

**Компонент ОПОП 09.03.01 Информатика и вычислительная техника,
направленность (профиль) «Технологии разработки программного обеспечения»**
наименование ОПОП

Б1.В.ДВ.02.01
шифр дисциплины

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Дисциплины
(модуля) **Системы управления контентом и headless-архитектура**

Разработчик (и):
Ляш О.И.
ФИО
зав.кафедрой
должность

канд.пед.наук,
доцент
ученая степень,
звание

Утверждено на заседании кафедры
информационных технологий
наименование кафедры

протокол № 6 от 24.02.2026

Заведующий кафедрой ИТ

 Ляш О.И.
подпись ФИО

1. Критерии и средства оценивания компетенций и индикаторов их достижения, формируемых дисциплиной (модулем)

Код и наименование компетенции	Код и наименование индикатора(ов) достижения компетенции	Результаты обучения по дисциплине (модулю)			Оценочные средства текущего контроля	Оценочные средства промежуточной аттестации
		<i>Знать</i>	<i>Уметь</i>	<i>Владеть</i>		
ПК-1 Способен разрабатывать технические спецификации на программные компоненты и их взаимодействие	ИД-1_{ПК-1} Анализирует существующую документацию (техническое задание, спецификации) и требования к программному обеспечению для выделения перечня подлежащих разработке программных компонентов. ИД-2_{ПК-1} Разрабатывает, согласовывает и утверждает технические спецификации на программные компоненты, описывающие их структуру, функции, логику работы и атрибуты качества. ИД-3_{ПК-1} Проектирует интерфейсы взаимодействия программных компонентов (API) и протоколы обмена данными между ними, фиксируя результаты в спецификации. ИД-4_{ПК-1} Оформляет техническую документацию на компоненты и их взаимодействие в соответствии с требованиями стандартов.	– основные понятия, типы и архитектуры систем управления контентом (CMS); – принципы работы и особенности популярных open-source CMS; – технологии виртуализации и контейнеризации для развертывания веб-сервисов; – методы проектирования и документирования API для headless-архитектур; – требования к серверной инфраструктуре	– анализировать требования предприятия и выбирать оптимальную CMS для решения профессиональных задач; – настраивать виртуальные машины и контейнеризованные среды для развертывания систем управления контентом; – разворачивать open-source CMS на серверной инфраструктуре с использованием средств автоматизации; – проектировать и настраивать API для интеграции headless-CMS с фронтенд-приложениями и	– навыками развертывания и администрирования open-source систем управления контентом в корпоративной инфраструктуре; – навыками настройки виртуальных машин, контейнеров и оркестрации для обеспечения отказоустойчивости веб-сервисов; – навыками проектирования headless-архитектур и интеграции компонентов через стандартизированные интерфейсы; – навыками оценки рисков, обеспечения информационной безопасности и масштабирования CMS-решений; – навыками составления проектной и	– комплект заданий для выполнения лабораторных (практических) работ; – тестовые задания;	Результаты текущего контроля

ПК-2 Способен проектировать компьютерное программное обеспечение	ИД-1_{ПК-2} Выбирает архитектуру программного обеспечения и декомпозирует его на модули и компоненты. ИД-2_{ПК-2} Применяет объектно-ориентированные или иные методологии для разработки моделей предметной области, структур данных и бизнес-логики будущего ПО. ИД-3_{ПК-2} Разрабатывает логическую и физическую схемы баз данных, а также осуществляет проектирование пользовательских интерфейсов с учетом требований эргономики. ИД-4_{ПК-2} Создает проектную документацию на программное обеспечение, позволяющую осуществить кодирование и дальнейшее развертывание продукта. ИД-5_{ПК-2} Оценивает соответствие разработанного проекта исходным требованиям и техническому заданию, а также анализирует возможные риски реализации.	и безопасности при внедрении CMS в корпоративной среде; – стандарты оформления технической документации на разворачиваемые программные компоненты.	внешними сервисами; – выполнять миграцию контента, конфигурацию прав доступа и базовую оптимизацию производительности; – документировать процесс развертывания, настройки и эксплуатации программного обеспечения. сервисов.	эксплуатационной документации в соответствии с требованиями стандартов.		
---	--	---	---	--	--	--

2. Оценка уровня сформированности компетенций (индикаторов их достижения)

Показатели оценивания компетенций (индикаторов их достижения)	Шкала и критерии оценки уровня сформированности компетенций (индикаторов их достижения)			
	Ниже порогового («неудовлетворительно»)	Пороговый («удовлетворительно»)	Продвинутый («хорошо»)	Высокий («отлично»)
Полнота знаний	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущены не грубые ошибки.	Уровень знаний в объёме, соответствующем программе подготовки. Допущены некоторые погрешности.	Уровень знаний в объёме, соответствующем программе подготовки.
Наличие умений	При выполнении стандартных заданий не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Выполнены типовые задания с не грубыми ошибками. Выполнены все задания, но не в полном объеме (отсутствуют пояснения, неполные выводы)	Продemonстрированы все основные умения. Выполнены все основные задания с некоторыми погрешностями. Выполнены все задания в полном объёме, но некоторые с недочетами.	Продemonстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Задания выполнены в полном объеме без недочетов.
Наличие навыков (владение опытом)	При выполнении стандартных заданий не продемонстрированы базовые навыки. Имели место грубые ошибки.	Имеется минимальный набор навыков для выполнения стандартных заданий с некоторыми недочетами.	Продemonстрированы базовые навыки при выполнении стандартных заданий с некоторыми недочетами.	Продemonстрированы все основные умения. Выполнены все основные и дополнительные задания без ошибок и погрешностей. Продemonстрирован творческий подход к решению нестандартных задач.
Характеристика сформированности компетенции	Компетенции фактически не сформированы. Имеющихся знаний, умений, навыков недостаточно для решения практических (профессиональных) задач. ИЛИ Зачетное количество баллов не набрано согласно установленному диапазону	Сформированность компетенций соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в целом достаточно для решения практических (профессиональных) задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону	Сформированность компетенций в целом соответствует требованиям. Имеющихся знаний, умений, навыков достаточно для решения стандартных профессиональных задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону	Сформированность компетенций полностью соответствует требованиям. Имеющихся знаний, умений, навыков в полной мере достаточно для решения сложных, в том числе нестандартных, профессиональных задач. ИЛИ Набрано зачетное количество баллов согласно установленному диапазону

3. Критерии и шкала оценивания заданий текущего контроля

За выполнение всех видов работ на курсе можно получить 100 баллов. При выполнении работ студент получает первичный балл (ПБ) в диапазоне от 0 до 1 в соответствии с приведенными ниже таблицами. Все баллы, полученные студентом за текущий контроль и промежуточную аттестацию суммируются и пропорционально преобразуются к итоговому значению не превышающему величину 100 баллов.

Для «зачёта» и «зачёта с оценкой» за текущий контроль можно получить максимально 100 баллов.

Для «экзамена» за текущий контроль можно получить максимально 80 баллов, а за промежуточную аттестацию можно получить максимально «20 баллов».

3.1 Критерии и шкала оценивания лабораторных работ

Перечень лабораторных, описание порядка выполнения и защиты работы, требования к результатам работы, структуре и содержанию отчета и т.п. представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ.

Оценка	Баллы	Критерии оценивания
Отлично	0.9-1	Задание выполнено полностью и правильно. Отчет по лабораторной/практической работе подготовлен качественно в соответствии с требованиями. Полнота ответов на вопросы преподавателя при защите работы.
Хорошо	0.89	Задание выполнено полностью, но нет достаточного обоснования или при верном решении допущена незначительная ошибка, не влияющая на правильную последовательность рассуждений. Все требования, предъявляемые к работе, выполнены.
Удовлетворительно	0.6-0.79	Задания выполнены частично с ошибками. Демонстрирует средний уровень выполнения задания на лабораторную/практическую работу. Большинство требований, предъявляемых к заданию, выполнены.
Неудовлетворительно	0-0.59	Задание выполнено со значительным количеством ошибок на низком уровне. Многие требования, предъявляемые к заданию, не выполнены. ИЛИ Задание не выполнено.

3.2. Критерии и шкала оценивания тестирования

Перечень тестовых вопросов и заданий, описание процедуры тестирования представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ.

В ФОС включен типовой вариант тестового задания:

При проектировании API взаимодействия между клиентской частью web-VR-приложения и бэкенд-сервером наиболее подходящим протоколом для передачи данных в реальном времени (позиции игроков, события) является:

- A) HTTP/1.1
- B) WebSocket
- C) FTP
- D) SMTP

Оценка	Баллы	Критерии оценки
--------	-------	-----------------

<i>Тест зачтён</i>	0.6-1	60-100 % правильных ответов
<i>Тест не зачтён</i>	0-0.59	59 % и менее правильных ответов

3.3. Критерии и шкала оценивания расчетно-графической работы

Перечень заданий для расчетно-графической работы, рекомендации по выполнению представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ.

В ФОС включен типовой вариант расчетно-графической работы:

Варианты предметных областей:

1. Корпоративный портал на базе Headless-CMS.
2. Образовательная платформа.
3. Медиа-платформа для издательского дома.
4. E-commerce витрина на базе Headless-CMS.
5. Корпоративная база знаний.

Состав и содержание расчетно-графической работы

Работа включает пояснительную записку и графическую часть (схемы, диаграммы, макеты сцен).

1. Анализ требований и выделение компонентов.
 - Описание функциональных и нефункциональных требований к системе (доступность, производительность, масштабируемость, безопасность).
 - Выделение перечня программных компонентов: CMS-ядро, API-шлюз, фронтенд-клиент, СУБД, система кэширования, служба резервного копирования, виртуальная инфраструктура.
 - Обоснование выбора open-source CMS и стека виртуализации/контейнеризации.
2. Разработка технических спецификаций на компоненты и их взаимодействие.
 - Для каждого компонента описать: структуру, функции, логику работы, атрибуты качества.
 - Спроектировать интерфейсы взаимодействия компонентов: REST API или GraphQL (указать методы, маршруты, форматы запросов/ответов, коды состояния, аутентификацию).
 - Описать протоколы обмена данными (HTTP/HTTPS, WebSocket, JSON, gRPC). Зафиксировать спецификацию в формате OpenAPI (Swagger). Предоставить фрагмент спецификации (YAML/JSON).
3. Проектирование архитектуры приложения
 - Выбрать архитектурный стиль (монолитный CMS, headless-разделение, микросервисный). Обосновать выбор.
 - Представить диаграмму компонентов и диаграмму развёртывания (UML).
 - Выполнить декомпозицию системы на модули. Описать взаимодействие между CMS, API-слоем, фронтендом и инфраструктурой (BM, Docker, сеть).
4. Проектирование базы данных.
 - Разработать логическую схему данных (ER-диаграмма или схема контента/коллекций).
 - Разработать физическую схему: типы данных, индексы, ограничения целостности, стратегии хранения медиафайлов.
 - Привести примеры запросов (SQL или к NoSQL/ORM) для ключевых операций (создание контента, фильтрация, пагинация, обновление статусов).
5. Проектирование интерфейсов и сценариев взаимодействия.

- Описать API как контракт для внешних клиентов (веб-приложение, мобильное приложение, IoT-устройства).
 - Привести примеры запросов и ответов (POST, GET, PUT/PATCH, DELETE). При необходимости – описать формат WebSocket-сообщений (realtime-уведомления).
 - Схематично изобразить основные сценарии взаимодействия (последовательность запросов при публикации, редактировании и выдаче контента).
6. Проектная документация и оценка рисков.
- Создать фрагмент проектной документации, достаточный для развёртывания (инструкция по подготовке VM, переменные окружения, docker-compose конфигурация, настройка SSL/TLS).
 - Оценить соответствие спроектированного решения исходным требованиям (таблица соответствия).
 - Провести анализ возможных рисков реализации: безопасность (уязвимости CMS, XSS/SQLi, несанкционированный доступ к API), производительность (event loop, запросы к БД, кэширование), масштабируемость, отказоустойчивость, риски при обновлениях open-source. Предложить меры по минимизации рисков.

Графическая часть (обязательные схемы)

1. Диаграмма компонентов (UML Component Diagram) – отражает CMS-бэкенд, API-слой, фронтенд-клиент, СУБД, кэш, прокси-сервер.
2. Диаграмма развёртывания (UML Deployment Diagram) – показывает узлы (виртуальные машины/контейнеры), сеть, балансировщик, хранилище артефактов.
3. ER-диаграмма/Схема данных – логическая структура таблиц или контентных типов, связи, ключевые индексы.
4. Схема взаимодействия компонентов при выполнении одного ключевого сценария – диаграмма последовательности (UML Sequence Diagram), например: «Аутентификация администратора → Создание контента → Публикация → Фронтенд запрашивает данные через API → Кэширование ответа».

Оценка	Баллы	Критерии оценивания
Отлично	0.9-1	Работа выполнена полностью, без ошибок (возможна одна неточность, описка, не являющаяся следствием непонимания материала).
Хорошо	0.8-0.89	Работа выполнена полностью, но обоснования шагов решения недостаточны, допущена одна негрубая ошибка или два-три недочета, не влияющих на правильную последовательность рассуждений.
Удовлетворительно	0.6-0.79	В работе допущено более одной грубой ошибки или более двух-трех недочетов, но обучающийся владеет обязательными умениями по проверяемой теме.
Неудовлетворительно	0-0.59	В работе есть грубые ошибки и недочеты ИЛИ Контрольная работа не выполнена.

3.3. Критерии и шкала оценивания реферата

Тематика рефератов по дисциплине (модулю), требования к структуре, содержанию и оформлению изложены в методических материалах по освоению дисциплины (модуля), представлены в электронном курсе в ЭИОС МАУ.

В ФОС включены примерные темы рефератов:

1. Эволюция систем управления контентом: от монолитных решений к headless-архитектурам

2. Сравнительный анализ традиционных, decoupled и headless CMS: преимущества, ограничения и сценарии применения
3. Архитектурные принципы headless-систем: API-first, JAMstack, MACH
4. Концепция Content-as-a-Service (CaaS) и её влияние на проектирование корпоративных порталов
5. Обзор современных open-source headless CMS: Strapi, Directus, Ghost, payload
6. Особенности проектирования контентных моделей, таксономий и метаданных в headless-среде
7. Сравнение WordPress в headless-режиме и нативных headless-платформ для enterprise-задач
8. Drupal как платформа для сложных информационных порталов: архитектура и возможности кастомизации
9. Интеграция Moodle и других LMS в корпоративную headless-экосистему: сценарии и ограничения
10. Роль микросервисных подходов в современных CMS-решениях: декомпозиция и взаимодействие компонентов
11. Развёртывание CMS в виртуальных машинах: выбор гипервизора, настройка ОС, веб-сервера и СУБД
12. Контейнеризация headless-CMS с использованием Docker и Docker Compose: практика и преимущества
13. Автоматизация развёртывания CMS: настройка CI/CD-пайплайнов для корпоративных сред
14. Настройка обратного прокси-сервера (Nginx/Traefik) для маршрутизации и балансировки headless-архитектур
15. Управление мультисерверными CMS-инфраструктурами: базовые принципы оркестрации
16. Проектирование REST API для headless CMS: стандарты, версионирование, пагинация и документирование
17. GraphQL в headless-экосистемах: сравнение с REST, настройка эндпоинтов и оптимизация запросов
18. Интеграция headless CMS с внешними сервисами: вебхуки, очереди сообщений и синхронизация данных
19. Настройка единой аутентификации и авторизации (SSO, JWT, LDAP/Active Directory) для корпоративных CMS
20. Ролевая модель и управление доступом в headless-системах: проектирование политик безопасности
21. Стратегии резервного копирования, миграции и восстановления контента в распределённых CMS
22. Мониторинг производительности headless-архитектур: метрики, централизованное логирование, трассировка запросов
23. Защита headless CMS от типовых веб-угроз: WAF, rate limiting, защита API-эндпоинтов
24. Оптимизация доставки контента: настройка CDN, кэширование, edge-вычисления и статическая генерация
25. Документирование архитектуры и процедур развёртывания CMS в соответствии со стандартами ЕСПД
26. Применение OpenAPI/Swagger для спецификации и согласования интерфейсов headless-систем
27. Headless-архитектуры в образовательных учреждениях: развёртывание портала, интеграция с LMS и системами дистанционного обучения
28. Корпоративные сценарии использования headless CMS: intranet-порталы, e-commerce витрины, медиа-платформы

29. Оценка рисков при внедрении headless-решений: производительность, масштабируемость, зависимость от сторонних сервисов
30. Тренды развития систем управления контентом: AI-ассистенты для авторов, headless-коммерция, low-code/no-code администрирование

Оценка	баллы	Критерии оценки
Отлично	0.9-1	Выполнены все требования к написанию и защите реферата: обозначена проблема и обоснована ее актуальность, сделан краткий анализ различных точек зрения на рассматриваемую проблему и логично изложена собственная позиция, сформулированы выводы, тема раскрыта полностью, выдержан объем, соблюдены требования к внешнему оформлению, даны правильные ответы на дополнительные вопросы.
Хорошо	0.8-0.89	Основные требования к реферату и его защите - выполнены, но при этом допущены недочеты. В частности, имеются неточности в изложении материала; отсутствует логическая последовательность в суждениях; не выдержан объем реферата; имеются упущения в оформлении; на дополнительные вопросы при защите даны неполные ответы.
Удовлетворительно	0.6-0.79	Имеются существенные отступления от требований к реферированию. В частности, тема освещена лишь частично; допущены фактические ошибки в содержании реферата или при ответе на дополнительные вопросы; во время защиты отсутствует вывод.
Неудовлетворительно	0-0.59	Тема реферата не раскрыта, обнаруживается существенное непонимание проблемы.

3.4. Критерии и шкала оценивания эссе

Тематика эссе по дисциплине (модулю), требования к структуре, содержанию и оформлению представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ.

В ФОС включены примерные темы эссе:

1. Эволюция систем управления контентом: от монолитных CMS к headless-архитектуре.
2. Практическое развертывание open-source headless CMS в корпоративной инфраструктуре на базе виртуальных машин.
3. Проектирование API-интерфейсов для headless CMS: сравнительный анализ REST и GraphQL подходов.
4. Архитектурные преимущества и ограничения headless-решений для многоканальных цифровых продуктов.
5. Перспективы развития систем управления контентом: влияние ИИ и компонуемых DXR-платформ на рынок веб-технологий.

Оценка	баллы	Критерии оценки
Отлично	0.9-1	Представлена собственная точка зрения (позиция, отношение) при раскрытии проблемы. Проблема раскрыта на теоретическом уровне, в связях и с обоснованиями, с корректным использованием обществоведческих терминов и понятий в контексте ответа. Предоставлена аргументация своего мнения с опорой на факты общественной жизни или личный социальный опыт.
Хорошо	0.8-0.89	Представлена собственная точка зрения (позиция, отношение) при раскрытии проблемы. Проблема раскрыта с корректным использованием терминов и понятий в контексте ответа (теоретические связи и обоснования не присутствуют или явно не прослеживаются). Представлена аргументация своего мнения с опорой на факты общественной жизни или личный социальный опыт.

Удовлетворительно	0.6-0.79	Представлена собственная точка зрения (позиция, отношение) при раскрытии проблемы; проблема раскрыта при формальном использовании обществоведческих терминов. Представлена аргументация своего мнения с опорой на факты общественной жизни или личный социальный опыт без теоретического обоснования.
Неудовлетворительно	0-0.59	Представлена собственная точка зрения (позиция, отношение) при раскрытии проблемы, но проблема раскрыта не полностью. Аргументация своего мнения слабо связана с раскрытием проблемы.

3.5. Критерии и шкала оценивания мультимедийной презентации

Требования к структуре, содержанию и оформлению представлены в методических материалах по освоению дисциплины (модуля) и в электронном курсе в ЭИОС МАУ.

Оценка	баллы	Критерии оценки
Отлично	0.9-1	Презентация соответствует теме самостоятельной работы. Оформлен титульный слайд с заголовком. Сформулированная тема ясно изложена и структурирована, использованы графические изображения (фотографии, картинки и т.п.), соответствующие теме, выдержан стиль, цветовая гамма, использована анимация, звук. Логично изложена собственная позиция, сформулированы выводы, тема раскрыта полностью, выдержан объём, соблюдены требования к внешнему оформлению. Работа оформлена и предоставлена в установленный срок.
Хорошо	0.8-0.89	Презентация соответствует теме самостоятельной работы. Имеются неточности в изложении материала. Отсутствует логическая последовательность в суждениях. Не выдержан объем презентации, имеются упущения в оформлении. На дополнительные вопросы при защите даны неполные ответы. Работа оформлена и предоставлена в установленный срок.
Удовлетворительно	0.6-0.79	Презентация соответствует теме самостоятельной работы. Сформулированная тема изложена и структурирована не в полном объеме. Не использованы графические изображения (фотографии, картинки и т.п.), соответствующие теме. Присутствуют существенные отступления от требований к составлению презентации. Допущены фактические ошибки в содержании или при ответе на дополнительные вопросы.
Неудовлетворительно	0-0.59	Работа не выполнена или не соответствует теме самостоятельной работы.

3.6. Критерии и шкала оценивания посещаемости занятий

Посещение занятий обучающимися определяется в процентном соотношении

Баллы	Критерии оценки
0.75-1	посещаемость 75 - 100 %
0.5-0.74	посещаемость 50 - 74 %
0-0.49	посещаемость менее 50 %

3.7. Критерии и шкала оценивания своевременной сдачи контрольных точек

Своевременность сдачи контрольных точек обучающимися определяется в процентном соотношении

Баллы	Критерии оценки
0.75-1	Своевременность сдачи 75 - 100 %
0.6-0.74	Своевременность сдачи 50 - 74 %
0-0.5	Своевременность сдачи 50 %

4. Критерии и шкала оценивания результатов обучения по дисциплине (модулю) при проведении промежуточной аттестации (зачёт с оценкой)

Критерии и шкала оценивания результатов освоения дисциплины (модуля) с зачетом с оценкой

Если обучающийся набрал зачетное количество баллов согласно установленному диапазону по дисциплине (модулю), то он считается аттестованным с оценкой согласно шкале баллов для определения итоговой оценки:

Оценка	Баллы	Критерии оценивания
<i>Отлично</i>	91 - 100	Набрано зачетное количество баллов согласно установленному диапазону
<i>Хорошо</i>	81 - 90	Набрано зачетное количество баллов согласно установленному диапазону
<i>Удовлетворительно</i>	60 - 80	Набрано зачетное количество баллов согласно установленному диапазону
<i>Неудовлетворительно</i>	менее 60	Зачетное количество согласно установленному диапазону баллов не набрано

5. Задания диагностической работы для оценки результатов обучения по дисциплине (модулю) в рамках внутренней и внешней независимой оценки качества образования

ФОС содержит задания для оценивания знаний, умений и навыков, демонстрирующих уровень сформированности компетенций и индикаторов их достижения в процессе освоения дисциплины (модуля).

Комплект заданий разработан таким образом, чтобы осуществить процедуру оценки каждой компетенции, формируемых дисциплиной (модулем), у обучающегося в письменной форме.

Содержание комплекта заданий включает: *тестовые задания*.

Комплект заданий диагностической работы

ПК-1 Способен разрабатывать технические спецификации на программные компоненты и их взаимодействие	
1.	При анализе технического задания на внедрение headless-CMS для корпоративного портала первым этапом необходимо выделить перечень: А) Цветовых палитр для фронтенда В) Программных и инфраструктурных компонентов (СУБД, веб-сервер, ядро CMS, API-шлюз, контейнеризатор) С) Имен пользователей социальных сетей компании D) Версий браузеров сотрудников
2.	В технической спецификации на компонент «Сервис доставки контента» headless-архитектуры атрибут качества «производительность» должен быть зафиксирован через: А) Максимально допустимое время отклика API и целевую нагрузку (RPS) В) Версию используемой базы данных С) Количество администраторов системы D) Выбранную тему оформления административной панели
3.	Какой формат спецификации является отраслевым стандартом для описания REST/GraphQL API headless-CMS и позволяет автоматически генерировать интерактивную документацию? А) WSDL В) OpenAPI (Swagger)

	C) SQL DDL D) Чистый Markdown без схем
4.	При проектировании взаимодействия headless-CMS с внешним фронтендом в спецификации необходимо зафиксировать: A) Стратегии инвалидации кэша, TTL, форматы JSON-схем запросов/ответов и коды состояний HTTP B) Только IP-адрес сервера C) Внутреннюю структуру оперативной памяти ВМ D) Названия пртм-пакетов для подключения
5.	Для обеспечения безопасного взаимодействия между headless-CMS и мобильным клиентом в технической спецификации необходимо предусмотреть: A) Протокол FTP для передачи данных B) Механизмы аутентификации (JWT/OAuth), валидацию входных данных и шифрование трафика (HTTPS/TLS) C) Открытые порты без фильтрации D) Синхронный обмен данными через WebSocket без авторизации
6.	В спецификации на программный компонент «Модуль ролевой модели» headless-CMS обязательно описывается: A) Архитектура процессора сервера B) Иерархия ролей, права доступа к типам контента и логика проверки разрешений C) Алгоритм сжатия графических изображений D) Версия операционной системы хоста
7.	При согласовании спецификации на компонент «Кэш на Redis» для headless-стека необходимо указать: A) Только IP-адрес сервера Redis B) Стратегии инвалидации кэша, время жизни ключей (TTL) и протокол обмена C) Внутреннее устройство Redis D) Сертификат SSL для веб-сервера
8.	Какой стандарт оформления технической документации на компоненты и их взаимодействие чаще всего регламентируется в отечественных ИТ-проектах? A) IEEE 829 B) ГОСТ 19 (ЕСПД) C) ISO 25010 D) W3C Recommendation
9.	В технической спецификации на взаимодействие компонентов через GraphQL атрибут качества «гибкость запросов» реализуется за счёт: A) Жёсткой привязки к конкретным эндпоинтам B) Возможности клиента запрашивать только необходимые поля и вложенные объекты в одном запросе C) Использования протокола UDP D) Отказа от типизации данных
10.	Какой раздел проектной документации на headless-CMS соответствует требованиям стандартов по фиксации интерфейсов взаимодействия? A) Перечень лицензий ПО B) Спецификация API с сигнатурами методов, схемами данных, примерами запросов/ответов и правилами обработки ошибок C) Инструкция по установке ОС на сервер D) Бюджет проекта и сроки окупаемости
ПК-2 Способен проектировать компьютерное программное обеспечение	
11.	При выборе архитектуры для корпоративного портала, требующего независимого масштабирования фронтенда и бэкенда, наиболее целесообразным решением является: A) Классическая монолитная CMS (например, WordPress с темой) B) Headless-архитектура с разделением CMS-бэкенда и клиентских приложений C) Настольное клиент-серверное приложение D) Статический сайт без системы управления
12.	Для моделирования предметной области headless-CMS (типы контента, связи, метаданные) применяется: A) Спецификация контентных моделей и ER-диаграммы B) Написание чистого SQL-кода без предварительного проектирования C) Выбор случайного шаблона темы оформления D) Копирование структуры из существующих Excel-таблиц

13.	При проектировании логической схемы базы данных для headless-CMS, где контент имеет сложную иерархию и вложенные отношения, предпочтительнее использовать: А) Реляционную СУБД (PostgreSQL/MySQL) с таблицами связей и внешними ключами В) Файловую систему в формате CSV С) Одностраничную таблицу без индексов Д) Ключ-значение хранилище без поддержки транзакций
14.	Проектирование пользовательского интерфейса (административной панели) headless-CMS должно учитывать требования эргономики, что подразумевает: А) Размещение всех настроек на одной странице без прокрутки В) Интуитивную навигацию, логическую группировку полей контента и поддержку стандартов доступности С) Использование исключительно командной строки для ввода данных Д) Отсутствие визуального разделения блоков контента
15.	Проектная документация на развёртывание headless-CMS в виртуальной среде должна обязательно содержать: А) Только лицензионное соглашение В) Инструкцию по настройке ВМ/контейнеров, параметры окружения (.env), схему сети, порядок запуска сервисов и скрипты инициализации С) Исходный код фронтенд-приложения Д) Рекламные материалы и медиа-кит
16.	Оценка соответствия спроектированной headless-архитектуры техническому заданию предполагает: А) Составление матрицы трассировки требований к реализованным компонентам и проверку SLA В) Измерение веса серверного оборудования С) Подсчёт количества строк кода в ядре CMS Д) Опрос пользователей об их предпочтениях в цветовой гамме
17.	При анализе риска «единая точка отказа» (SPOF) в развёрнутой headless-архитектуре на ВМ проектировщик должен предложить: А) Отказ от использования серверов в пользу P2P-сети В) Настройку кластеризации, балансировку нагрузки и репликацию базы данных С) Увеличение мощности одного сервера в 10 раз без резервирования Д) Ручное перезапуск сервисов при падении
18.	Для декомпозиции headless-CMS на модули и компоненты при проектировании архитектуры применяется подход: А) Разделение по функциональным доменам (контент, медиа, аутентификация, API-шлюз, кэширование) В) Объединение всего кода в один монолитный файл без разбиения на слои С) Использование только сторонних плагинов без собственной архитектурной схемы Д) Копирование кода между сервисами для ускорения разработки
19.	Какая диаграмма UML наиболее точно документирует физическое развёртывание headless-CMS в контейнерной среде (Docker) на виртуальных машинах? А) Диаграмма классов В) Диаграмма развёртывания (deployment diagram) С) Диаграмма состояний Д) Диаграмма вариантов использования
20.	Анализ риска безопасности при внедрении headless-CMS с открытым API требует обязательного проектирования: А) Механизмов rate limiting, CORS-политик, валидации входных данных и JWT-аутентификации В) Открытого доступа ко всем эндпоинтам без ограничений С) Хранения паролей в открытом виде в конфигурационных файлах Д) Отказа от использования HTTPS в локальной сети

Ключ к тесту: 1-В; 2-А; 3-В; 4-А; 5-В; 6-В; 7-В; 8-В; 9-В; 10-В; 11-В; 12-А; 13-А; 14-В; 15-В; 16-А; 17-В; 18-А; 19-В; 20-А.