru, YandexCloud, AWS/GCP/Azure – облачные вычисления
2. Системы управления версиями и коллаборации  
Git/GitHub/GitLab – контроль версий кода и совместная разработка
4. Система управления обучением  
Moodle – сдача работ

#### **5.5 Перечень лицензионного и свободно распространяемого программного обеспечения**

1. Свободное ПО (Open Source)  
GitLab, GIT, VS Code.

### **6. Методические указания для обучающихся по освоению дисциплины (модуля)**

По курсу предусмотрено проведение лекционных занятий, на которых дается систематизированный материал по веб-разработке. В ходе лекций рассматриваются ключевые концепции. После каждой лекции рекомендуется выполнение практических заданий для закрепления ключевых понятий и методов.

Лабораторные занятия курса посвящены практическому освоению веб-разработки. За основу берется веб-приложение, использующее LLM API для решения задач от промышленных партнеров.

При самостоятельной работе студентам необходимо изучать рекомендованную литературу в виде официальной документации к используемым открытым программным продуктам, облачным платформам.

Важнейшим компонентом курса является самостоятельная проектная работа, в ходе которой студент разрабатывает законченное решение с уровнем технологической готовности (УТГ) 5-7 с применением CI/CD/CT для решения задач (кейсов) промышленных партнеров. Допускается выполнение проектов в командах 3-4 человека.

#### **Кейсы ПАО «Сбербанк»**

##### **1. Генеративный ИИ для автоматического составления инвестиционных обзоров**

###### **Описание:**

Аналитики Сбера ежедневно составляют десятки аналитических и инвестиционных обзоров по рынкам, компаниям, макроэкономике. Задача — исследовать применение LLM для генерации кратких сводок и аналитических отчетов на основе входных данных: биржевые котировки, макроэкономические показатели, рыночные события.

###### **Цель:**

Разработать инструмент, способный по структурированным данным и краткому описанию формировать инвестиционный обзор в деловом стиле.

**Ожидаемый результат:**

Модель, генерирующая аналитические тексты длиной 500–1000 слов с разделами «обзор событий», «рекомендации», «прогнозы», оформленные в формате банка.

**2. NLP-анализ жалоб клиентов в свободной форме****Описание:**

В рамках клиентского сервиса Сбербанк обрабатывает обращения из чатов, мобильного приложения и жалобной формы. Требуется построить модель семантического анализа, выделяющую суть обращения, определяющую тональность и потенциальную серьезность инцидента.

**Цель:**

Автоматизировать классификацию обращений для ускорения маршрутизации и выявления повторяющихся болевых точек в продуктах и процессах.

**Ожидаемый результат:**

Прототип модели, автоматически выделяющей темы жалоб (например, «ошибка в приложении», «двойное списание»), их эмоциональную окраску и критичность.

**3. Генерация сценариев фишинговых писем для обучения сотрудников****Описание:**

Банк проводит киберучения, включая рассылку тестовых фишинговых писем сотрудникам для повышения их устойчивости к социальным атакам. Проект предполагает использование генеративной модели для создания реалистичных фишинговых писем различных типов (поддельные счета, HR-запросы, ИТ-поддержка).

**Цель:**

Создать генератор, способный на основе заданных параметров (тема, стиль, уровень угрозы) создавать тексты фишинга для тренировок.

**Ожидаемый результат:**

Набор разнообразных примеров фишинга и оценка их эффективности по реакции сотрудников, а также классификация моделей угроз.

**Кейсы от «АВАЛАБ»****1. LLM и RAG для BI-системы Fastboard****Описание:**

Для разрабатываемой компанией BI-системы Fastboard требуется разработать интерфейс на естественном языке для построения отчетов на больших массивах данных в ClickHouse. С помощью LLM необходимо классифицировать запросы пользователей на естественном языке и извлекать фактические параметры для дальнейшего вызова веб-сервиса отчетов.

**Цель:**

Разработать промпты для классификации и обработки запросов пользователей LLM и преобразования их к вызовам типовых отчетов с фактическими параметрами, извлекаемыми из запроса.

**Ожидаемый результат:**

Инструмент на основе LLM, позволяющий запрашивать данные о продажах.

**2. Анализ обращений клиентов и CRM-переписки****Описание:**

В службе клиентского сервиса застройщика ежедневно обрабатываются десятки обращений (e-mail, звонки, мессенджеры). Требуется реализовать систему семантического анализа и классификации NLU: выявлять суть обращений, уровень удовлетворенности, отслеживать повторяющиеся запросы.

**Цель:**

Автоматизировать первичный разбор и маршрутизацию запросов по тематике (сдача объекта, отделка, документы, жалоба и т.д.).

**Ожидаемый результат:**

Прототип, который выделяет суть обращений и формирует дашборд по текущим «болям» клиентов.

**3. Генерация рекламного контента для жилых комплексов****Описание:**

«АВА ГРУПП» регулярно запускает маркетинговые кампании для жилых комплексов. Необходимо исследовать использование диффузионных моделей для генерации изображений (визуализации интерьеров, окрестностей, видов из окон) и LLM — для описаний квартир, преимуществ района, инфраструктуры.

**Цель:**

Создать инструменты для быстрой генерации продающих материалов без привлечения дизайнеров и копирайтеров на первых этапах.

**Ожидаемый результат:**

Набор сгенерированных карточек объектов с текстом, изображением и логикой «живого» рекламного сообщения.

**4. Генерация документации и шаблонов договоров****Описание:**

Юридический департамент регулярно работает с договорами долевого участия, актами приёма-передачи и другими документами. Использование LLM может значительно сократить время на подготовку черновиков — достаточно ввести параметры сделки.

**Цель:**

Создать систему, которая генерирует адаптированные тексты документов по вводным данным (тип объекта, этаж, площадь, ФИО, сроки и пр.).

**Ожидаемый результат:**

Генератор документов в формате Word или PDF с автоматической подстановкой параметров и соблюдением юридического стиля.

**5. Обратная генерация — ИИ-помощник для покупателей квартир****Описание:**

Будущие покупатели часто задают типовые вопросы о квартирах, планировках, ипотеке, акциях, сроках. Вместо call-центра предлагается реализовать LLM-бота, который обрабатывает текстовые и голосовые запросы, показывает планировки, ссылается на PDF-документы и может «объяснять» информацию простым языком.

**Цель:**

Упростить коммуникацию с клиентами на этапе выбора квартиры и повысить качество первичного контакта.

**Ожидаемый результат:**

Демо-бот, способный отвечать на вопросы о жилом комплексе, ориентируясь в его характеристиках и маркетинговых документах.

Для студентов с ограниченными возможностями здоровья предусмотрены дополнительные индивидуальные консультации, на которых преподаватель подробно разъясняет сложные аспекты дисциплины, помогает адаптировать практические задания и обеспечивает специальные условия для освоения методов работы с системами искусственного интеллекта. Индивидуальный подход позволяет таким студентам полноценно участвовать в учебном процессе и достигать требуемых результатов обучения.

**7. Материально-техническое обеспечение по дисциплине (модулю)**

Облачные ресурсы предоставляются технологическим партнером Yandex Cloud: 60 виртуальных машин, она на команду до 3-х студентов.

№	Вид работ	Наименование учебной аудитории, ее оснащенность оборудованием и техническими средствами обучения
1	Лекционные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения
2	Лабораторные занятия	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, проектором, программным обеспечением
3	Групповые (индивидуальные) консультации	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
4	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная специализированной мебелью и техническими средствами обучения, компьютерами, программным обеспечением
5	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.

Примечание: Конкретизация аудиторий и их оснащение определяется ОПОП.