
УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«25» мая 2026 г.
Протокол № 2

**Рабочая программа дисциплины (модуля)
«Академия Бэкенд»**

Направление подготовки: 02.04.01 Математика и компьютерные науки

Направленность (профиль) подготовки: Управление командами
разработки

Квалификация (степень) выпускника: магистр

Форма обучения: очная

Срок освоения программы: 2 года

Год набора: 2026

**Москва
2026**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения.....	5
3. Тематический план.....	7
4. Содержание дисциплины (модуля).....	7
5. Учебно-методическое обеспечение	9
6. Материально-техническое обеспечение	9
7. Методические и оценочные материалы	11

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Академия Бэкенд» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – магистратура по направлению 02.04.01 Математика и компьютерные науки, профиль Управление командами разработки, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 810 от 23.08.2017 года.

Изучение дисциплины (модуля) «Академия Бэкенд» позволяет студентам освоить ключевые технологии и инструменты разработки серверной части приложений, что является основой для создания масштабируемых и эффективных программных решений. Кроме того, знание принципов работы с базами данных, API и архитектурными паттернами способствует развитию навыков, необходимых для успешной карьеры в области программирования и разработки программного обеспечения.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки магистратуры по направлению 02.04.01 Математика и компьютерные науки, профиль Управление командами разработки и входит в Блок Факультативные дисциплины.

Дисциплина (модуль) изучается на 1 курсе во 2 семестре.

Цель изучения дисциплины (модуля): формирование понимания идеологии и ключевых аспектов разработки внутренней и вычислительной логики веб-сайтов или веб-приложения, а также иного программного обеспечения и информационных систем.

Задачи изучения дисциплины (модуля):

- изучение построения внутренней структуры веб-приложений;
- изучения основных методик, подходов и библиотек для построения внутренней структуры веб-приложения на языках C+ и Python;
- освоить работу с реляционными и нереляционными базами данных в составе серверного приложения;
- развить навыки применения современных фреймворков, паттернов проектирования и инструментов автоматизации в бэкенд-разработке.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- архитектурные модели веб-приложений;
- принципы работы с базами данных;
- основные концепции REST, GraphQL, WebSocket и построения API;
- структуру HTTP-запросов и ответов, методы, заголовки, коды состояния;
- паттерны проектирования, применяемые в бэкенд-разработке.

уметь:

- разрабатывать серверную логику на языках Python и C++;
- создавать и документировать API с использованием современных стандартов;
- подключать и эффективно использовать базы данных в приложениях;
- применять фреймворки;
- организовывать проект по принципам модульности, тестируемости и масштабируемости.

владеть:

- навыками отладки и тестирования бэкенд-приложений (юнит- и интеграционные тесты);
- методами безопасной обработки пользовательских данных и аутентификации (JWT, OAuth);
- инструментами логирования, мониторинга и обработки ошибок в продакшене;

- навыками работы с системами контроля версий и CI/CD в контексте бэкенд-разработки;
- практическими приемами оптимизации производительности и управления состоянием серверного приложения.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
УК-6.	Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки	УК-6.1.	Знает основные методы самооценки и анализа своей деятельности, а также принципы управления временем и целеполагания
		УК-6.2.	Умеет ставить реалистичные и достижимые цели, определять приоритеты в своей деятельности, а также разрабатывать и внедрять планы по совершенствованию своих навыков и компетенций на основе полученной самооценки
		УК-6.3.	Имеет практический опыт применения методов самооценки в своей профессиональной деятельности, включая участие в тренингах, семинарах и проектах, направленных на развитие личной эффективности и профессионального роста
ОПК-2.	Способен создавать и исследовать новые математические модели в естественных науках, совершенствовать и разрабатывать концепции, теории и методы	ОПК-2.1.	Знает современные подходы к построению математических и статистических моделей, применяемых в электронной коммерции, включая модели поведения пользователей, системы рекомендаций, модели удержания клиентов, а также методы анализа неполных, зашумлённых и временных данных
		ОПК-2.2.	Умеет разрабатывать, адаптировать и проверять математические модели для решения прикладных задач, оценивать их точность и устойчивость, интерпретировать результаты с учетом экономических показателей деятельности организации

		ОПК-2.3.	Имеет практический опыт разработки и исследования математических моделей в рамках прикладных проектов, включая подготовку данных, настройку параметров моделей, проверку их качества на реальных данных и представление результатов в аналитических отчетах
ПК-3.	Способен решать задачи профессиональной деятельности, формулировать результат, увидеть следствия полученного результата	ПК-3.1.	Знает основные принципы и методы решения задач профессиональной деятельности, а также способы формулирования и представления результатов, включая анализ последствий и их значимость в контексте проекта
		ПК-3.2.	Умеет применять математические и компьютерные методы для решения конкретных задач, формулировать четкие и обоснованные результаты, а также анализировать их последствия для дальнейших действий и решений
		ПК-3.3.	Имеет практический опыт в решении профессиональных задач, включая участие в проектах, где были получены результаты и проанализированы их следствия, что способствовало принятию обоснованных решений

3. Тематический план

№ п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Аудиторная работа		Контроль	Самостояте льная работа	
		Лекции	Семинары (практичес кие занятия)			
1	Backend разработка	2	2		7	Домашнее задание
2	Минимально жизнеспособный продукт (MVP), будущий проект	4	4		7	Домашнее задание
3	Проектирование сервиса	3	3		7	Домашнее задание
4	Хостинг. Коммуникация с сервисом. Документация API	2	2		7	Домашнее задание
5	Реализация сервиса	2	2		7	Домашнее задание
6	Рефакторинг сервиса в соответствии с общеизвестными паттернами проектирования приложений	2	2		7	Домашнее задание
	Зачет			4		Проект
Итого:		15	15	4	42	
Объем дисциплины (модуля) (в ак. ч.)		76				
Объем дисциплины (модуля) (в зач. ед.)		2				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Backend разработка	Декомпозиция процесса разработки backend включает в себя достижение поставленных целей и реализацию задач дисциплины (модуля), а также определение планов разработки проекта. Для изучения темы применяются различные формы и методы проведения занятий, включая практические занятия
2	Минимально жизнеспособный продукт (MVP), будущий проект	Формирование функциональных и технических требований к создаваемому сервису также осуществляется через практические занятия, которые используют разнообразные образовательные технологии
3	Проектирование сервиса	Процесс формирования архитектуры сервиса и схемы базы данных (БД) проходит в рамках практических занятий, что позволяет студентам лучше усвоить материал
4	Хостинг. Коммуникация с сервисом. Документация API	Развертывание сервиса на внешнем и локальном хостинге, а также предоставление доступа к автоматически генерируемой документации API осуществляется через практические занятия, применяющие современные образовательные подходы
5	Реализация сервиса	Создание проекта и организация его структуры, а также реализация функционального ядра сервиса также проводятся в формате

		практических занятий, что способствует активному обучению
6	Рефакторинг сервиса в соответствии с общеизвестными паттернами проектирования приложений	Приведение кодовой базы к формату, необходимому для командной работы, и изучение возможностей расширения и масштабирования проекта также реализуются через практические занятия, что обеспечивает глубокое понимание темы

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Кубенский, А. А. Функциональное программирование : учебник и практикум для вузов / А. А. Кубенский. — Москва : Издательство Юрайт, 2025. — 348 с. — (Высшее образование). — ISBN 978-5-9916-9242-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561074>.

2. Тузовский, А. Ф. Проектирование и разработка web-приложений : учебник для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2025. — 219 с. — (Высшее образование). — ISBN 978-5-534-16300-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561176>.

Дополнительная литература:

1. Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 285 с. — (Высшее образование). — ISBN 978-5-534-16031-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/560815>

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- механическими калькуляторами;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		

Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Академия Бэкенд» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, проект, домашние задания, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар – это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Проект – исследовательская работа по дисциплине (модулю) и презентация результатов.

Для успешной подготовки к проекту: четко определите цели и задачи проекта, распределите роли и обязанности между участниками, а также установите сроки выполнения каждой части работы. Регулярно проводите встречи для обсуждения прогресса и решения возникающих вопросов.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Академия Бэкенд»

Оценивание уровня учебных достижений обучающихся по дисциплине (модулю) осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме *зачета*, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину (модуль). Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
9	Отлично	Зачтено	
8	Отлично	Зачтено	
7	Хорошо	Зачтено	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать,
6	Хорошо	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
5	Удовлетворительно	Зачтено	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Академия Бэкенд» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашнее задание	40%	3	Набор задач по темам недели
Зачет	60%	1	Проект –исследовательская работа по дисциплине (модулю) и презентация результатов

Формула расчёта итоговой оценки по дисциплине (модулю) «Академия Бэкенд»:
« $0,4 \times \text{Домашнее задание} + 0,6 \times \text{Зачет}$ ».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание. Формирование архитектуры сервиса и схемы базы данных:

1. Проанализировать требования к сервису и выделить основные компоненты архитектуры.
2. Нарисовать диаграмму архитектуры сервиса с описанием каждого компонента.
3. Спроектировать схему базы данных, определив таблицы, поля и связи между ними.

4. Обосновать выбор типа базы данных (реляционная, NoSQL и т.д.) для данного сервиса.
5. Подготовить краткий отчет с описанием архитектуры и схемы БД.

Домашнее задание. Развертывание сервиса и предоставление доступа к документации API:

1. Настроить локальный хостинг и развернуть на нем тестовый сервис.
2. Развернуть сервис на внешнем хостинге (например, Heroku, AWS, DigitalOcean).
3. Настроить автоматическую генерацию документации API (Swagger, Redoc или аналог).
4. Предоставить ссылку на развернутый сервис и документацию API.
5. Описать процесс развертывания и возникавшие сложности в письменном виде.

Домашнее задание. Создание проекта, организация структуры и реализация функционального ядра сервиса:

1. Создать новый проект с правильной организацией каталогов и файлов.
2. Реализовать основные функции сервиса (например, регистрацию пользователя, обработку запросов).
3. Написать модульные тесты для ключевых функций.
4. Описать структуру проекта и логику реализации функционального ядра.
5. Подготовить README файл с инструкциями по запуску и использованию сервиса.

Примерное описание проекта

Проект: Приведение кодовой базы к формату, необходимому для командной работы, и изучение возможностей расширения и масштабирования проекта

Основные этапы реализации проекта:

1. Анализ существующей кодовой базы

- Изучить текущую структуру проекта и качество кода.
- Определить проблемные места, препятствующие командной работе и масштабированию.

2. Рефакторинг и стандартизация кода

- Внедрить единые стандарты кодирования (например, PEP8 для Python, ESLint для JavaScript).
- Разделить код на модули и компоненты с понятными интерфейсами.
- Добавить комментарии и документацию к ключевым частям кода.

3. Организация системы контроля версий

- Настроить репозиторий (Git).
- Определить правила ветвления и слияния (Git Flow или аналог).
- Настроить шаблоны для коммитов и пулл-реквестов.

4. Автоматизация сборки и тестирования

- Настроить систему автоматического запуска тестов (CI/CD).
- Реализовать модульные и интеграционные тесты.
- Организовать автоматическую проверку стиля кода и статический анализ.

5. Планирование расширения и масштабирования

- Разработать архитектурные решения, поддерживающие масштабирование (микросервисы, кэширование, балансировка нагрузки).
- Подготовить документацию по расширению функционала.

- Продemonстрировать пример добавления нового модуля или компонента.

Критерии оценивания:

Критерий	Максимальный балл	Описание
Качество рефакторинга кода	10	Чистота, читаемость, соответствие стандартам кодирования
Организация структуры проекта	10	Логичная модульная структура, понятные интерфейсы
Настройка системы контроля версий	10	Правильное использование Git, правила ветвления, информативные коммиты
Автоматизация тестирования и сборки	10	Наличие тестов, их покрытие, корректная работа CI/CD
Документация и комментарии	10	Полнота и качество документации, понятность комментариев
Планирование масштабирования	10	Обоснованность архитектурных решений, готовность к расширению
Демонстрация расширяемости	10	Успешное добавление нового функционала или модуля без нарушения существующего кода

Рекомендации по выполнению:

- Используйте современные инструменты и практики командной разработки.
- Обязательно проводите код-ревью внутри команды.
- Документируйте все изменения и решения.
- Планируйте работу с учётом возможного роста проекта.
- Активно используйте систему контроля версий для отслеживания прогресса.

Итоговый результат:

- Репозиторий с приведённой в порядок кодовой базой.
- Настроенный CI/CD процесс.
- Документация по структуре проекта и правилам командной работы.
- Демонстрация расширения проекта новым функционалом.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Какой первый шаг при декомпозиции процесса backend-разработки? а) Написание кода б) Определение основных функций сервиса в) Тестирование г) Развёртывание на хостинге	б	УК-6
2.	Как называется документ, в котором фиксируются цели и задачи проекта?	План разработки /	УК-6

		техническое задание	
3.	Какая форма занятий наиболее эффективна для освоения практических навыков backend-разработки?	Практическое занятие	УК-6
4.	Что из перечисленного относится к функциональным требованиям? а) Скорость отклика сервиса б) Возможность регистрации пользователей в) Используемый язык программирования г) Архитектура базы данных	б	ОПК-2
5.	Как называется процесс описания и документирования требований к системе?	Спецификация требований	ОПК-2
6.	Что является основным элементом реляционной базы данных? а) Таблица б) Файл в) Класс г) Модуль	а	ПК-3
7.	Как называется диаграмма, отображающая структуру базы данных?	ER-диаграмма / диаграмма сущность-связь	ПК-3
8.	Как называется подход, при котором сервис развёртывается на внешнем или локальном сервере?	Развёртывание / деплоймент	ПК-3
9.	Какой из перечисленных языков чаще всего используется для backend-разработки? а) HTML б) JavaScript (Node.js) в) CSS г) Photoshop	б	УК-6
10.	Какой протокол используется для обмена данными между клиентом и сервером в REST API? а) FTP б) HTTP/HTTPS в) SMTP г) TCP	б	ПК-3
11.	Что такое API? а) Графический интерфейс пользователя б) Интерфейс программирования приложений в) Антивирусная программа г) Автоматизированная платформа интерфейсов	б	ОПК-2
12.	Какой метод HTTP-запроса используется для создания нового ресурса на сервере? а) GET б) POST в) PUT г) DELETE	б	ПК-3
13.	Как называется архитектурный подход, при котором backend разделён на независимые сервисы?	Микросервисы / микросервисная архитектура	ОПК-2

14.	Какой инструмент используется для управления версиями кода? а) Figma б) Git в) Excel г) Notion	б	УК-6
15.	Какой тип базы данных используется для хранения иерархичных или неструктурированных данных (например, JSON)? а) Реляционная б) NoSQL в) Файловая г) Оперативная	б	ПК-3
16.	Что такое middleware в backend-разработке?	Программный слой, обрабатывающий запросы между клиентом и сервером	ОПК-2
17.	Какой из перечисленных этапов следует после написания кода? а) Документирование требований б) Тестирование в) Планирование интерфейса г) Сбор обратной связи от пользователей	б	УК-6
18.	Как называется процесс проверки корректности работы кода?	Тестирование / unit-тесты	ПК-3
19.	Какой формат данных чаще всего используется для обмена информацией в REST API?	JSON	ОПК-2
20.	Как называется среда, в которой backend-приложение запускается для тестирования перед продакшеном?	Тестовая среда / staging	ПК-3