

Кафедра _____ прикладной информатики и высшей математики

Б1.В.ДВ.02.02

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА)

Учебная дисциплина	Разработка облачных приложений и сервисов
По направлению подготовки	09.03.03
	«Прикладная информатика»
Профиль (программа бакалавриата)	«Дизайн и разработка цифровых программных продуктов»
Форма обучения	Очная

Оценочные материалы (средства) дисциплины рассмотрены (актуализированы) и утверждены на заседании кафедры прикладной информатики и высшей математики

Протокол заседания № 10 от «25» мая 2026 г.

Заведующий кафедрой Стрекалова Наталья Борисовна

ОГЛАВЛЕНИЕ

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ И ЭТАПЫ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ.....	3
2. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ОБЕСПЕЧИВАЮЩИХ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ.....	8
3. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ.....	23
4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНОВАНИЯ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ.....	28
5. ДИАГНОСТИЧЕСКИЕ ЗАДАНИЯ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ УРОВЕНЬ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ.....	44
6. ЛИСТ СОГЛАСОВАНИЯ.....	45

1. ПЕРЕЧЕНЬ КОМПЕТЕНЦИЙ И ЭТАПЫ ИХ ФОРМИРОВАНИЯ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Оценочные материалы (средства) сформированы в соответствии с ФГОС ВО - бакалавриат по направлению подготовки 09.03.03 «Прикладная информатика», утвержденный приказом Министерства образования и науки Российской Федерации от 19.09.2017 № 922 (с изменениями и дополнениями от 26.11.2020, 08.02.2021). В соответствии с матрицей компетенций основной профессиональной образовательной программы по направлению подготовки 09.03.03 «Прикладная информатика» (уровень бакалавриата) в процессе обучения по дисциплине «Разработка облачных приложений и сервисов» происходит формирование закрепленных за дисциплиной компетенций обучающихся. Оценка сформированности компетенций на каждом этапе обучения происходит через оценку планируемых результатов обучения по дисциплине (знаний, умений, навыков).

Шифр и название компетенции	Индикаторы компетенции	Планируемые результаты обучения по дисциплине	Этапы формирования компетенции (семестры и темы)	Оценочные средства
ПК-1. Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта	ПК-1.1. Проводит обследование организаций, выявляет информационные потребности заказчика, используя цифровые инструменты и облачные решения для сбора и анализа данных	Знать: - назначение, суть и способы проведения предпроектного исследования организации облачного приложения; Уметь: - составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения; Владеть: - навыками формализованного представления требований к облачному приложению.	Тема 1. История основных типов высоко-производительных вычислений, тенденции развития современных инфраструктурных решений Тема 2. Виртуализация. Сервисы. Основные направления развития Тема 5. Обзор существующих сервисов. Обзор существующих платформ Тема 9. Программное обеспечение Тема 10. Технологии разработки облачных решений	- устный опрос по темам 1-2, 9-10 - выполнение индивидуального проектного задания по теме 5 - выполнение практического задания по теме 5, -тестирование по теме 5 - ответ на теоретический вопрос на экзамене
	ПК-1.2. Разрабатывает техническое задание, формализует требования заказчика к разработке цифрового программного продукта	Знать: - основные принципы и порядок разработки технического задания в соответствии с требованиями заказчика; Уметь: - формулировать и составлять требования входящие в	Тема 11. Создание сценариев на РНР. Типы данных, переменные, операторы	- выполнение практического задания по теме 11, - ответ на теоретический вопрос на экзамене

		техническое задание на разработку облачного приложения; Владеть: - навыками оформления технического задания в соответствии с ГОСТ.		
ПК-2. Способен проектировать цифровой программный продукт	ПК-2.1. Разрабатывает модель цифрового программного продукта с учетом внедрения передовых ИТ-решений прикладной сферы	Знать: - методологии описания процессов, применяемых в информационной сфере; Уметь: - строить концептуальное решение будущего приложения для предметной области; Владеть: - навыками моделирования прикладных процессов в облачных приложениях;	Тема 4. Экономика облачных вычислений. Достоинства и недостатки облачных вычислений Тема 10. Технологии разработки облачных решений	- устный опрос по теме 10 - выполнение индивидуального проектного задания по теме 4 - выполнение практического задания по теме 4, - ответ на теоретический вопрос на экзамене
	ПК-2.2. Проектирует базы данных, в том числе используя специализированные цифровые платформы	Знать: - компонентный состав современных баз данных и методологические основы их проектирования с использованием различных цифровых платформ; Уметь: - разрабатывать эскизный проект облачного приложения, проектировать модель базы данных; Владеть: - навыками моделирования и описания моделей по заданным требованиям;	Тема 2. Виртуализация. Сервисы. Основные направления развития Тема 3. Введение в понятия облачных вычислений Тема 11. Создание сценариев на РНР. Типы данных, переменные, операторы Тема 19. Сценарии сайта	- устный опрос по теме 2-3, - выполнение индивидуального проектного задания по теме 3 - выполнение практического задания по теме 3,11,19 - ответ на теоретический вопрос на экзамене
	ПК-2.3. Проектирует пользовательский интерфейс с учетом	Знать - способы и методы проектирования пользовательского	Тема 3. Введение в понятия облачных вычислений Тема 19. Сценарии	- устный опрос по теме 3, - выполнение индивидуального

	тенденций развития цифровых технологий	интерфейса облачного приложения; Уметь - разрабатывать макеты пользовательского интерфейса облачного приложения; Владеть - навыками разработки макетов пользовательского интерфейса облачного приложения;	сайта	проектного задания по теме 3 - выполнение практического задания по теме 19 - ответ на теоретический вопрос на экзамене
	ПК-2.4 Проводит анализ современных программно-технических средств, специализированных цифровых платформ, облачных и отраслевых решений для разработки цифрового программного продукта	Знать: - методы и инструменты исследования и анализа жизненного цикла цифрового продукта Уметь: - масштабировать, развертывать и делать резервное копирование облачного приложения Владеть: - навыками сбора необходимой информации облачных приложений;	Тема 6. Технологии облачных вычислений Тема 8. Работа с данными в облачных офисах	- устный опрос по теме 8 - выполнение индивидуального проектного задания по теме 6 - выполнение практического задания по теме 6 - ответ на теоретический вопрос на экзамене
ПК-3. Способен разрабатывать цифровой программный продукт	ПК-3.1. Осуществляет разработку программного кода и компонентов цифрового программного продукта с помощью специализированных цифровых платформ для индивидуальной и совместной разработки	Знать - синтаксис современных языков программирования применяемых в разработке облачных приложений; - специализированные цифровые платформы; Уметь - пользоваться приемами облачного программирования; Владеть - навыками работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений.	3 семестр Тема 4. Экономика облачных вычислений. Достоинства и недостатки облачных вычислений Тема 5. Обзор существующих сервисов. Обзор существующих платформ Тема 6. Технологии облачных вычислений Тема 7. Миграция из стандартной среды в облачные приложения Тема 8. Работа с данными в облачных офисах 4 семестр Тема 12. Операции	- устный опрос по теме 3,8,17,22 - выполнение индивидуального проектного задания по теме 3-6,12-16,18,21 - выполнение практического задания по теме 4-7,19-20, - тестирование по теме 6 - ответ на теоретический вопрос на экзамене

			и управляющие конструкции Тема 13. Функции и повторное использование кода Тема 14. Массивы Тема 15. Передача данных через HTML-формы Тема 16. Работа с файлами Тема 17. Строковые функции и регулярные выражения Тема 18. Cookies и управление сессиями Тема 19. Сценарии сайта Тема 20. Расширение mysqli для работы с базами данных Тема 21. XML и PHP	
	ПК-3.2. Обеспечивает интеграцию программных модулей и компонентов цифрового программного продукта на основе облачных и отраслевых ИТ-решений	Знать - основные понятия и терминологию облачных технологий; - области применения облачных технологий; Уметь - применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений; Владеть - навыками работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений;	Тема 12. Операции и управляющие конструкции Тема 13. Функции и повторное использование кода Тема 14. Массивы Тема 15. Передача данных через HTML-формы Тема 18. Cookies и управление сессиями	- выполнение индивидуального проектного задания по теме 12-15,18 - ответ на теоретический вопрос на экзамене
ПК-5. Способен обеспечивать качество	ПК-5.2. Проводит юзабилити-аудит цифрового программного	Знать: - концепцию облачных вычислений применительно к	Тема 6. Технологии облачных вычислений	- устный опрос по теме 7,22 - выполнение индивидуального

работы цифрового программного продукта	продукта	бизнес-деятельности; - инфраструктуру облачных вычислений; Уметь: - оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений; Владеть: - навыком разработки программного обеспечения облачных систем.	Тема 7. Миграция из стандартной среды в облачные приложения Тема 8. Работа с данными в облачных офисах Тема 16. Работа с файлами Тема 20. Расширение mysql для работы с базами данных	проектного задания по теме 6,16 - выполнение практического задания по теме 6-7,20 - тестирование по теме 6 - ответ на теоретический вопрос на экзамене
	ПК-5.3. Осуществляет работы по интеграционному тестированию цифрового программного продукта с внешним окружением	Знать: - способы тестирования и отладки облачных приложений; Уметь: - осуществлять организационную подготовку тестирования облачного приложения (или отдельных компонентов) представителями заказчика, проведение и фиксацию результатов; Владеть: - навыком системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках.	Тема 7. Миграция из стандартной среды в облачные приложения Тема 8. Работа с данными в облачных офисах Тема 16. Работа с файлами	- устный опрос по теме 7, - выполнение индивидуального проектного задания по теме 16 - выполнение практического задания по теме 7 - ответ на теоретический вопрос на экзамене

2. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ И ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ОБЕСПЕЧИВАЮЩИХ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ В ПРОЦЕССЕ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Устные опросы

Примерные вопросы для проведения устного опроса по теме 1 «История основных типов высокопроизводительных вычислений, тенденции развития современных инфраструктурных решений»

1. Основные этапы развития аппаратного и программного обеспечения.
2. Современные тенденции развития аппаратного обеспечения облачных вычислений.
3. Появление, развитие и использовании технологий облачных вычислений.
4. Многопроцессорные и многоядерные вычислительные системы.
5. Появление систем и сетей хранения данных.
6. Консолидация инфраструктуры.

*Примерные вопросы для проведения устного опроса
по теме 2 «Виртуализация. Сервисы. Основные направления развития»*

1. Основные типы виртуализации.
2. Обзор программных продуктов крупнейших компаний виртуализации.
3. Виртуальная машина.
4. Виртуализация серверов.
5. Виртуализация приложений.
6. Виртуализация представлений (рабочих мест).
7. Разновидности архитектуры гипервизора.

*Примерные вопросы для проведения устного опроса
по теме «Введение в понятия облачных вычислений»*

1. Архитектура облачных систем.
2. Модели развёртывания облаков: частное облако.
3. Модели развёртывания облаков: публичное облако.
4. Модели развёртывания облаков: гибридное облако.
5. Модели развёртывания облаков: общественное облако.
6. Основные модели предоставления услуг облачных вычислений: Software as a Service (SaaS) (ПО-как-услуга).
7. Основные модели предоставления услуг облачных вычислений: Platform as a Service (PaaS).
8. Основные модели предоставления услуг облачных вычислений: инфраструктура как сервис (Infrastructure as a Service, IaaS).
9. Другие облачные сервисы (XaaS).
10. Различия между облачными и кластерными (распределенными, или - Grid-технологиями) вычислениями.

*Примерные вопросы для проведения устного опроса
по теме «Работа с данными в облачных офисах»*

1. Методы примитивов.
2. Числа. Строки.
3. Массивы. Методы массивов.
4. Перебираемые объекты.
5. Map и Set. WeakMap и WeakSet. Object.keys, values, entries.
6. Деструктурирующее присваивание.
7. Дата и время.
8. Формат JSON, метод toJSON

*Примерные вопросы для проведения устного опроса
по теме «Программное обеспечение»*

1. История развития Интернет.
2. Архитектура облачного приложения.
3. Передача информации между приложением и веб-сервером.
4. Сессии. Куки.

*Примерные вопросы для проведения устного опроса
по теме «Технологии разработки облачных решений»*

1. Особенности создания программного продукта.
2. Понятие «Локального сервера».
3. Разновидности программных средств для разработки ПО.
4. ПО «Open Server» (Apache, PHP, MySQL, Perl).
5. Программная оболочка для разработки сайтов на локальной машине.
6. Оценка стоимости ошибок.

*Примерные вопросы для проведения устного опроса
по теме «Строковые функции и регулярные выражения»*

1. Строки в РНР.
2. Строковые операции.
3. Регулярные выражения.
4. Динамическое включение файлов.

*Примерные вопросы для проведения устного опроса
по теме «Новостная лента RSS»*

1. Новостная лента RSS в формате 2.0.
2. Обязательные и необязательные элементы канала.
3. Кроссбраузерность.
4. Кроссплатформенность.
5. Адаптивная верстка.
6. Media запросы.
7. Адаптивное меню.
8. Адаптивная галерея.
9. Требования к макету.

Критерии оценивания устного опроса:

- оценка «*отлично*» выставляется обучающемуся, если он ответил развернуто на один из заданных вопросов, активно дополнял ответы других обучающихся или задавал им дополнительные вопросы;
- оценка «*хорошо*» выставляется обучающемуся, если он ответил развернуто на один из заданных вопросов или ответил кратко на ряд вопросов;
- оценка «*удовлетворительно*» выставляется обучающемуся, если он кратко ответил на один из задаваемых вопросов или просто дополнял ответы других обучающихся, задавал им дополнительные вопросы;
- оценка «*неудовлетворительно*» выставляется обучающемуся, если он не смог ответить правильно на заданный вопрос, либо не отвечал на вопросы и не участвовал в их обсуждении.

Практические работы

Практическое задание по теме «Экономика облачных вычислений. Достоинства и недостатки облачных вычислений»

Задание: На основе представленного кейса рассчитать экономические показатели облачного решения.

Критерии оценивания:

- оценка «*отлично*» выставляется обучающемуся, если задание было выполнено полном объеме, расчеты верны, обучающийся грамотно объясняет полученное решение;
- оценка «*хорошо*» выставляется обучающемуся, если задание было выполнено частично, расчеты верны, обучающийся объясняет полученное решение;

- оценка «удовлетворительно» выставляется обучающемуся, если задание было выполнено частично, но расчетах имеются критические ошибки, обучающийся объясняет полученное решение не полностью;
- оценка «неудовлетворительно» выставляется обучающемуся, если задание не выполнено или выполнено неверно, не отвечает на поставленные вопросы.

Практическое задание по теме «Обзор существующих сервисов. Обзор существующих платформ»

Задание: На основе представленного кейса провести обзор существующих платформ и обосновать выбор конкретного. Оформить отчет по требованиям.

Критерии оценивания:

- оценка «отлично» выставляется обучающемуся, если задание было выполнено полном объеме, отчет оформлен по требованиям, обучающийся грамотно объясняет полученное решение;
- оценка «хорошо» выставляется обучающемуся, если задание было выполнено частично, отчет оформлен по требованиям, обучающийся грамотно объясняет полученное решение;
- оценка «удовлетворительно» выставляется обучающемуся, если задание было выполнено частично, но оформлен с критическими ошибками, обучающийся частично объясняет полученное решение;
- оценка «неудовлетворительно» выставляется обучающемуся, если задание не выполнено или выполнено неверно, не отвечает на поставленные вопросы.

Практическое задание по теме «Технологии разработки облачных вычислений»

Задание: Провести аналитический обзор программных средств для разработки ПО и программных оболочек для разработки сайтов на локальной машине. Результат работы представить в виде аналитического отчета.

Критерии оценивания:

- оценка «отлично» выставляется обучающемуся, если составлен в полном объеме аналитический обзор рынка программных средств для разработки ПО и программных оболочек, а также проведён сравнительный анализ программно-технических средств и информационных продуктов в соответствии с полученным заданием;
- оценка «хорошо» выставляется обучающемуся, если аналитический обзор рынка программных средств для разработки ПО и программных оболочек выполнен с незначительными замечаниями, а также проведён сравнительный анализ программно-технических средств и информационных продуктов в соответствии с полученным заданием;
- оценка «удовлетворительно» выставляется обучающемуся, если аналитический обзор рынка программных средств для разработки ПО и программных оболочек выполнен с критическими ошибками и замечаниями, частично проведён сравнительный анализ

программно-технических средств и информационных продуктов в соответствии с полученным заданием;

- оценка *«неудовлетворительно»* выставляется обучающемуся, если аналитический отчет не составлен.

Практическое задание по теме «Миграция из стандартной среды в облачные приложения»

Задание: Провести аналитический обзор программных средств и решить проблемы перехода (технических, финансовых, безопасности, лицензионных и законодательных) по представленному кейсу. Результат работы представить в виде аналитического отчета.

Критерии оценивания:

- оценка *«отлично»* выставляется обучающемуся, если составлен в полном объеме аналитический обзор рынка программных средств для разработки ПО и программных оболочек, а также проведен сравнительный анализ программно-технических средств и информационных продуктов в соответствии с полученным заданием;
- оценка *«хорошо»* выставляется обучающемуся, если аналитический обзор рынка программных средств для разработки ПО и программных оболочек выполнен с незначительными замечаниями, а также проведен сравнительный анализ программно-технических средств и информационных продуктов в соответствии с полученным заданием;
- оценка *«удовлетворительно»* выставляется обучающемуся, если аналитический обзор рынка программных средств для разработки ПО и программных оболочек выполнен с критическими ошибками и замечаниями, частично проведен сравнительный анализ программно-технических средств и информационных продуктов в соответствии с полученным заданием;
- оценка *«неудовлетворительно»* выставляется обучающемуся, если аналитический отчет не составлен.

Тестирование «Технологии облачных вычислений»

1. Каким образом можно подключить скрипты к HTML?
прописать внутри кода страницы
подключить отдельным файлом
оба варианты верны
2. Какой оператор вернет тип переменной, переданной ему в качестве аргумента?
native
instanceof
typeof
3. Что определяет переменная?

Символическое имя для значения

Тип данных

Тип значения

4. Что означает значение undefined?

Ошибочный результат вычислений

Что значение не было присвоено

Ссылкой на несуществующий объект

5. К какому типу относится значение NaN?

К числовому

К строковому

К логическому

6. Есть ли разница в JavaScript между двойными и одинарными кавычками при записи строковых литералов?

Нет

Да

Только при работе с jQuery

7. Что представляет собой значение Infinity?

Вычислительную ошибку

Математическую бесконечность

Неопределённое значение

8. К какому типу относится строка в JavaScript?

Объект

Простой тип

9. К какому типу относится значение null?

К символьному

К строковому

К числовому

К логическому

Ни к одному из перечисленных

10. Чем в JavaScript является функция?

Объектом

Последовательностью

Строкой

11. Существует ли в JavaScript символьный тип данных?

Нет

Да

Только при работе с Node.js

12. Можно ли в JavaScript в переменную, в которой ранее хранилось целое число записать

объект?

Можно

Нельзя

13. К какой типизации относится JavaScript?

К динамической и слабой

К сильной и статической

К явной и динамической

14. Сколько значений в JavaScript может принимать логический тип?

Одно

Два

Три

15. Существует ли в JavaScript отдельный тип данных для чисел с плавающей запятой?

Да, это тип float

Да, это два типа float и double

Нет, за целочисленные значения и числа с плавающей точкой в JavaScript отвечает тип number

Критерии оценивания тестирования:

- оценка «*отлично*» выставляется обучающемуся, если он ответил верно на 13-15 вопросов;
- оценка «*хорошо*» выставляется обучающемуся, если он ответил верно на 10-12 вопросов;
- оценка «*удовлетворительно*» выставляется обучающемуся, если он ответил верно на 8-10 вопросов;
- оценка «*неудовлетворительно*» выставляется обучающемуся, если ответил верно менее, чем на 8 вопросов.

Практическое задание по теме «Создание сценариев на PHP. Типы данных, переменные, операторы»

Задание 1: Запустить «OpenServer». После запуска создается диск Z. На диске Z в папке www создать папку myphp. Создать файл в Notepad++ с именем index.php. Наполнить файл html-тэгами так, чтобы получился каркас странички html. Вписать в тело странички php скрипт для вывода на экран строки текста. Тут же создаем файл style.css, заполняем его информацией о стилях отображения странички. Теперь открываем браузер и в адресной строке набираем адрес нового сайта и полный путь до запускаемого файла, в нашем случае этот путь будет выглядеть вот так: `http://localhost/myphp/index.php` и нажимаем Enter. Если Вы все сделали правильно, то на мониторе (в браузере) Вы увидите свою первую страничку.

Критерии оценивания:

- оценка «*отлично*» выставляется обучающемуся, если страничка сверстана правильно, используются стили, php-скрипт обработан препроцессором и выводит на экран нужную информацию;
- оценка «*хорошо*» выставляется обучающемуся, если страничка сверстана правильно. Php-скрипт обработан препроцессором и выводит на экран нужную информацию;
- оценка «*удовлетворительно*» выставляется обучающемуся, если страничка сверстана правильно, используются стили\$
- оценка «*неудовлетворительно*» выставляется обучающемуся, если страничка сверстана правильно, нет понимания работы препроцессора.

Задание 2:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServerZ:\home\localhost\www. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Изучить материал о переменных и о выводе информации на экран в PHP. Составить код программы в связке html и php для вывода на экран значений переменных, строк, а также выводим одновременно переменные и текстовые строки.

Критерии оценивания:

- оценка «*отлично*» выставляется обучающемуся, если все созданные файлы с расширением php. Файлы обрабатываются препроцессором правильно и выводится нужная информация. Студент понимает принципы работы с переменными и командами отображения информации на экране;
- оценка «*хорошо*» выставляется обучающемуся, если все созданные файлы с расширением php, файлы обрабатываются препроцессором правильно и выводится нужная информация, студент понимает принципы работы с командами отображения информации на экране;
- оценка «*удовлетворительно*» выставляется обучающемуся, если все созданные файлы с расширением php, файлы обрабатываются препроцессором правильно и выводится нужная информация, студент понимает принципы работы с переменными;
- оценка «*неудовлетворительно*» выставляется обучающемуся, если студент не понимает принципов работы с переменными и командами отображения информации на экране.

Практическое задание по теме «Операции и управляющие конструкции»

Задание:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServerZ:\home\localhost\www. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Изучить материал об операторах и операциях в PHP. Составить коды программ в связке html и php для решения заданий:

- написать страничку php, где вычисляется текущая дата и время;
- написать код программы, позволяющий выводить на экран сумму кубов и квадратов натуральных чисел, начиная с 10 до 20;
- составить код программы, позволяющий выводить на экран только четные числа квадратов и кубов натуральных чисел с 10 до 20;
- составить код программы, позволяющий выводить на экран только нечетные числа квадратов и кубов чисел, начиная с 10 до 20;
- составить код программы, позволяющий выводить на экран разность, умножение и деление кубов и квадратов натуральных чисел, начиная с 20 до 30;
- составить код программы, где переменной \$a присваивается значение 4, а переменной \$b значение 5 и вывести на экран переменные и значения в них.
- составить код программы, где используются операции инкремента и декремента;
- составить код программы, где используются битовые операции;
- составить код программы, где используются операции сравнения.

Критерии оценивания:

- оценка «отлично» выставляется обучающемуся, если в архиве содержаться 9 файлов php, где один из них играет ведущую роль, в каждом файле составлен рабочий код программы, согласно заданиям, все файлы имеют презентабельный вид;
- оценка «хорошо» выставляется обучающемуся, если в архиве содержаться 7 файлов php, где один из них играет ведущую роль, в каждом файле составлен рабочий код программы, согласно заданиям, все файлы имеют презентабельный вид;
- оценка «удовлетворительно» выставляется обучающемуся, если в архиве содержаться 6 файлов php, где один из них играет ведущую роль или 8 простых файлов php, в каждом файле составлен рабочий код программы, согласно заданиям;
- оценка «неудовлетворительно» выставляется обучающемуся, если в архиве содержаться файлы php. В файлах составлен код программ, который не работоспособен.

Практическое задание по теме «Функции и повторное использование кода»

Задание:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServer. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Составить код программы PHP всех операций и вывести результаты вычислений на экран для конструкций:

- if-else;
- двойного условия if-else;
- вложенных if-else;

- break;
- switch-case.

Критерии оценивания:

- оценка «*отлично*» выставляется обучающемуся, если в архиве содержаться 6 файлов php, каждый из них выводит результаты вычислений на экран, в каждом файле составлен рабочий код программы, согласно заданиям, все файлы имеют презентабельный вид;
- оценка «*хорошо*» выставляется обучающемуся, если в архиве содержаться 4 файла php, каждый из них выводит результаты вычислений на экран, в каждом файле составлен рабочий код программы, согласно заданиям, все файлы имеют презентабельный вид;
- оценка «*удовлетворительно*» выставляется обучающемуся, если в архиве содержаться 3 файла php каждый из них выводит результаты вычислений на экран, в каждом файле составлен рабочий код программы, согласно заданиям;
- оценка «*неудовлетворительно*» выставляется обучающемуся, если в архиве содержаться файлы php, в файлах составлен код программ, который не работоспособен.

Практическое задание по теме «Массивы»

Задание:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServerZ:\home\localhost\www. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Изучить материал о разновидности массивов в PHP. Составить коды программ для реализации работы массивов:

- простого массива;
- автомассива;
- ассоциативного массива;
- многомерного массива.

Критерии оценивания:

- оценка «*отлично*» выставляется обучающемуся, если в архиве содержаться 6 файлов php, каждый из которых отображает работу массива, все массивы в заданиях разнообразны, в каждом файле составлен рабочий код программы, согласно заданиям, все файлы имеют презентабельный вид;
- оценка «*хорошо*» выставляется обучающемуся, если в архиве содержаться 5 файлов php, каждый из которых отображает работу массива, все массивы в заданиях разнообразны, в каждом файле составлен рабочий код программы, согласно заданиям, все файлы имеют презентабельный вид;
- оценка «*удовлетворительно*» выставляется обучающемуся, если в архиве содержаться 4 файлов php, каждый из которых отображает работу массива, все массивы в заданиях разнообразны, в каждом файле составлен рабочий код программы, согласно заданиям;

- оценка «*неудовлетворительно*» выставляется обучающемуся, если в архиве содержатся файлы php, в файлах составлен код программ, который не работоспособен.

Практическое задание по теме «Передача данных через HTML-формы»

Задание:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServerZ:\home\localhost\www. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Изучить материал о формах в HTML. Разработать формы HTML с различными элементами управления.

3. Изучить материал о передачи данных из форм HTML средствами php. Разработать формы и сценарий для передачи данных из формы методами POST, GET.

Критерии оценивания:

Критерии оценивания:

- оценка «*отлично*» выставляется обучающемуся, если в архиве содержатся 18 файлов php, 2 из которых отображает построение форм с различными элементами управления, все массивы в заданиях разнообразны, в каждом файле составлен рабочий код программы, согласно заданиям, все файлы имеют презентабельный вид;
- оценка «*хорошо*» выставляется обучающемуся, если в архиве содержатся 5 файлов php, каждый из которых отображает работу массива, все массивы в заданиях разнообразны, в каждом файле составлен рабочий код программы, согласно заданиям. Все файлы имеют презентабельный вид;
- оценка «*удовлетворительно*» выставляется обучающемуся, если в архиве содержатся 4 файлов php, каждый из которых отображает работу массива, все массивы в заданиях разнообразны, в каждом файле составлен рабочий код программы, согласно заданиям;
- оценка «*неудовлетворительно*» выставляется обучающемуся, если в архиве содержатся файлы php. В файлах составлен код программ, который не работоспособен.

Практическое задание по теме «Работа с файлами»

Задание:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServerZ:\home\localhost\www. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Получение практических навыков работы с файлами. Создать панель управления файлами согласно рисунку.

Критерии оценивания:

- оценка *«отлично»* выставляется обучающемуся, если набранный код программы имеет вид панели управления файлами, панель управления файлами функционирует, в панели отображаются имена файлов и размер, выбранные файлы в панели можно удалить, переименовать, выгрузить;
- оценка *«хорошо»* выставляется обучающемуся, если набранный код программы имеет вид панели управления файлами, панель управления файлами функционирует, в панели отображаются имена файлов и размер, выбранные файлы в панели можно удалить, переименовать;
- оценка *«удовлетворительно»* выставляется обучающемуся, если набранный код программы имеет вид панели управления файлами, панель управления файлами частично функционирует, в панели отображаются имена файлов и размер;
- оценка *«неудовлетворительно»* выставляется обучающемуся, если набранный код программы не функционирует.

Практическое задание по теме «Cookies и управление сессиями»

Задание:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServerZ:\home\localhost\www. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Получение практических навыков работы с cookie. Работа с cookies файлами. Создать панель управления cookie согласно рисунку.

Критерии оценивания:

- оценка *«отлично»* выставляется обучающемуся, если набранный код программы имеет вид панели управления cookie, панель управления cookie функционирует, в панели отображаются количество посещения страницы пользователем, цвета фона, новый цвет текста и возможность удаления куки;
- оценка *«хорошо»* выставляется обучающемуся, если набранный код программы имеет вид панели управления cookie, панель управления cookie функционирует, в панели отображаются количество посещения страницы пользователем, цвета фона, новый цвет текста;
- оценка *«удовлетворительно»* выставляется обучающемуся, если набранный код программы имеет вид панели управления cookie, панель управления cookie частично функционирует;
- оценка *«неудовлетворительно»* выставляется обучающемуся, если набранный код программы не функционирует.

Практическое задание по теме «Сценарии сайта»

Задание: создать домашнюю страницу сайта с возможностью выбора и заказа тура. Также создать страницы описаний стран, городов и отелей. Учитывать функцию администрирования сайта.

Критерии оценивания:

- оценка «отлично» выставляется обучающемуся, если набранный код программы соответствует теме, страницы работают и наполнены контентом, учтен вход администратора сайта;
- оценка «хорошо» выставляется обучающемуся, набранный код программы соответствует теме, страницы работают и наполнены контентом с незначительными замечаниями, учтен вход администратора сайта;
- оценка «удовлетворительно» выставляется обучающемуся, если набранный код программы соответствует теме, страницы работают и наполнены контентом со значительными замечаниями, учтен вход администратора сайта;
- оценка «неудовлетворительно» выставляется обучающемуся, если код сайта не функционирует.

Практическое задание по теме «Расширение mysqli для работы с базами данных»

Задание: создать скрипт для работы с MySQL. Осуществить подключение к серверу и выбор базы данных. Выполнить запросы к базе данных.

Критерии оценивания:

- оценка «отлично» выставляется обучающемуся, если верно и в полном объеме написан скрипт для работы с MySQL, осуществлено подключение к серверу и произведён выбор базы данных, запросы выполнены к базе данных;
- оценка «хорошо» выставляется обучающемуся, верно и в полном объеме написан скрипт для работы с MySQL, осуществлено подключение к серверу и произведён выбор базы данных, запросы выполнены к базе данных с незначительными замечаниями;
- оценка «удовлетворительно» выставляется обучающемуся, если верно и в полном объеме написан скрипт для работы с MySQL, осуществлено подключение к серверу и произведён выбор базы данных, запросы выполнены к базе данных со значительными замечаниями;
- оценка «неудовлетворительно» выставляется обучающемуся, если скрипт и подключение не выполнено.

Практическое задание по теме «XML и PHP»

Задание:

1. Необходимо создать свою папку хранения интернет - страниц на web - сервере OpenServerZ:\home\localhost\www. Имя папке даем произвольное. Выполненные работы необходимо сохранять с расширением «.php».

2. Получение практических навыков работы с XML. Организация работы с данными в xml-формате. Для этого изучить теоретический материал и исходные тексты примеров серверных скриптов
3. Выполните верстку дополнительных страниц сайта.
4. Протестировать и отладить скрипты, динамически формирующие страницы сайта.

Критерии оценивания:

- оценка «*отлично*» выставляется обучающемуся, если набранный код программы содержит данные в xml-формате, сайт полностью функционирует, результаты тестирования запротоколированы;
- оценка «*хорошо*» выставляется обучающемуся, если набранный код программы содержит данные в xml-формате, сайт полностью функционирует, результаты тестирования зафиксированы;
- оценка «*удовлетворительно*» выставляется обучающемуся, если набранный код программы содержит данные в xml-формате, сайт частично функционирует;
- оценка «*неудовлетворительно*» выставляется обучающемуся, если набранный код программы не функционирует.

Задание для работы в малых группах (либо индивидуальное выполнение), выполняемого в 4 семестре:

Цель: *разработка облачного решения для заданной предметной области.*

Общее задание: Обучающиеся разбиваются на небольшие группы из 2-3 человек. По выданному преподавателем техническому заданию разрабатывается каркас сайта, дизайн сайта и его контент. Приветствуется реальное техническое задание. В противном случае тема может быть выбрана обучающимися произвольно.

Критерии оценивания:

оценка «*отлично*» выставляется обучающемуся, если он понимает и воспроизводит принципы построения Web-приложений, составляет код программы и воспроизводит взаимодействие PHP, HTML и MySQL, составленный код программы работает без сбоев, организована проверка на ошибки при вводе данных, предусмотрена защита данных;

оценка «*хорошо*» выставляется обучающемуся, если он понимает и воспроизводит принципы построения Web-приложений, воспроизводит взаимодействие PHP, HTML и MySQL, составленный код программы работает без сбоев, организована проверка на ошибки при вводе данных;

оценка «*удовлетворительно*» выставляется обучающемуся, если он понимает принципы построения Web-приложений, воспроизводит взаимодействие каркаса и базы данных web-приложений, составленный код программы работает без сбоев;

оценка «неудовлетворительно» выставляется обучающемуся, если не понимает принципы построения Web-приложений, не знает определение баз данных и характеристику средств разработки, программный код не составляет

3. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (СРЕДСТВА) ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ

ОЦЕНКА ЗНАНИЙ ОБУЧАЮЩИХСЯ НА ЭКЗАМЕНЕ

Список вопросов для подготовки к экзамену

ПК-1. Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта

ПК-1.1. Проводит обследование организаций, выявляет информационные потребности заказчика, используя цифровые инструменты и облачные решения для сбора и анализа данных

Обучающийся знает: назначение, суть и способы проведения предпроектного исследования организации облачного приложения

1. Понятие облачного решения
2. Информационная модель данных и ее структура.
3. Основные понятия реляционных баз данных: поля, записи, индексы, ключи, таблицы, отношения.
4. Уровни модели данных

ПК-1. Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта

ПК-1.2. Разрабатывает техническое задание, формализует требования заказчика к разработке цифрового программного продукта

Обучающийся знает: основные принципы и порядок разработки технического задания в соответствии с требованиями заказчика

1. Инструментарий для проектирования и документирования баз данных
2. Выборка информации из БД.
3. Резервное копирование. Восстановление базы данных.
4. Защита базы данных резервного копирования или каталога
5. Создание и поддержка открытых отчетов, отчетов об открытых проблемах
6. Жизненный цикл программы

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.1. Разрабатывает модель цифрового программного продукта с учетом внедрения передовых ИТ-решений прикладной сферы

Обучающийся знает: методологии описания процессов, применяемых в информационной сфере

1. Основные преимущества и недостатки моделей облачных вычислений и предлагаемых на их основе решений.
2. Понятие «Локального сервера».
3. Разновидности программных средств для разработки ПО. ПО «Open Server» (Apache, PHP, MySQL, Perl).

4. Программная оболочка для разработки сайтов на локальной машине.
5. Оценка стоимости ошибок.

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.2. Проектирует базы данных, в том числе используя специализированные цифровые платформы

Обучающийся знает: компонентный состав современных баз данных и методологические основы их проектирования с использованием различных цифровых платформ

1. Назначение и виды информационных систем, создаваемых на основе современных СУБД
2. Основные функции СУБД в разных типах информационных систем
3. Информационная модель данных и ее структура
4. Логические модели баз данных и их типы
5. Формы нормализации данных в базе
6. Инструменты и принципы разработки Web- приложения
7. Жизненный цикл программы
8. Этапы создания программных продуктов

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.3. Проектирует пользовательский интерфейс с учетом тенденций развития цифровых технологий

Обучающийся знает: способы и методы проектирования пользовательского интерфейса облачного приложения

1. Организационные процессы жизненного цикла программного продукта
2. Этапы создания программных продуктов
3. Модели жизненного цикла
4. Тестирование
5. Техническое задание. ГОСТ 34.602-89, 15.016-2016

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.4. Проводит анализ современных программно-технических средств, специализированных цифровых платформ, облачных и отраслевых решений для разработки цифрового программного продукта

Обучающийся знает: методы и инструменты исследования и анализа жизненного цикла цифрового продукта

1. Инструменты и принципы разработки Web- приложения
2. Методы анализа рынка программно-технических средств, информационных продуктов и услуг
3. Основные источники профессионально-ориентированной информации, предназначенной для проведения анализа рынка программно-технических средств
4. Использование информационно-библиографических и справочно-правовых систем

ПК-3. Способен разрабатывать цифровой программный продукт

ПК-3.1. Осуществляет разработку программного кода и компонентов цифрового программного продукта с помощью специализированных цифровых платформ для индивидуальной и совместной разработки

Обучающийся знает: синтаксис современных языков программирования применяемых в разработке облачных приложений; специализированные цифровые платформы

1. Структура кода.
2. Основные этапы построения web-приложений.
3. Основы конструирования запросов в СУБД.
4. Назначение и команды структурированного языка запросов (SQL)
5. Облачные цифровые платформы

ПК-3. Способен разрабатывать цифровой программный продукт

ПК-3.2. Обеспечивает интеграцию программных модулей и компонентов цифрового программного продукта на основе облачных и отраслевых ИТ-решений

Обучающийся знает: основные понятия и терминологию облачных технологий; области применения облачных технологий

1. Фазы миграции в облако.
2. Выбор подходящей модели развертывания в соответствии с существующими бизнес-задачами.
3. Выбор подходящего поставщика облачных услуг.
4. Концепция SLA..
5. Концепция вендора.
6. Открытые стандарты для обеспечения облачных услуг.
7. Решение проблем перехода: технических, финансовых, безопасности, лицензионных и законодательных.

ПК-5. Способен обеспечивать качество работы цифрового программного продукта

ПК-5.2. Проводит юзабилити-аудит цифрового программного продукта

Обучающийся знает: концепцию облачных вычислений применительно к бизнес-деятельности; инфраструктуру облачных вычислений

1. Понятие юзабилити
2. Руководство пользователя, администратора
3. Инструменты разработки руководства пользователя
4. Правила подготовки презентации.
5. Техническое задание. ГОСТ 34.602-89, 15.016-2016
6. Основные источники профессионально-ориентированной информации, предназначенной

ПК-5. Способен обеспечивать качество работы цифрового программного продукта

ПК-5.3. Осуществляет работы по интеграционному тестированию цифрового программного продукта с внешним окружением

Обучающийся знает: способы тестирования и отладки облачных приложений

1. Назначение тестирования
2. Виды и способы тестирования баз данных и web-приложений
3. Процедура тестирования
4. Качество кода

ОЦЕНКА УМЕНИЙ И НАВЫКОВ ОБУЧАЮЩИХСЯ НА ЭКЗАМЕНЕ

ПК-1. Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта

ПК-1.1. Проводит обследование организаций, выявляет информационные потребности заказчика, используя цифровые инструменты и облачные решения для сбора и анализа данных

Обучающийся умеет: составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения

Обучающийся владеет: навыками формализованного представления требований к облачному приложению

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практического задания, в котором обучающийся демонстрирует проведенное обследование организации, определяет информационные потребности пользователей к облачному приложению

ПК-1. Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта

ПК-1.2. Разрабатывает техническое задание, формализует требования заказчика к разработке цифрового программного продукта

Обучающийся умеет: формулировать и составлять требования входящие в техническое задание на разработку облачного приложения

Обучающийся владеет: навыками оформления технического задания в соответствии с ГОСТ.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практического задания, в котором обучающийся формирует техническое задание к проекту.

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.1. Разрабатывает модель цифрового программного продукта с учетом внедрения передовых ИТ-решений прикладной сферы

Обучающийся умеет: строить концептуальное решение будущего облачного приложения для предметной области

Обучающийся владеет: навыками моделирования прикладных процессов в облачных приложениях

Оценка достижения запланированного результата обучения осуществляется по результатам индивидуального проектного задания и практических заданий в малых группах, в котором обучающийся обосновывает выбор инструментария для реализации поставленных задач во время разработки облачного приложения на основе проектного решения

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.2. Проектирует базы данных, в том числе используя специализированные цифровые платформы

Обучающийся умеет: разрабатывать эскизный проект облачного приложения, проектировать модель базы данных

Обучающийся владеет: навыками моделирования и описания моделей по заданным требованиям.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практического задания, в котором обучающийся демонстрирует передачу информации между приложением и веб-сервером, адаптирует под конкретные потребности предприятия.

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.3. Проектирует пользовательский интерфейс с учетом тенденций развития цифровых технологий

Обучающийся умеет: разрабатывать макеты пользовательского интерфейса облачного приложения

Обучающийся владеет: навыками разработки макетов пользовательского интерфейса облачного приложения.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практического задания, в котором обучающийся демонстрирует разработанный пользовательский интерфейс облачного приложения

ПК-2. Способен проектировать цифровой программный продукт

ПК-2.4. Проводит анализ современных программно-технических средств, специализированных цифровых платформ, облачных и отраслевых решений для разработки цифрового программного продукта

Обучающийся умеет: масштабировать, развертывать и делать резервное копирование облачного приложения

Обучающийся владеет: навыками сбора необходимой информации облачных приложений

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практического задания, в котором обучающийся демонстрирует работу с программно-технологическими средствами (поисковыми машинами, справочно-правовыми и экспертными системами, открытыми информационными ресурсами и т.п.) для выполнения задач исследовательского и аналитического характера

ПК-3. Способен разрабатывать цифровой программный продукт

ПК-3.1. Осуществляет разработку программного кода и компонентов цифрового программного продукта с помощью специализированных цифровых платформ для индивидуальной и совместной разработки

Обучающийся умеет: пользоваться приемами облачного программирования

Обучающийся владеет: навыками работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений.

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам выполнения практических заданий и практических заданий в малых группах, в котором обучающийся разрабатывает облачное приложение

ПК-3. Способен разрабатывать цифровой программный продукт

ПК-3.2. Обеспечивает интеграцию программных модулей и компонентов цифрового программного продукта на основе облачных и отраслевых ИТ-решений

Обучающийся умеет: применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений

Обучающийся владеет: навыками работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практических заданий и практического задания в малых группах, в котором обучающийся демонстрирует работу с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений.

ПК-5. Способен обеспечивать качество работы цифрового программного продукта

ПК-5.2. Проводит юзабилити-аудит цифрового программного продукта

Обучающийся умеет: оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений

Обучающийся владеет: навыками разработки программного обеспечения облачных систем

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практических заданий в малых группах, в котором обучающийся проводит и протоколирует результаты тестирования.

ПК-5. Способен обеспечивать качество работы цифрового программного продукта

ПК-5.3. Осуществляет работы по интеграционному тестированию цифрового программного продукта с внешним окружением

Обучающийся умеет осуществлять организационную подготовку тестирования облачного приложения (или отдельных компонентов) представителями заказчика, проведение и фиксацию результатов

Обучающийся владеет: навыками системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках

Оценка достижения обучающимся запланированного результата обучения осуществляется по результатам практических заданий и практических заданий в малых группах, в котором обучающийся проводит и протоколирует результаты тестирования.

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНОВАНИЯ ЗНАНИЙ, УМЕНИЙ И НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ ЭТАПЫ ФОРМИРОВАНИЯ КОМПЕТЕНЦИЙ

ПРОЦЕДУРА ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Для контроля усвоения обучающимися данной дисциплины и достижения запланированных результатов обучения учебным планом предусмотрен экзамен, проводимый в форме устного ответа на теоретический вопрос и выполнение практического задания. При выставлении итоговой оценки по дисциплине учитываются результаты текущего контроля, отражающего сформированность практических умений и навыков обучающегося. Во время обучения обучающиеся участвуют в разных мероприятиях текущего контроля: участвуют в устных теоретических опросах, выполняют практические задания и практические задания в малых группах. Для допуска к экзамену обучающемуся необходимо выполнить все практические задания и выполнить к экзамену задание в малых группах. Допустимо совместить защиту практического задания в малых группах и ответ на теоретический вопрос экзамена.

Экзамен в третьем семестре

Отметка выставляется обучающемуся, который выполнил все практические задания на отметку не ниже «удовлетворительно», выполнил все практические задания, ответил на вопросы по различным заданиям, ответил на устный вопрос преподавателя из списка вопросов по одной из пройденных тем.

Отметка не выставляется обучающемуся, который не выполнил в полном объеме все практические задания.

Экзамен в четвертом семестре

Отметка выставляется обучающемуся, который выполнил все практические задания на отметку не ниже «удовлетворительно», выполнил и защитил групповое задание, ответил на устный вопрос экзаменационного билета и выполнил практическое задание.

Отметка не выставляется обучающемуся, который не выполнил в полном объеме все практические задания и групповое задание.

Примерный перечень вопросов для подготовки к экзамену в 3 и 4 семестре:

Примерный перечень вопросов для подготовки к экзамену в третьем семестре:

1. Сколько поколений компьютеров описывает история?
2. Опишите различия кластерных, грид и облачных вычислений.
3. Каковы основные преимущества и недостатки блейд-систем?

4. Назовите основные преимущества облачных систем хранения данных.
5. Дайте определение облачных вычислений.
6. Какие виды облаков существуют?
7. Расскажите об особенностях публичных, частных, гибридных облаков.
8. Что предоставляют поставщики услуг IaaS?
9. Что скрывается под аббревиатурой PaaS?
10. Что скрывается под аббревиатурой SaaS?
11. Отметьте основные преимущества SaaS для клиентов.
12. Назовите основные преимущества облачных вычислений.
13. Назовите основные недостатки облачных вычислений.
14. Назовите основные преимущества технологии виртуализации.
15. Укажите основные разновидности виртуализации.
16. Назовите основные платформы виртуализации.
17. Технологии NoSQL, их значимость для облачных вычислений.
18. Теорема CAP и ее влияние на технологии NoSQL.
19. NoSQL – основные разновидности NoSQL баз данных.
20. Технология MapReduce.
21. Принципы работы Hadoop.
22. Назовите основные препятствия развитию облачных технологий в России.
23. Расскажите о основных облачных вендорах и их концепциях.
24. Расскажите о основных особенностях AWS (Amazon web Services)
25. Основные преимущества использования Windows Azure.
26. Отметьте основные возможности Google Apps.
27. Проведите сравнительный анализ открытых облачных платформ и проприетарных решений.
28. Вопросы безопасности облаков.
29. Концепции масштабирования, развертывания, резервного копирования в контексте облачной инфраструктуры.
30. Переход от стандартной к облачной инфраструктуре предприятия.
31. Вопросы безопасности, масштабирования, развертывания, резервного копирования в контексте облачной инфраструктуры.
32. Преимущества облачной инфраструктуры в области масштабирования приложений.
33. Особенности аварийного восстановления в облачной среде.
34. Концепция миграции.
35. Выбор подходящей модели развертывания в соответствии с существующими бизнес-задачами.

36. Выбор подходящего поставщика облачных услуг.
37. Концепция SLA.
38. Производительность облачной инфраструктуры.
39. Концепция вендора.
40. Открытые стандарты для обеспечения облачных услуг.
41. Решение проблем перехода: технических, финансовых, безопасности, лицензионных и законодательных.

Примерный перечень вопросов для подготовки к экзамену в 4 семестре:

1. Архитектура облачного приложения.
2. Особенности создания облачных офисов.
3. Понятие «Локального сервера».
4. Разновидности программных средств для разработки ПО.
5. Программная оболочка для разработки сайтов на локальной машине.
6. Принципы разработки пользовательского интерфейса облачного приложения.
7. Этапы создания облачного приложения.
8. Назовите основные преимущества облачных систем хранения данных.
9. Дайте определение облачных вычислений.
10. Какие виды облаков существуют?
11. Расскажите об особенностях публичных, частных, гибридных облаков.
12. Что предоставляют поставщики услуг IaaS?
13. Типы данных.
14. Формулирование технического задания к проекту.
15. Доменное имя сайта, коммерческие и свободные площадки. Хостинг коммерческий и в свободном доступе.
16. Социальный хостинг. Взаимодействие хостинга и доменного имени.
17. DNS.
18. Базовый синтаксис php.
19. Управляющие конструкции php.
20. Инструкции php.
21. Объединение массивов.
22. Передача данных через HTML-формы.
23. Работа с формами в PHP.
24. Использование элементов CheckBox, CheckBoxList, RadioButton.
25. Использование Image, ImageMap, ImageButton.
26. Методы GET и POST. Недостатки и преимущества каждого метода.

27. HTTP-запрос. Внесение данных в БД и извлечение информации.
28. Строки в PHP. Строковые операции. Регулярные выражения. Динамическое включение файлов.
29. Сессии в PHP. Авторизация. Шифрование данных при авторизации.
30. Авторизация через социальные сети.
31. Установка массива cookie. Доступ к cookie. Удаление cookie. Отправка cookie.
32. Процедурный стиль создания скрипта для работы с MySQL.
33. Объектный подход.
34. Подключение к серверу и выбор базы данных. Запросы к базе данных. Чтение
35. Синтаксис XML.
36. Кроссбраузерность. Кроссплатформенность.
37. Адаптивная верстка.
38. Требования к макету.
39. Выбор подходящего поставщика облачных услуг.
40. Производительность облачной инфраструктуры.

Пример экзаменационного билета в третьем семестре:

Экзаменационный билет 1

Теоретический вопрос: Основные модели предоставления услуг облачных вычислений: Software as a Service (SaaS) (ПО-как-услуга).

Практическое задание: Провести аналитический обзор программных средств и решить проблемы перехода (технических, финансовых, безопасности, лицензионных и законодательных) по представленному кейсу. Обосновать свое решение. Результат работы представить в виде отчета (pptx, pdf форматы).

Пример экзаменационного билета в четвёртом семестре:

Экзаменационный билет 1

Теоретический вопрос: Архитектура облачного приложения

Практическое задание: Разработать формы с различными элементами управления и сценарий для передачи данных из формы методами POST, GET

КРИТЕРИИ ОЦЕНИВАНИЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ДЛЯ ФОРМИРОВАНИЯ КОМПЕТЕНЦИИ И КРИТЕРИИ ИХ ОЦЕНИВАНИЯ

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
ПК-1. Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта ПК-1.1. Проводит обследование организаций, выявляет информационные потребности заказчика, используя цифровые инструменты и облачные решения для сбора и анализа данных					
Знать: назначение, суть и способы проведения предпроектного исследования организации облачного приложения	Не знает	Фрагментарные знания о назначении, суть и способы проведения предпроектного исследования организации облачного приложения	Общие, но не структурированные знания о назначении, суть и способы проведения предпроектного исследования организации облачного приложения	Сформированные, но содержащие отдельные пробелы знания о назначении, суть и способы проведения предпроектного исследования организации облачного приложения	Сформированные систематизированные прочные знания о назначении, суть и способы проведения предпроектного исследования организации облачного приложения
Уметь: составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения	Не умеет	Частично освоенные умения составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения	Успешно сформированные, самостоятельно и осознанно выполняемые умения составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения
Владеть: навыками формализованного представления требований к облачному приложению	Не владеет	Фрагментарно сформированные навыки формализованного представления требований к облачному приложению	В целом сформированные, но выполняемые на элементарном уровне навыки формализованного представления требований к облачному приложению	Сформированные, но не устойчивые в сложных ситуациях навыки формализованного представления требований к облачному приложению	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки формализованного представления требований к облачному приложению
ПК-1. Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта ПК-1.2. Разрабатывает техническое задание, формализует требования заказчика к разработке цифрового программного продукта					

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
Знать: основные принципы и порядок разработки технического задания в соответствии с требованиями заказчика	Не знает	Фрагментарны е знания об основных принципах и порядке разработки технического задания в соответствии с требованиями заказчика	Общие, но не структурированны е знания об основных принципах и порядке разработки технического задания в соответствии с требованиями заказчика	Сформированные , но содержащие отдельные пробелы знания об основных принципах и порядке разработки технического задания в соответствии с требованиями заказчика	Сформированные систематизирован ные прочные знания об основных принципах и порядке разработки технического задания в соответствии с требованиями заказчика
Уметь: формулировать и составлять требования входящие в техническое задание на разработку облачного приложения	Не умеет	Частично освоенные умения формулировать и составлять требования входящие в техническое задание на разработку облачного приложения	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения формулировать и составлять требования входящие в техническое задание на разработку облачного приложения	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения формулировать и составлять требования входящие в техническое задание на разработку облачного приложения	Успешно сформированные, самостоятельно и осознанно выполняемые умения формулировать и составлять требования входящие в техническое задание на разработку облачного приложения
Владеть: - навыками оформления технического задания в соответствии с ГОСТ	Не владеет	Фрагментарно сформированн ые навыки документирования требований заказчика и навыки оформления технического задания в соответствии с ГОСТ	В целом сформированные, но выполняемые на элементарном уровне навыки документирования требований заказчика и навыки оформления технического задания в соответствии с ГОСТ	Сформированные , но не устойчивые в сложных ситуациях навыки документирования требований заказчика и навыки оформления технического задания в соответствии с ГОСТ	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки документирования требований заказчика и навыки оформления технического задания в соответствии с ГОСТ
ПК-2. Способен проектировать цифровой программный продукт ПК-2.1. Разрабатывает модель цифрового программного продукта с учетом внедрения передовых ИТ-решений прикладной сферы					
Знать: методологии описания процессов, применяемых в информационной	Не знает	Фрагментарн ые знания о методологиях описания процессов, применяемых	Общие, но не структурированны е знания о методологиях описания процессов,	Сформированные, но содержащие отдельные пробелы знания о методологиях описания	Сформированные систематизирован ные прочные знания о методологиях описания

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
сфере		в информацион ной сфере	применяемых в информационной сфере	процессов, применяемых в информационной сфере	процессов, применяемых в информационной сфере
Уметь: строить концептуальное решение будущего облачного приложения для предметной области	Не умеет	Частично освоенные умения строить концептуальн ое решение будущего облачного приложения для предметной области	В целом освоенные, но не подкрепляемые знаниями и самостоятельност ью выполнения умения строить концептуальное решение будущего облачного приложения для предметной области	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения строить концептуальное решение будущего облачного приложения для предметной области	Успешно сформированные, самостоятельно и осознанно выполняемые умения строить концептуальное решение будущего облачного приложения для предметной области
Владеть: - навыками моделирования прикладных процессов в облачных приложениях	Не владеет	Фрагментарн о сформирован ные навыки документиро вания требований заказчика и навыки моделирован ия прикладных процессов в облачных приложениях	В целом сформированные, но выполняемые на элементарном уровне навыки документировани я требований заказчика и навыки моделирования прикладных процессов в облачных приложениях	Сформированные, но не устойчивые в сложных ситуациях навыки документировани я требований заказчика и навыки моделирования прикладных процессов в облачных приложениях	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки документирования требований заказчика и навыки моделирования прикладных процессов в облачных приложениях
ПК-2. Способен проектировать цифровой программный продукт ПК-2.2. Проектирует базы данных, в том числе используя специализированные цифровые платформы					
Знать: компонентный состав современных баз данных и методологические основы их проектирования с использованием различных цифровых платформ	Не знает	Фрагментарны е знания о компонентном составе современных баз данных и методологическ их основах их проектировани я с использование м различных цифровых платформ	Общие, но не структурированн ые знания о компонентном составе современных баз данных и методологическ их основах их проектирования с использованием различных цифровых платформ	Сформированные, но содержащие отдельные пробелы знания о компонентном составе современных баз данных и методологических основах их проектирования с использованием различных цифровых платформ	Сформированные систематизирован ные прочные знания о компонентном составе современных баз данных и методологических основах их проектирования с использованием различных цифровых платформ
Уметь: разрабатывать эскизный проект облачного	Не умеет	Частично освоенные умения разрабатывать	В целом освоенные, но не подкрепляемые знаниями и	В целом сформированные и самостоятельно выполняемые, но	Успешно сформированные, самостоятельно и осознанно

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
приложения, проектировать модель базы данных		эскизный проект облачного приложения, проектировать модель базы данных	самостоятельностью выполнения умения разрабатывать эскизный проект облачного приложения, проектировать модель базы данных	не всегда интеллектуально обоснованные умения разрабатывать эскизный проект облачного приложения, проектировать модель базы данных	выполняемые умения разрабатывать эскизный проект облачного приложения, проектировать модель базы данных
Владеть: навыками моделирования и описания моделей по заданным требованиям	Не владеет	Фрагментарно сформированные навыки моделирования и описания моделей по заданным требованиям	В целом сформированные, но выполняемые на элементарном уровне навыки моделирования и описания моделей по заданным требованиям	Сформированные, но не устойчивые в сложных ситуациях навыки моделирования и описания моделей по заданным требованиям	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки моделирования и описания моделей по заданным требованиям
ПК-2. Способен проектировать цифровой программный продукт					
ПК-2.3. Проектирует пользовательский интерфейс с учетом тенденций развития цифровых технологий					
Знать: способы и методы проектирования пользовательского интерфейса облачного приложения	Не знает	Фрагментарные знания о способах и методах проектирования пользовательского интерфейса облачного приложения	Общие, но не структурированные знания о способах и методах проектирования пользовательского интерфейса облачного приложения	Сформированные, но содержащие отдельные пробелы знания о способах и методах проектирования пользовательского интерфейса облачного приложения	Сформированные систематизированные прочные знания о способах и методах проектирования пользовательского интерфейса облачного приложения
Уметь: разрабатывать макеты пользовательского интерфейса облачного приложения	Не умеет	Частично освоенные умения разрабатывать макеты пользовательского интерфейса облачного приложения	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью умения разрабатывать макеты пользовательского интерфейса облачного приложения	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения разрабатывать макеты пользовательского интерфейса облачного приложения	Успешно сформированные, самостоятельно и осознанно выполняемые умения разрабатывать макеты пользовательского интерфейса облачного приложения
Владеть: - навыками разработки макетов пользовательского	Не владеет	Фрагментарно сформированные навыки разработки макетов	В целом сформированные, но выполняемые на элементарном уровне навыки	Сформированные, но не устойчивые в сложных ситуациях	Успешно сформированные, устойчивые, технологически грамотно

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
интерфейса облачного приложения		пользовательского интерфейса облачного приложения	разработки макетов пользовательского интерфейса облачного приложения	навыки разработки макетов пользовательского интерфейса облачного приложения	выполняемые навыки разработки макетов пользовательского интерфейса облачного приложения
ПК-2. Способен проектировать цифровой программный продукт ПК-2.4. Проводит анализ современных программно-технических средств, специализированных цифровых платформ, облачных и отраслевых решений для разработки цифрового программного продукта					
Знать: методы и инструменты исследования и анализа жизненного цикла цифрового продукта	Не знает	Фрагментарные знания о методах и инструментах исследования и анализа жизненного цикла цифрового продукта	Общие, но не структурированные знания о методах и инструментах исследования и анализа жизненного цикла цифрового продукта	Сформированные, но содержащие отдельные пробелы знания о методах и инструментах исследования и анализа жизненного цикла цифрового продукта	Сформированные систематизированные прочные знания о методах и инструментах исследования и анализа жизненного цикла цифрового продукта
Уметь: масштабировать, развертывать и делать резервное копирование облачного приложения	Не умеет	Частично освоенные умения масштабировать, развертывать и делать резервное копирование облачного приложения	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения масштабировать, развертывать и делать резервное копирование облачного приложения	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения масштабировать, развертывать и делать резервное копирование облачного приложения	Успешно сформированные, самостоятельно и осознанно выполняемые умения масштабировать, развертывать и делать резервное копирование облачного приложения
Владеть: - навыками сбора необходимой информации облачных приложений	Не владеет	Фрагментарно сформированные навыки сбора необходимой информации облачных приложений	В целом сформированные, но выполняемые на элементарном уровне навыки сбора необходимой информации облачных приложений	Сформированные, но не устойчивые в сложных ситуациях навыки сбора необходимой информации облачных приложений	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки сбора необходимой информации облачных приложений
ПК-3. Способен разрабатывать цифровой программный продукт ПК-3.1. Осуществляет разработку программного кода и компонентов цифрового программного продукта с помощью специализированных цифровых платформ для индивидуальной и совместной разработки					
Знать: - синтаксис современных	Не знает	Фрагментарные знания синтаксиса	Общие, но не структурированные знания	Сформированные, но содержащие отдельные	Сформированные систематизированные прочные

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
языков программирования применяемых в разработке облачных приложений; специализированные цифровые платформы		современных языков программирования применяемых в разработке облачных приложений; специализированных цифровых платформ	синтаксиса современных языков программирования применяемых в разработке облачных приложений; специализированных цифровых платформ	пробелы знания синтаксиса современных языков программирования применяемых в разработке облачных приложений; специализированных цифровых платформ	знания синтаксиса современных языков программирования применяемых в разработке облачных приложений; специализированных цифровых платформ
Уметь: пользоваться приемами облачного программирования	Не умеет	Частично освоенные умения пользоваться приемами облачного программирования	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения пользоваться приемами облачного программирования	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения пользоваться приемами облачного программирования	Успешно сформированные, самостоятельно и осознанно выполняемые умения пользоваться приемами облачного программирования
Владеть: навыками работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений	Не владеет	Фрагментарно сформированные навыки работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений	В целом сформированные, но выполняемые на элементарном уровне навыки работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений	Сформированные, но не устойчивые в сложных ситуациях навыки работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки с работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений
ПК-3. Способен разрабатывать цифровой программный продукт ПК-3.2. Обеспечивает интеграцию программных модулей и компонентов цифрового программного продукта на основе облачных и отраслевых ИТ-решений					
Знать: основные понятия и терминологию облачных технологий; области применения облачных технологий	Не знает	Фрагментарные знания основных понятий и терминологии облачных технологий; областей применения облачных технологий	Общие, но не структурированные знания основных понятий и терминологии облачных технологий; областей применения облачных технологий	Сформированные, но содержащие отдельные пробелы знания основных понятий и терминологии облачных технологий; областей	Сформированные систематизированные прочные знания основных понятий и терминологии облачных технологий; областей применения облачных

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
				применения облачных технологий	технологий
Уметь: применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений	Не умеет	Частично освоенные умения применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений	Успешно сформированные, самостоятельно и осознанно выполняемые умения применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений
Владеть: - навыками работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений	Не владеет	Фрагментарно сформированные навыки работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений	В целом сформированные, но выполняемые на элементарном уровне навыки работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений	Сформированные, но не устойчивые в сложных ситуациях навыки работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений
ПК-5. Способен обеспечивать качество работы цифрового программного продукта ПК-5.2. Проводит юзабилити-аудит цифрового программного продукта					
Знать: концепцию облачных вычислений применительно к бизнес-деятельности; инфраструктуру облачных вычислений	Не знает	Фрагментарные знания концепции облачных вычислений применительно к бизнес-деятельности; инфраструктуры облачных вычислений	Общие, но не структурированные знания концепции облачных вычислений применительно к бизнес-деятельности; инфраструктуры облачных вычислений	Сформированные, но содержащие отдельные пробелы знания концепции облачных вычислений применительно к бизнес-деятельности; инфраструктуры облачных вычислений	Сформированные систематизированные прочные знания концепции облачных вычислений применительно к бизнес-деятельности; инфраструктуры облачных вычислений
Уметь:	Не умеет	Частично	В целом	В целом	Успешно

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений		освоенные умения оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений	освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений	сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений	сформированные, самостоятельно и осознанно выполняемые умения оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений
Владеть: навыками разработки программного обеспечения облачных систем	Не владеет	Фрагментарно сформированные навыки разработки программного обеспечения облачных систем	В целом сформированные, но выполняемые на элементарном уровне навыки разработки программного обеспечения облачных систем	Сформированные, но не устойчивые в сложных ситуациях навыки разработки программного обеспечения облачных систем	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки разработки программного обеспечения облачных систем
ПК-5. Способен обеспечивать качество работы цифрового программного продукта ПК-5.3. Осуществляет работы по интеграционному тестированию цифрового программного продукта с внешним окружением					
Знать: способы тестирования и отладки облачных приложений	Не знает	Фрагментарные знания способов тестирования и отладки облачных приложений	Общие, но не структурированные знания способов тестирования и отладки облачных приложений	Сформированные, но содержащие отдельные пробелы знания способов тестирования и отладки облачных приложений	Сформированные систематизированные прочные знания способов тестирования и отладки облачных приложений
Уметь: осуществлять организационную подготовку тестирования облачного приложения (или отдельных компонентов) представителями заказчика, проведение и фиксацию результатов	Не умеет	Частично освоенные умения осуществлять организационную подготовку тестирования облачного приложения (или отдельных компонентов) представителями заказчика, проведение и	В целом освоенные, но не подкрепляемые знаниями и самостоятельностью выполнения умения осуществлять организационную подготовку тестирования облачного приложения (или отдельных компонентов)	В целом сформированные и самостоятельно выполняемые, но не всегда интеллектуально обоснованные умения осуществлять организационную подготовку тестирования облачного приложения	Успешно сформированные, самостоятельно и осознанно выполняемые умения осуществлять организационную подготовку тестирования облачного приложения (или отдельных компонентов) представителями

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения				
	1	2	3	4	5
		фиксацию результатов	представителями заказчика, проведение и фиксацию результатов	(или отдельных компонентов) представителям и заказчика, проведение и фиксацию результатов	заказчика, проведение и фиксацию результатов
Владеть: навыками системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках	Не владеет	Фрагментарно сформированные навыки системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках	В целом сформированные, но выполняемые на элементарном уровне навыки системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках	Сформированные, но не устойчивые в сложных ситуациях навыки системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках	Успешно сформированные, устойчивые, технологически грамотно выполняемые навыки системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках

КРИТЕРИИ ОЦЕНКИ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

При выставлении итоговой оценки за дисциплину необходимо учитывать результаты текущего контроля и критерии оценивания сформированности компетенций.

Оценка **«отлично»** выставляется обучающемуся, если:

- при ответе на теоретический вопрос были продемонстрированы глубокие, систематизированные и прочные знания (в соответствии с критериями сформированности компетенций), умения логично и обоснованно излагать материал, быстро ориентироваться в дополнительных вопросах и давать полные ответы;
- задание в малых группах выполнено безошибочно и в полном соответствии с требованиями на оценку «отлично», обучающийся в работе продемонстрировал устойчивые навыки работы и осознанно исполняемые действия по построению облачных приложений, составляет код программы и воспроизводит взаимодействие модулей, составленный код программы работает без сбоев, организована проверка на ошибки при вводе данных, предусмотрена защита данных;
- обучающийся участвовал во всех видах текущего контроля и получал по ним максимальные баллы (допустимо отдельные отметки «хорошо»).

Все вышеперечисленное говорит о сформированности запланированных компетенций на высоком уровне.

Оценка **«хорошо»** выставляется обучающемуся, если:

- при ответе на теоретический вопрос было продемонстрировано хорошее, но недостаточно полное изложение материала, с отдельными пробелами в знаниях (возможно, незначительными

неточностями в употреблении терминов), логичное изложение материала, но недостаточно быстрая ориентация в нем при ответе на дополнительные вопросы;

- индивидуальное проектное задание выполнено на оценку «отлично» или «хорошо» (с незначительными замечаниями и/или ошибками), обучающийся продемонстрировал хорошие навыки работы и осознанно исполняемые действия по построению облачных приложений, составляет код программы и воспроизводит взаимодействие модулей, составленный код программы работает без сбоев, организована проверка на ошибки при вводе данных, предусмотрена защита данных;
- обучающийся участвовал во всех видах текущего контроля и получал по ним в основном отметки «хорошо».

Все вышеперечисленное говорит о сформированности запланированных компетенций на хорошем уровне.

Оценка **«удовлетворительно»** выставляется обучающемуся, если:

- при ответе на теоретический вопрос было продемонстрировано поверхностное владение материалом и фрагментарные знания, отсутствие стройного и логично выстроенного ответа, затруднение в употреблении терминов и приведении примеров, плохая ориентация в материале при ответе на дополнительные вопросы;
- индивидуальное проектное задание выполнено на оценку «хорошо» или «удовлетворительно» (с ошибками и замечаниями), с отклонениями от требований, обучающийся затруднялся в объяснении применяемых средств построения облачных приложений, составляет код программы и воспроизводит взаимодействие модулей, код программы работает со сбоями, не организована проверка на ошибки при вводе данных, не предусмотрена защита данных;
- обучающийся участвовал не во всех мероприятиях текущего контроля и получал по ним в основном удовлетворительные отметки.

Все вышеперечисленное говорит о сформированности запланированных компетенций на низком уровне.

Оценка **«неудовлетворительно»** выставляется обучающемуся, если:

- при ответе на теоретический вопрос было продемонстрировано незнание значительного объема материала, «фрагментарность» имеющихся знаний, отсутствие стройного и логично выстроенного ответа, затруднение в употреблении терминов и приведении примеров, неумение выделить главное, сделать выводы и обобщения, неспособность отвечать на дополнительные вопросы;
- индивидуальное проектное задание выполнено на оценку «удовлетворительно» со значительными отклонениями от требований, с грубыми ошибками и множеством замечаний, обучающийся продемонстрировал частично сформированные навыки работы, затрудняется в объяснении применяемых средств по построению облачных приложений, составляет код программы и воспроизводит взаимодействие модулей, составленный код программы не работает, не организована проверка на ошибки при вводе данных, не предусмотрена защита данных;
- обучающийся участвовал не во всех мероприятиях текущего контроля и получал по ним в основном удовлетворительные отметки.

Все вышеперечисленное говорит о частичной сформированности запланированных компетенций.

5. ДИАГНОСТИЧЕСКИЕ ЗАДАНИЯ ДЛЯ ОЦЕНКИ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ, ХАРАКТЕРИЗУЮЩИХ УРОВЕНЬ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Компетенция ПК-1 Способен выявлять информационные потребности пользователей и составлять техническое задание на разработку цифрового программного продукта.

Обучающийся знает: назначение, суть и способы проведения предпроектного исследования организации облачного приложения; основные принципы и порядок разработки технического задания в соответствии с требованиями заказчика; методы обработки информации; формы представления результатов анализа.

Обучающийся умеет: составлять техническую документацию по результатам обследования организации и выявления потребностей пользователей по созданию облачного приложения; формулировать и составлять требования входящие в техническое задание на разработку облачного приложения; осуществлять опросы и анкетирование целевой аудитории и заинтересованных сторон с помощью цифровых инструментов; анализировать и систематизировать разнородные данные, визуализировать результаты анализа.

Обучающийся владеет: навыками формализованного представления требований к облачным приложениям; навыками оформления технического задания в соответствии с ГОСТ; навыками работы с цифровыми и облачными сервисами.

ЗАДАНИЯ ЗАКРЫТОГО ТИПА

1. Какая информация необходима для определения требований к безопасности облачного приложения?
А) Макеты интерфейса
В) Бизнес-процессы
С) Версии используемых операционных систем
D) Цветовая гамма приложения
Правильный ответ: В) Бизнес-процессы
2. Какие информационные потребности могут возникнуть у пользователей облачного приложения?
А) Цветовая тема интерфейса
В) Системные требования
С) Тарифные планы облачного хранилища
D) Обновления софта на компьютере
Правильный ответ: С) Тарифные планы облачного хранилища
3. Что представляет собой предпроектное обследование организации?
А) Разработка концепции нового продукта
В) Оценка текущего состояния бизнес-процессов и инфраструктуры организации
С) Анализ финансовых показателей компании
D) Изучение потребностей рынка
Ответ: В) Оценка текущего состояния бизнес-процессов и инфраструктуры организации
4. Какие методы могут использоваться при проведении предпроектного обследования?
А) Опрос сотрудников
В) Анализ финансовых отчетов
С) Интервью с клиентами
D) Все вышеперечисленные
Ответ: D) Все вышеперечисленные

5. Какие данные можно использовать при проведении информационного обследования организации?
- A) Финансовые отчеты
 - B) Отзывы клиентов
 - C) Статистические данные
 - D) Все вышеперечисленные

Ответ: D) Все вышеперечисленные

6. Что необходимо делать после завершения предпроектного и информационного обследования организации?
- A) Разработать новую стратегию развития
 - B) Провести анализ конкурентов
 - C) Подготовить отчет о результатах обследования
 - D) Начать реализацию проекта

Ответ: C) Подготовить отчет о результатах обследования

7. Какие критерии влияют на выбор пользователя web-приложения для получения информации?
- A) Удобство использования
 - B) Качество контента
 - C) Скорость загрузки
 - D) Все вышеперечисленное

Правильный ответ: D) Все вышеперечисленное

8. На каких устройствах пользователи обычно используют web-приложения для получения информации?
- A) ПК
 - B) Смартфоны
 - C) Планшеты
 - D) Все вышеперечисленное

Правильный ответ: D) Все вышеперечисленное

9. Какие функции web-приложений могут быть полезны для пользователей?
- A) Персонализированные рекомендации
 - B) Уведомления о новостях
 - C) Возможность сохранения контента для просмотра офлайн
 - D) Все вышеперечисленное

Правильный ответ: D) Все вышеперечисленное

10. Какие факторы могут повлиять на удовлетворенность пользователей информацией, полученной из web-приложений?
- A) Актуальность контента
 - B) Легкость навигации
 - C) Качество изображений
 - D) Все вышеперечисленное

Правильный ответ: D) Все вышеперечисленное

ЗАДАНИЯ ОТКРЫТОГО ТИПА

11. Какие основные этапы в создании облачного приложения?

Ответ: Определение целей и требований, выбор облачного провайдера, разработка приложения, тестирование, внедрение и поддержка.

Каковы преимущества использования облачных приложений для бизнеса? Ответ: Гибкость и масштабируемость, снижение затрат на ИТ-инфраструктуру, повышение безопасности данных, удобство работы сотрудников.

12. Какие критерии следует учитывать при выборе облачного провайдера?

Ответ: Надежность и безопасность, стоимость услуг, уровень поддержки, соответствие требованиям по защите персональных данных.

13. Задание: Какие основные цели проведения предпроектного обследования организации?

Ответ: Основные цели проведения предпроектного обследования организации включают определение текущего состояния бизнес-процессов, выявление проблем и недостатков, выработка стратегии развития и определение возможных направлений улучшения работы организации.

14. Задание: Какие этапы включает предпроектное обследование организации?

Ответ: Проведение анализа текущего состояния организации, выявление потенциальных проблем и узких мест в бизнес-процессах, разработка стратегии развития, определение необходимых изменений и подготовка рекомендаций для оптимизации работы организации.

15. Задание: Какие методы и инструменты могут использоваться при информационном обследовании организации?

Ответ: Методы интервьюирования сотрудников и руководителей, анализ статистических данных и отчетов, SWOT-анализ для выявления сильных и слабых сторон организации, а также возможностей и угроз.

16. Задание: Какие выводы могут быть сделаны на основе информационного обследования организации?

Ответ: На основе информационного обследования можно сделать выводы о текущем состоянии организации, выявить ключевые проблемы и узкие места, определить потенциальные возможности для развития и роста, а также сформулировать рекомендации по улучшению работы организации.

17. Задание: Каким образом результаты предпроектного и информационного обследования могут помочь организации?

Ответ: Результаты предпроектного и информационного обследования могут помочь организации определить стратегию развития, выработать план действий для улучшения бизнес-процессов, увеличить эффективность работы и конкурентоспособность, а также повысить уровень удовлетворенности клиентов и сотрудников.

18. Задание: Определите основные информационные потребности пользователей, которые могут возникнуть при использовании веб-приложений для заказа и доставки еды.

Ответ: К основным информационным потребностям пользователей, которые возникают при использовании веб-приложений для заказа и доставки еды можно отнести:

1. Информация о доступных блюдах и их составе.
2. Цены на блюда и напитки.
3. Возможность заказа еды на дом или в офис.
4. Время доставки и стоимость доставки.
5. Возможность оплаты заказа онлайн.
6. Отзывы других клиентов о блюдах и сервисе.
7. Контактный телефон и адрес электронной почты ресторана.

19. Задание: Проанализируйте информационные потребности пользователя при использовании веб-приложения для поиска и бронирования отелей.

Ответ: К основным информационным потребностям пользователей при использовании веб-приложения для поиска и бронирования отелей можно отнести:

1. Расположение отеля на карте.
2. Фотографии номеров и территории отеля.
3. Цены на номера и скидки.
4. Наличие свободных номеров.
5. Отзывы других гостей.
6. Услуги, предоставляемые отелем.
7. Контактная информация отеля.

20. Задание: Разработайте опрос для определения информационных потребностей пользователей при использовании веб-приложения для покупки авиабилетов.

Ответ: опрос для определения информационных потребностей пользователей при использовании веб-приложения для покупки авиабилетов может содержать:

1. Как часто вы планируете покупать авиабилеты?
2. Какие направления вас интересуют?
3. Важна ли для вас стоимость авиабилетов?
4. Нужно ли вам удобное мобильное приложение для покупки авиабилетов?
5. Важно ли для вас наличие онлайн-регистрации на рейс?
6. Нужно ли вам получать уведомления об изменении расписания рейсов?
7. Нужно ли вам иметь возможность бронировать отели через то же приложение, в котором вы покупаете авиабилеты?

21. Задание: Оцените информационные потребности пользователей при использовании веб-приложения для изучения иностранных языков и определите возможные способы их удовлетворения.

Ответ: К основным информационным потребностям пользователей при использовании веб-приложения для изучения иностранных языков и определите возможные способы их удовлетворения можно отнести:

1. Доступ к большому количеству материалов для изучения языка (тексты, аудио, видео).
2. Возможность выбора уровня сложности материала.
3. Наличие тестов и упражнений для закрепления знаний.
4. Возможность общения с носителями языка.
5. Доступ к словарю и грамматическим правилам.
6. Возможность отслеживания прогресса в изучении языка.
7. Наличие мобильной версии приложения.

22. Задание: Опишите, какие основные функции для обеспечения требований по функциональности может включать в себя web-приложение для управления проектами.

Ответ: Список основных функций, которые должны быть включены в web-приложение для управления проектами, может включать в себя:

- ~ создание и управление проектами;
- ~ распределение задач между участниками проекта;
- ~ отслеживание прогресса выполнения задач;
- ~ обмен информацией и документами между участниками проекта;
- ~ анализ результатов проекта и принятие решений на основе данных.

Каждая из этих функций может помочь пользователям в управлении проектами, обеспечивая эффективное планирование, координацию и контроль над выполнением задач.

23. Задание: Какие риски могут возникнуть при разработке web-приложения и какие меры для их минимизации можно предпринять?

Ответ: При разработке web-приложения могут возникнуть следующие риски:

- технические риски, связанные с ошибками в коде, сбоями в работе системы и т. д.;
- риски безопасности, связанные с утечкой данных, несанкционированным доступом и т. п.;
- финансовые риски, связанные с перерасходом бюджета, изменением рыночных условий и т. д.

Для минимизации этих рисков следующие меры:

- провести тщательное тестирование приложения перед запуском;
- обеспечить защиту персональных данных пользователей;
- регулярно обновлять систему безопасности;
- контролировать расходы и следить за изменениями рыночных условий.

24. Какие функциональные требования должно удовлетворять web-приложение?

Ответ: Web-приложение должно обеспечивать возможность регистрации и авторизации пользователей, выполнения определенных действий с данными, обработки запросов и отображения информации в удобной форме.

25. Какие интерфейсные требования должно соответствовать web-приложение?

Ответ: Web-приложение должно иметь интуитивно понятный и привлекательный пользовательский интерфейс, который удовлетворяет требованиям юзабилити и обеспечивает удобство использования.

ПК-2. Способен проектировать цифровой программный продукт

Знать:

- процесс и средства разработки web-приложений;
- факторы сложности разработки web-приложений, технологии, подходы и принципы к исследованию и созданию информационных систем, эргономические требования к ним
- содержание этапов разработки и принципы организации проектирования web-приложений, методики сравнения проектных решений;
- информационную модель данных и ее структуру, логические модели баз данных и их типы, формы нормализации данных в базе;
- принципы разработки пользовательского интерфейса web-приложения;
- методы анализа рынка программно-технических средств, информационных продуктов и услуг;
- основные источники профессионально-ориентированной информации, предназначенной для проведения анализа рынка программно-технических средств;

Уметь:

- определять факторы сложности разработки программных систем, выбирать адекватные технологии для их проектирования
- разрабатывать концептуальную, логическую и физическую модели web-приложений;
- определять необходимый инструментарий по уровню обеспечения информационных систем при разработке web-приложений;
- проектировать пользовательский интерфейс;
- составлять аналитические обзоры рынка программно-технических средств, информационных продуктов и услуг;
- проводить сравнительный анализ программно-технических средств и информационных продуктов в соответствии с полученным заданием;

Владеть:

- навыками разработки архитектуры web-приложения с учетом внедрения сквозных цифровых технологий и передовых ИТ-решений прикладной сферы
- навыками проектирования концептуальной модели БД и ее модификации;
- навыками разработки модели web-приложения на основе проектного решения;
- навыками подготовки и демонстрации прототипа решения;

- навыками проектирования и разработки пользовательского интерфейса web-приложения;
- навыками работы с программно-технологическими средствами (поисковыми машинами, справочно-правовыми и экспертными системами, открытыми информационными ресурсами и т.п.) для выполнения задач исследовательского и аналитического характера

Компетенция ПК-2 Способен проектировать цифровой программный продукт.

Обучающийся знает: методологии описания процессов, применяемых в информационной сфере; компонентный состав современных баз данных и методологические основы их проектирования с использованием различных цифровых платформ; способы и методы проектирования пользовательского интерфейса облачного приложения; методы и инструменты исследования и анализа жизненного цикла цифрового продукта.

Обучающийся умеет: строить концептуальное решение будущего облачного приложения для предметной области; разрабатывать эскизный проект облачного приложения, проектировать модель базы данных; разрабатывать макеты пользовательского интерфейса облачного приложения; масштабировать, развертывать и делать резервное копирование облачного приложения.

Обучающийся владеет: навыками моделирования прикладных процессов в облачных приложениях; навыками моделирования и описания моделей по заданным требованиям; навыками разработки макетов пользовательского интерфейса облачного приложения; навыками сбора необходимой информации облачных приложений.

ЗАДАНИЯ ЗАКРЫТОГО ТИПА

1. Что такое MVC в контексте веб-разработки?

- a) Модульный веб-компонент
- b) Модуль контроля версий
- c) Модель-вид-контроллер
- d) Многоверсионный контейнер

Правильный ответ: c) Модель-вид-контроллер

2. 2. Какая из следующих технологий является клиентской стороной веб-приложения?

- a) Node.js
- b) Express.js
- c) MongoDB
- d) React.js

Правильный ответ: d) React.js

3. Какая роль отвечает за представление данных в MVC архитектуре?

- a) Модель
- b) Вид
- c) Контроллер
- d) Сервер

Правильный ответ: b) Вид

4. Что такое RESTful API?

- a) Программный интерфейс для доступа к базе данных
- b) Модуль управления сетью
- c) Архитектурный стиль для построения веб-сервисов
- d) Контроллер обработки запросов

Правильный ответ: c) Архитектурный стиль для построения веб-сервисов

5. Какие HTTP методы используются для чтения данных в RESTful API?

- a) GET
- b) POST
- c) PUT
- d) DELETE

Правильный ответ: a) GET

6. Какая роль отвечает за бизнес-логику в MVC архитектуре?

- a) Модель
- b) Вид
- c) Контроллер
- d) Хранилище данных

Правильный ответ: a) Модель

7. Какая база данных чаще всего используется в web-приложениях?

- a) SQL Server
- b) NoSQL
- c) SQLite
- d) PostgreSQL

Правильный ответ: b) NoSQL

8. Какой протокол обмена данными часто используется для взаимодействия между клиентом и сервером в современных веб-приложениях?

- a) FTP
- b) HTTP
- c) SMTP
- d) TCP

Правильный ответ: b) HTTP

9. Какая технология позволяет реализовать реактивные интерфейсы в веб-приложениях?

- a) AJAX
- b) WebSocket
- c) GraphQL
- d) REST API

Правильный ответ: b) WebSocket

10. Какой тип архитектуры обычно используется для создания масштабируемых веб-приложений?

- a) Монолитная архитектура
- b) Клиент-серверная архитектура
- c) Микросервисная архитектура
- d) Централизованная архитектура

Правильный ответ: c) Микросервисная архитектура

ЗАДАНИЯ ОТКРЫТОГО ТИПА

11. Какие инструменты можно использовать для разработки облачных приложений?

Ответ: Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform, IBM Cloud.

12. Как обеспечить масштабируемость и быстрый отклик облачного приложения?

Ответ: Использовать микросервисную архитектуру, использовать кэширование данных и CDN, оптимизировать код приложения.

13. Опишите процесс разработки архитектуры веб-приложения.

Ответ: Разработка архитектуры веб-приложения обычно включает в себя несколько этапов. На первом этапе проводится анализ требований, в ходе которого определяются основные функциональные и нефункциональные требования к приложению. Затем происходит проектирование, на этом этапе решается, какие технологии, инструменты и архитектурные шаблоны будут использоваться для реализации приложения. После этого разрабатывается архитектурный дизайн, включающий в себя построение модели данных, определение компонентов приложения и их взаимосвязей. Наконец, на этапе реализации кодируются компоненты приложения согласно разработанной архитектуре.

14. Какие основные преимущества имеет микросервисная архитектура перед монолитной?

Ответ: Микросервисная архитектура предлагает ряд преимуществ перед монолитной архитектурой. Во-первых, микросервисы могут быть независимо развернуты и масштабированы, что облегчает поддержку и управление приложением. Во-вторых, они обеспечивают гибкость и модульность, позволяя быстрее разрабатывать, тестировать и развертывать новые функции. Кроме того, микросервисы позволяют использовать различные технологии и языки программирования для каждого сервиса, что способствует выбору наилучшего инструмента для конкретной задачи.

15. Что такое RESTful API и почему его часто используют в веб-приложениях?

Ответ: RESTful API - это архитектурный стиль для построения веб-сервисов, который основан на принципах REST (Representational State Transfer). Он позволяет организовать взаимодействие между клиентом и сервером с помощью стандартных HTTP методов, таких как GET, POST, PUT и DELETE, и обмен данными в формате JSON или XML. RESTful API широко используется в веб-приложениях из-за своей простоты, гибкости и масштабируемости. Он упрощает разработку и интеграцию различных компонентов приложения, а также обеспечивает четкую структуру запросов и ответов.

16. Какие основные компоненты входят в MVC архитектуру?

Ответ: MVC (Model-View-Controller) - это архитектурный шаблон, который разделяет приложение на три основных компонента: модель (Model), представление (View) и контроллер (Controller). Модель отвечает за бизнес-логику и работу с данными приложения, представление отображает данные пользователю и обрабатывает пользовательский ввод, а контроллер управляет взаимодействием между моделью и представлением, а также обрабатывает запросы от пользователя.

17. Что такое SPA (Single Page Application) и какие преимущества оно предлагает?

Ответ: SPA (Single Page Application) - это тип веб-приложения, которое загружает единственную HTML страницу и динамически обновляет ее содержимое без перезагрузки страницы при взаимодействии пользователя. Основные преимущества SPA включают более быструю загрузку и отзывчивость интерфейса, так как большая часть ресурсов загружается один раз при запуске приложения, а также улучшенный пользовательский опыт благодаря плавным переходам и анимациям.

18. Какие методы аутентификации могут быть использованы в веб-приложениях?

Ответ: В веб-приложениях могут использоваться различные методы аутентификации, включая аутентификацию на основе токенов (например, JWT), аутентификацию с использованием сессий, аутентификацию через социальные сети, а также аутентификацию с помощью сторонних сервисов (например, OAuth). Выбор метода зависит от требований к безопасности, удобства использования и функциональности приложения.

19. Какие инструменты можно использовать для тестирования производительности веб-приложений?

Ответ: Для тестирования производительности веб-приложений можно использовать различные инструменты, включая Apache JMeter, LoadRunner, Gatling, Siege и другие.

20. Задание: Разработайте модель web-приложения для онлайн-магазина. Опишите структуру базы данных, включающую таблицы для хранения информации о продуктах, пользователях, заказах и платежах. Предложите схему взаимодействия клиентской и серверной частей приложения, включая функциональные возможности каждой части.

Решение: Для модели web-приложения онлайн-магазина мы можем разработать следующую структуру базы данных:

Таблица «Продукты» с полями: id продукта, название, описание, цена, количество на складе и т.д.

Таблица «Пользователи» с полями: id пользователя, имя, адрес электронной почты, хэш пароля и т.д.

Таблица «Заказы» с полями: id заказа, id пользователя, дата заказа, статус заказа и т.д.

Таблица «Платежи» с полями: id платежа, id заказа, сумма, дата и т.д.

Взаимодействие клиентской и серверной частей приложения будет осуществляться с помощью RESTful API, где клиент будет отправлять запросы на добавление товаров в корзину, оформление заказа и оплату, а сервер будет обрабатывать эти запросы, взаимодействуя с базой данных.

21. Задание: Представьте модель web-приложения для социальной сети. Опишите структуру базы данных для хранения профилей пользователей, сообщений, фотографий, друзей и лайков.

Обсудите архитектурные решения для реализации новостной ленты, системы уведомлений и поиска пользователей.

Решение: Для модели web-приложения социальной сети мы можем разработать следующую структуру базы данных:

Таблица «Пользователи» с полями: id пользователя, имя, дата рождения, фото профиля и т.д.

Таблица «Сообщения» с полями: id сообщения, id отправителя, id получателя, текст сообщения, дата отправки и т.д.

Таблица «Друзья» с полями: id пользователя, id друга и т.д.

Таблица «Лайки» с полями: id пользователя, id записи, дата лайка и т.д.

Архитектурное решение будет включать в себя разделение приложения на клиентскую и серверную части, где клиентская часть будет отвечать за отображение профилей пользователей, новостной ленты и личных сообщений, а серверная часть будет обрабатывать запросы от клиента и взаимодействовать с базой данных.

22. Задание: Разработайте модель web-приложения для онлайн-обучения. Опишите сущности базы данных, такие как курсы, уроки, студенты и преподаватели. Предложите структуру для организации курсов, включая моделирование уровней доступа и рейтинговой системы для оценки успеваемости студентов.

Решение: Для модели web-приложения онлайн-обучения мы можем разработать следующую структуру базы данных:

Таблица «Курсы» с полями: id курса, название, описание, длительность и т.д.

Таблица «Уроки» с полями: id урока, id курса, название, описание, материалы и т.д.

Таблица «Студенты» с полями: id студента, имя, адрес электронной почты, курсы и т.д.

Таблица «Преподаватели» с полями: id преподавателя, имя, адрес электронной почты, преподаваемые курсы и т.д.

Архитектурное решение будет включать в себя создание интерфейса для студентов и преподавателей, где студенты смогут просматривать курсы, регистрироваться на них и проходить уроки, а преподаватели смогут добавлять новые уроки и оценивать успеваемость студентов.

23. Задание: Представьте модель web-приложения для финансового учета. Опишите базу данных для хранения информации о транзакциях, счетах пользователей, бюджетах и категориях расходов. Рассмотрите алгоритмы для автоматического распознавания и категоризации транзакций.

Решение: Для модели web-приложения финансового учета мы можем разработать следующую структуру базы данных:

Таблица «Транзакции» с полями: id транзакции, id пользователя, сумма, категория, дата и т.д.

Таблица «Счета» с полями: id счета, id пользователя, баланс, валюта и т.д.

Таблица «Категории расходов» с полями: id категории, название, описание и т.д.

Архитектурное решение включает разработку интерфейса для пользователей, позволяющего им добавлять новые транзакции, просматривать и анализировать свои расходы и доходы, а также управлять своими счетами и бюджетами.

24. Задание: Разработайте модель web-приложения для электронной медицинской карты. Обсудите структуру базы данных для хранения информации о пациентах, истории болезней, назначениях и результатах анализов. Рассмотрите механизмы обеспечения безопасности и конфиденциальности данных.

Решение: Для модели web-приложения электронной медицинской карты мы можем разработать следующую структуру базы данных:

Таблица «Пациенты» с полями: id пациента, имя, дата рождения, медицинская история и т.д.

Таблица «Диагнозы» с полями: id диагноза, id пациента, дата постановки, описание и т.д.

Таблица «Назначения» с полями: id назначения, id пациента, дата, описание, лекарства и т.д.

Архитектурное решение будет включать в себя создание защищенного интерфейса для врачей и медицинского персонала, где они смогут просматривать и редактировать медицинские записи пациентов.

25. Задание: Представьте модель web-приложения для планирования путешествий. Опишите структуру базы данных для хранения информации о турах, отелях, бронированиях и отзывах пользователей.

Решение: Для модели web-приложения для планирования путешествий нам необходимо разработать структуру базы данных, которая будет содержать информацию о турах, отелях, бронированиях и отзывах пользователей. Давайте опишем возможную структуру базы данных:

Таблица «Туры»:

tour_id: уникальный идентификатор тура

название: название тура

описание: описание тура

стоимость: стоимость тура

дата_начала: дата начала тура

дата_окончания: дата окончания тура

место_посещения: место, которое будет посещено в рамках тура

тип_тура: тип тура (например, экскурсионный, пляжный и т.д.)

Таблица «Отели»:

hotel_id: уникальный идентификатор отеля

название: название отеля

описание: описание отеля

адрес: адрес отеля

рейтинг: рейтинг отеля

цена_за_ночь: стоимость проживания за ночь

Таблица «Бронирования»:

booking_id: уникальный идентификатор бронирования

tour_id: идентификатор тура, на который осуществляется бронирование

hotel_id: идентификатор отеля, в котором осуществляется бронирование

дата_заселения: дата заселения в отель

дата_выселения: дата выселения из отеля

количество_гостей: количество гостей в брони

статус: статус бронирования (подтверждено, ожидает подтверждения и т.д.)

Таблица «Отзывы»:

review_id: уникальный идентификатор отзыва

user_id: идентификатор пользователя, оставившего отзыв

tour_id: идентификатор тура, к которому относится отзыв

отель_id: идентификатор отеля, к которому относится отзыв

оценка: оценка тура или отеля

текст_отзыва: текст отзыва

Компетенция ПК-3 Способен разрабатывать цифровой программный продукт.

Обучающийся знает: синтаксис современных языков программирования применяемых в разработке облачных приложений; специализированные цифровые платформы; основные понятия и терминологию облачных технологий; области применения облачных технологий.

Обучающийся умеет: пользоваться приемами облачного программирования; применять облачные технологии для интеграции программных модулей и компонентов программного обеспечения для web-приложений.

Обучающийся владеет: навыками работы со специализированными цифровыми платформами, компонентами и средствами, используемыми для разработки облачных приложений, навыками работы с облачными технологиями для интеграции программных модулей и компонентов для облачных приложений.

ЗАДАНИЯ ЗАКРЫТОГО ТИПА

1. Какой язык программирования является стандартом для создания гипертекстовых и гипермедийных документов в интернете?

- A) HTML
- B) JavaScript
- C) PHP
- D) CSS

Ответ: A) HTML

2. Какой из перечисленных языков программирования отвечает за динамическое изменение содержимого и поведения веб-страницы?

- A) HTML
- B) JavaScript
- C) PHP
- D) CSS

Ответ: B) JavaScript

3. Какой язык программирования широко используется для создания серверной части веб-приложений?

- A) HTML
- B) JavaScript

C) PHP
D) CSS
Ответ: C) PHP

4. Для чего используется язык программирования CSS?
A) Для описания структуры HTML-документа
B) Для добавления интерактивности на веб-странице
C) Для определения внешнего вида элементов HTML
D) Для работы с серверной частью веб-приложения
Ответ: C) Для определения внешнего вида элементов HTML

5. Какой из языков программирования используется для стилизации веб-страницы и задания внешнего вида элементов?
A) HTML
B) JavaScript
C) PHP
D) CSS
Ответ: D) CSS

6. Что означает термин "кроссбраузерность" в веб-разработке?
A) Совместимость веб-сайта с различными операционными системами
B) Совместимость веб-сайта с различными браузерами и их версиями
C) Возможность отображения веб-сайта на разных устройствах
D) Возможность взаимодействия веб-сайта с разными базами данных
Ответ: B) Совместимость веб-сайта с различными браузерами и их версиями

7. Что означает термин "кроссплатформенность" в веб-разработке?
A) Возможность отображения веб-сайта на различных устройствах
B) Возможность взаимодействия веб-сайта с разными базами данных
C) Возможность работы веб-сайта на разных операционных системах
D) Возможность создания резервных копий веб-сайта на разных серверах
Ответ: C) Возможность работы веб-сайта на разных операционных системах

8. Что означает термин "адаптивная верстка"?
A) Возможность веб-сайта работать на различных браузерах
B) Возможность веб-сайта работать на различных операционных системах
C) Возможность веб-сайта отображаться корректно на различных устройствах и разрешениях экранов
D) Возможность веб-сайта адаптироваться под разные языки и культуры
Ответ: C) Возможность веб-сайта отображаться корректно на различных устройствах и разрешениях экранов

9. Что такое медиазапросы в CSS?
A) Элементы CSS, которые определяют внешний вид веб-сайта
B) Конструкции CSS, которые позволяют применять стили в зависимости от различных устройств и условий
C) Параметры CSS, определяющие кроссбраузерное поведение веб-сайта
D) Подход в CSS, который обеспечивает кроссплатформенность веб-сайта
Ответ: B) Конструкции CSS, которые позволяют применять стили в зависимости от различных устройств и условий

10. Какие из следующих методов являются основными для обеспечения кроссбраузерности веб-сайта?

А) Использование вендорных префиксов в CSS

В) Использование JavaScript-библиотек для стандартизации поведения в разных браузерах

С) Тестирование и отладка в различных браузерах и их версиях

Д) Все перечисленные методы

Ответ: D) Все перечисленные методы

ЗАДАНИЯ ОТКРЫТОГО ТИПА

1. Напишите JavaScript функцию, которая принимает на вход массив чисел и возвращает сумму всех элементов этого массива.

Решение:

JavaScript функция для вычисления суммы всех элементов массива:

```
```javascript
function sumArray(array) {
 return array.reduce((sum, current) => sum + current, 0);
}
```

// Пример использования:

```
const numbers = [1, 2, 3, 4, 5];
console.log(sumArray(numbers)); // Выведет: 15
...`
```

2. Создайте PHP-скрипт, который принимает на вход POST-запросом два числа и возвращает их сумму.

Решение:

Пример PHP-скрипта для вычисления суммы двух чисел из POST-запроса:

```
```php
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $num1 = $_POST["num1"];
    $num2 = $_POST["num2"];
    $sum = $num1 + $num2;
    echo $sum;
}
?>
...`
```

3. Напишите JavaScript функцию, которая проверяет, является ли строка палиндромом (читается одинаково как с начала, так и с конца) без учета регистра символов и пробелов.

Решение:

JavaScript функция для проверки строки на палиндром:

```
```javascript
```

```
function isPalindrome(str) {
 const cleanStr = str.toLowerCase().replace(/[^a-za-яё]/g, "");
 return cleanStr === cleanStr.split("").reverse().join("");
}
// Пример использования:
console.log(isPalindrome("А роза упала на лапу Азора")); // Выведет: true
...
```

4. Разработайте PHP-скрипт, который принимает на вход GET-параметр "username" и выводит сообщение "Добро пожаловать, [username]!".

Решение:

Пример PHP-скрипта для вывода персонализированного приветствия:

```
```php
<?php
if (isset($_GET["username"])) {
    $username = $_GET["username"];
    echo "Добро пожаловать, $username!";
}
?>
...

```

5. Напишите JavaScript функцию, которая принимает на вход строку и возвращает новую строку, в которой каждое слово начинается с заглавной буквы.

Решение:

JavaScript функция для преобразования каждого слова в строке, чтобы оно начиналось с заглавной буквы:

```
```javascript
function capitalizeWords(str) {
 return str.replace(/\b\w/g, char => char.toUpperCase());
}
// Пример использования:
console.log(capitalizeWords("hello world")); // Выведет: Hello World
...

```

6. Создайте PHP-скрипт, который принимает на вход POST-запросом текст и возвращает количество слов в этом тексте.

Решение:

PHP-скрипт для подсчета количества слов в тексте, переданном POST-запросом:

```
```php
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {

```

```

$text = $_POST["text"];
$wordCount = str_word_count($text);
echo $wordCount;
}
?>
...

```

7. Напишите JavaScript функцию, которая принимает на вход число и возвращает его факториал.

Решение:

JavaScript функция для вычисления факториала числа:

```

```javascript
function factorial(n) {
 if (n === 0 || n === 1) {
 return 1;
 }
 return n * factorial(n - 1);
}
// Пример использования:
console.log(factorial(5)); // Выведет: 120
...

```

8. Разработайте PHP-скрипт, который принимает на вход GET-параметры "width" и "height" и выводит HTML-код для создания прямоугольника с заданными размерами.

Решение:

Пример PHP-скрипта для создания прямоугольника с заданными размерами через GET-параметры:

```

```php
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Rectangle</title>
</head>
<body>
    <?php
    if (isset($_GET["width"]) && isset($_GET["height"])) {
        $width = $_GET["width"];
        $height = $_GET["height"];
    }
}

```

```

        echo "<div style='width: {$width}px; height: {$height}px; background-color: #ccc;'></div>";
    }
?>
</body>
</html>
```

```

9. Напишите JavaScript функцию, которая принимает на вход массив объектов (каждый объект содержит поля "имя" и "возраст") и возвращает массив имен тех объектов, у которых возраст больше 18 лет.

Решение:

JavaScript функция для фильтрации объектов по возрасту и получения массива их имен:

```

```javascript
function getAdultNames(objects) {
    return objects
        .filter(obj => obj.age > 18)
        .map(obj => obj.name);
}
// Пример использования:
const people = [
    { name: 'Alice', age: 25 },
    { name: 'Bob', age: 17 },
    { name: 'Charlie', age: 30 }
];
console.log(getAdultNames(people)); // Выведет: ["Alice", "Charlie"]
```

```

10. Создайте PHP-скрипт, который принимает на вход POST-запросом массив чисел и возвращает их среднее значение.

Решение:

PHP-скрипт для вычисления среднего значения чисел из массива, переданного POST-запросом:

```

```php
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $numbers = $_POST["numbers"];
    $average = array_sum($numbers) / count($numbers);
    echo $average;
}
?>
```

```

...

11. Напишите PHP-скрипт для создания прямоугольника с заданными размерами через GET-параметры:

Решение:

Пример PHP-скрипта для создания прямоугольника с заданными размерами через GET-параметры:

```
```php
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Rectangle</title>
</head>
<body>
  <?php
    if (isset($_GET["width"]) && isset($_GET["height"])) {
      $width = $_GET["width"];
      $height = $_GET["height"];
      echo "<div style='width: {$width}px; height: {$height}px; background-color: #ccc;'></div>";
    } else {
      echo "Параметры width и height не указаны!";
    }
  ?>
</body>
</html>
```
```

12. Напишите HTML-форму для отправки массива чисел на сервер с использованием метода POST:

Решение:

Пример HTML-формы для отправки массива чисел на сервер с использованием метода POST:

```
```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Submit Numbers</title>
</head>
<body>
  <form action="average.php" method="post">
    <label for="numbers">Enter numbers (comma-separated):</label><br>
    <input type="text" id="numbers" name="numbers"><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
...

```

Этот HTML-код создает форму, где пользователь может ввести числа, разделенные запятыми, которые будут отправлены на сервер методом POST.

13. Напишите JavaScript функцию для проверки, является ли переданная строка палиндромом без учета пробелов и регистра символов.

Решение:

```

```javascript
function isPalindromeIgnoringCaseAndSpace(str) {
 const cleanStr = str.toLowerCase().replace(/\s+/g, "");
 return cleanStr === cleanStr.split("").reverse().join("");
}
// Пример использования:
console.log(isPalindromeIgnoringCaseAndSpace("A man a plan a canal Panama")); // Выведет: true
...

```

Эта функция принимает строку, приводит её к нижнему регистру, удаляет все пробелы и затем сравнивает полученную строку с её обратным представлением.

14. Создайте PHP-скрипт, который принимает на вход POST-запросом массив строк и возвращает массив строк, в которых содержится более двух слов.

Решение:

```

```php
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $inputStrings = $_POST["strings"];
  $resultStrings = array_filter($inputStrings, function($str) {
    return str_word_count($str) > 2;
  });
  echo json_encode($resultStrings);
}

```

```
}  
?>  
...
```

Этот скрипт принимает массив строк через POST-запрос, фильтрует строки, оставляя только те, в которых более двух слов, и возвращает результат в формате JSON.

15. Напишите JavaScript функцию для нахождения наибольшего общего делителя (НОД) двух чисел.

Решение:

```
```javascript  
function gcd(a, b) {
 if (b === 0) {
 return a;
 } else {
 return gcd(b, a % b);
 }
}
// Пример использования:
console.log(gcd(48, 18)); // Выведет: 6
...
```

Эта функция использует алгоритм Евклида для нахождения наибольшего общего делителя двух чисел.

**Компетенция ПК-5** Способен обеспечивать качество работы цифрового программного продукта.

**Обучающийся знает:** концепцию облачных вычислений применительно к бизнес-деятельности; инфраструктуру облачных вычислений; способы тестирования и отладки облачных приложений.

**Обучающийся умеет:** оценивать эффективность применения, долгосрочных перспектив, изучение экономики облачных вычислений; осуществлять организационную подготовку тестирования облачного приложения (или отдельных компонентов) представителями заказчика, проведение и фиксацию результатов.

**Обучающийся владеет:** навыком разработки программного обеспечения облачных систем; навыком системного администрирования для разработки, тестирования и сопровождения приложений, развертываемых в облаках.

## ЗАДАНИЯ ЗАКРЫТОГО ТИПА

1. Какой из следующих факторов оказывает наибольшее влияние на пользовательское восприятие скорости загрузки веб-приложения?

- А) Объем данных, передаваемых по сети
- В) Размер изображений на странице
- С) Сложность JavaScript-кода

D) Количество используемых CSS файлов

Ответ: A) Объем данных, передаваемых по сети

2. Что означает термин «респонсивность» в контексте веб-приложений?

A) Возможность приложения адаптироваться под разные типы устройств

B) Быстрый отклик сервера на запросы пользователя

C) Возможность пользователя взаимодействовать с приложением в реальном времени

D) Способность приложения корректно обрабатывать ошибки и исключения

Ответ: A) Возможность приложения адаптироваться под разные типы устройств

3. Какие из следующих практик способствуют улучшению производительности веб-приложения?

A) Минификация и сжатие файлов CSS и JavaScript

B) Добавление максимального количества изображений на страницу

C) Использование наибольшего возможного числа сторонних библиотек и плагинов

D) Все вышеперечисленные практики

Ответ: A) Минификация и сжатие файлов CSS и JavaScript

4. Что такое "оптимизация изображений" в контексте веб-приложений?

A) Процесс выбора самых красивых изображений для использования на веб-страницах

B) Процесс уменьшения размера файлов изображений без значительной потери качества

C) Процесс добавления дополнительных изображений на веб-страницы для улучшения дизайна

D) Процесс оптимизации цветовой гаммы изображений для лучшего отображения на различных устройствах

Ответ: B) Процесс уменьшения размера файлов изображений без значительной потери качества

5. Что такое "кэширование" веб-ресурсов?

A) Процесс сохранения копий веб-страниц на локальном компьютере пользователя

B) Процесс хранения часто используемых данных на сервере приложения

C) Процесс временного хранения ранее загруженных ресурсов на стороне клиента или сервера

D) Процесс сжатия и упаковки файлов ресурсов для более быстрой загрузки

Ответ: C) Процесс временного хранения ранее загруженных ресурсов на стороне клиента или сервера

6. Какая из следующих характеристик наиболее существенна для оценки юзабилити веб-приложения?

A) Графический дизайн и использование цветовой схемы

B) Возможность масштабирования и адаптивности интерфейса

C) Наличие сложных анимаций и переходов между страницами

D) Корректная работа JavaScript-функционала

Ответ: B) Возможность масштабирования и адаптивности интерфейса

7. Какая из перечисленных функциональных особенностей веб-приложения имеет наибольшее влияние на его юзабилити?

A) Возможность работы с базой данных

B) Функционал поиска и фильтрации данных

C) Наличие слайдеров и каруселей для просмотра контента

D) Автоматическое обновление страницы без перезагрузки

Ответ: B) Функционал поиска и фильтрации данных

8. Какой из следующих аспектов веб-приложения необходимо оценивать при аудите юзабилити?

A) Доступность и навигация

- В) Цветовая гамма и графический дизайн
  - С) Скорость загрузки и производительность
  - Д) Все вышеперечисленные аспекты
- Ответ: Д) Все вышеперечисленные аспекты

9. Что означает термин «доступность» в контексте юзабилити веб-приложений?

- А) Возможность приложения быть доступным для пользователей в разных странах
- В) Возможность приложения быть доступным для пользователей с ограниченными возможностями (например, с ограниченным зрением или слухом)
- С) Возможность приложения быть доступным для пользователей через мобильные устройства
- Д) Возможность приложения быть доступным для пользователей с разным уровнем технической подготовки

Ответ: В) Возможность приложения быть доступным для пользователей с ограниченными возможностями (например, с ограниченным зрением или слухом)

10. Какие метрики помогают оценить удобство использования форм на веб-страницах?

- А) Время загрузки страницы и скорость отклика сервера
- В) Количество полей ввода и длина формы
- С) Процент успешно отправленных форм
- Д) Все вышеперечисленные метрики

Ответ: Д) Все вышеперечисленные метрики

## ЗАДАНИЯ ОТКРЫТОГО ТИПА

1. Задание: Опишите процесс тестирования производительности веб-приложения. Какие инструменты и методики могут быть использованы для оценки производительности?

Ответ:

Процесс тестирования производительности веб-приложения включает в себя несколько этапов.

Сначала необходимо определить цели тестирования, например, проверка скорости загрузки страниц, времени отклика сервера, производительности базы данных и т.д.

Далее составляется план тестирования, где определяются критерии оценки производительности и методы сбора данных.

Затем проводится непосредственно тестирование, включающее нагрузочное тестирование, стресс-тестирование, тестирование производительности UI и т.д.

После тестирования анализируются полученные данные, выявляются узкие места и проблемы производительности.

На основе результатов анализа принимаются меры по оптимизации приложения.

В качестве инструментов для оценки производительности веб-приложения могут использоваться JMeter, Apache Benchmark, LoadRunner, WebPageTest, Lighthouse и другие. Методики могут включать в себя сравнение средних значений времени ответа сервера, анализ пропускной способности сети, определение времени загрузки страниц и ресурсов, и т.д.

2. Задание: Объясните, что такое автоматизированное тестирование веб-приложений. Какие инструменты и подходы могут быть использованы для автоматизации тестирования?

Ответ:

Автоматизированное тестирование веб-приложений - это процесс использования специальных программных средств для выполнения тестовых сценариев без необходимости вручную повторять действия тестировщика. Это позволяет повысить эффективность тестирования, ускорить процесс выявления ошибок и уменьшить вероятность человеческих ошибок.

Для автоматизации тестирования веб-приложений могут использоваться различные инструменты и подходы:

Фреймворки для написания автотестов, такие как Selenium WebDriver, Puppeteer, Cypress.io.

Инструменты для тестирования API, например, Postman, SoapUI.

Инструменты для генерации тестовых данных, например, Faker.js.

Средства контроля версий для хранения тестовых сценариев и результатов их выполнения, такие как Git, SVN.

Инструменты для непрерывной интеграции и развертывания, например, Jenkins, GitLab CI/CD.

Использование языков программирования, таких как Python, JavaScript, для написания скриптов автоматизации тестирования.

3. Задание: Какие методы и инструменты могут использоваться для тестирования безопасности веб-приложений?

Ответ:

Тестирование безопасности веб-приложений - это процесс проверки наличия уязвимостей и защищенности приложения от различных видов атак, таких как инъекции, переполнения буфера, утечки информации и другие. Для этого могут использоваться различные методы и инструменты:

Анализ исходного кода приложения с использованием специализированных инструментов, например, SonarQube, ESLint.

Проведение сканирования уязвимостей, например, с помощью Burp Suite, OWASP ZAP.

Тестирование на проникновение, включающее в себя моделирование атак и проверку реакции приложения на них.

Аудит конфигурации серверов и баз данных на предмет наличия уязвимостей.

Обучение персонала безопасности и разработчиков в области лучших практик безопасного программирования и защиты данных.

4. Задание: Какие основные аспекты следует учитывать при тестировании кроссбраузерной совместимости веб-приложения?

Ответ:

При тестировании кроссбраузерной совместимости веб-приложения необходимо учитывать следующие аспекты:

Проверка отображения и работы приложения в различных браузерах (Chrome, Firefox, Safari, Edge) и их версиях.

Тестирование на различных операционных системах (Windows, MacOS, Linux) и устройствах (настольные компьютеры, ноутбуки, планшеты, мобильные устройства).

Проверка корректности отображения и интерактивности пользовательского интерфейса, включая формы, кнопки, меню, анимации и т.д.

Адаптация приложения под разные разрешения экранов и устройства с разной плотностью пикселей.

Проверка работы функциональности приложения, включая AJAX-запросы, анимации, клиентские скрипты, при работе с разными браузерами и устройствами.

5. Задание: Каким образом можно провести тестирование доступности веб-приложения для пользователей с ограниченными возможностями?

Ответ:

Тестирование доступности веб-приложения для пользователей с ограниченными возможностями включает в себя ряд мероприятий:

Проверка соответствия веб-приложения стандартам доступности, таким как WCAG (Web Content Accessibility Guidelines).

Тестирование навигации по приложению с помощью клавиатуры, без использования мыши.

Проверка совместимости с ассистивными технологиями, такими как программы чтения с экрана (screen readers).

Оценка контрастности цветов, используемых на веб-страницах, для обеспечения читаемости для людей с нарушениями зрения.

Тестирование доступности форм и интерактивных элементов управления для пользователей с ограниченными двигательными возможностями.

Проверка использования семантических элементов HTML для правильного описания структуры документа и его контента для программ чтения с экрана.

Аудит доступности изображений, таблиц и мультимедийного контента для пользователей с нарушениями зрения или слуха.

6. Задание: Опишите процесс функционального тестирования веб-приложения. Какие шаги включает в себя функциональное тестирование?

Ответ:

Функциональное тестирование веб-приложения направлено на проверку корректности работы его функциональности согласно требованиям и ожиданиям пользователей. Этот процесс включает в себя следующие шаги:

1. Подготовка тестовых сценариев. На этом этапе определяются основные функциональные возможности приложения, которые подлежат тестированию, и разрабатываются соответствующие тестовые сценарии.

2. Написание тестовых случаев. Для каждого тестового сценария разрабатываются конкретные тестовые случаи, которые определяют шаги тестирования, ожидаемые результаты и критерии успешного выполнения.

3. Выполнение тестов. На этом этапе тестировщики запускают тестовые случаи, выполняют шаги тестирования в соответствии с тестовыми сценариями и записывают результаты выполнения.

4. Анализ результатов. Полученные результаты тестирования анализируются для выявления ошибок, несоответствий требованиям или других проблем функциональности приложения.

5. Отчетность. На основе анализа результатов тестирования разрабатывается отчет, в котором фиксируются выявленные проблемы, предложения по их устранению и общая оценка качества функциональности приложения.

7.Задание: Как можно проверить надежность и стабильность веб- приложения?

Ответ: Для проверки надежности и стабильности веб-приложения можно использовать следующие методы и инструменты:

Мониторинг производительности. Организация постоянного мониторинга работы веб-приложения с целью выявления узких мест, проблем производительности и недоступности.

Тестирование стрессоустойчивости. Проведение тестовых нагрузок, имитирующих высокую нагрузку на сервер и сеть, для оценки реакции приложения на такие условия.

Тестирование восстановления после сбоев. Проверка возможности восстановления приложения после сбоев, включая проверку работы механизмов резервного копирования, восстановления данных и т.д.

Мониторинг ошибок и исключений. Анализ логов приложения и системных регистров для выявления ошибок, исключений и нештатных ситуаций, которые могут привести к сбоям.

Тестирование безопасности. Проверка приложения на наличие уязвимостей и защищенность от атак, таких как инъекции, переполнения буфера, утечки информации и другие.

Регулярное обновление и тестирование. Проведение регулярных обновлений и тестирования новых версий приложения с целью выявления и исправления проблем до их обнаружения пользователями.

8.Задание: Как можно обеспечить высокую степень защиты данных в веб-приложении?

Ответ: Для обеспечения высокой степени защиты данных в веб-приложении необходимо применять следующие меры:

Шифрование данных. Использование шифрования для защиты конфиденциальных данных при передаче по сети и хранении на сервере.

Механизмы аутентификации и авторизации. Реализация механизмов аутентификации пользователей и управления правами доступа для предотвращения несанкционированного доступа к данным.

Обновление системы и патчи безопасности. Регулярное обновление серверного и клиентского программного обеспечения с целью устранения известных уязвимостей и установки последних патчей безопасности.

Мониторинг безопасности. Постоянный мониторинг системы на предмет обнаружения подозрительной активности, атак и попыток несанкционированного доступа.

Резервное копирование данных. Создание регулярных резервных копий данных с целью восстановления информации в случае ее потери или повреждения.

Обучение сотрудников. Проведение обучения персонала по вопросам безопасности информации, осведомление их о потенциальных угрозах и методах защиты.

9.Задание: Как можно оценить качество кода веб-приложения?

Ответ: Качество кода веб-приложения можно оценить с помощью следующих методов:

Анализ кода статическими анализаторами. Использование специализированных инструментов для анализа и проверки исходного кода на соответствие стандартам кодирования, выявление потенциальных уязвимостей и ошибок.

Code review. Организация процесса код-ревью, в ходе которого другие члены команды проверяют код на соответствие стандартам, наличие ошибок и улучшения.

Тестирование кода. Написание и выполнение автоматизированных тестов для проверки корректности работы функциональности приложения, а также тестов на производительность, безопасность и доступность.

Анализ метрик кода. Использование метрик кода, таких как цикломатическая сложность, покрытие тестами, количественные характеристики кода (количество строк, функций и т.д.), для оценки его качества.

Применение лучших практик программирования. Следование принципам SOLID, DRY, KISS и другим лучшим практикам программирования для создания чистого, эффективного и поддерживаемого кода.

10. Задание: Какие могут быть стратегии и методы тестирования веб-приложений для обеспечения их качества?

Ответ: Для обеспечения качества веб-приложений могут применяться различные стратегии и методы тестирования:

Методы тестирования по уровням. Это включает в себя модульное тестирование (unit testing), интеграционное тестирование (integration testing), системное тестирование (system testing) и приемочное тестирование (acceptance testing).

Методы тестирования по типам.

11. Задание: Проверка соединения с базой данных

Решение:

Шаги:

1. Запустите приложение или скрипт, который должен использовать базу данных.

2. Попробуйте установить соединение с базой данных, используя предварительно заданные учетные данные и параметры подключения.

3. Проверьте, было ли соединение успешно установлено.

4. Верните сообщение об успешном соединении или сообщение об ошибке, если соединение не удалось.

Ответ:

- Если соединение установлено успешно, выдаётся сообщение: "Соединение с базой данных установлено успешно."

- Если соединение не удалось установить из-за неверных учетных данных или параметров подключения, выдаётся сообщение: "Ошибка: Не удалось установить соединение с базой данных."

## 12. Задание: Проверка наличия таблицы в базе данных

Решение:

Шаги:

1. Убедитесь, что соединение с базой данных установлено успешно.

2. Выполните запрос к базе данных, чтобы проверить наличие конкретной таблицы.

3. Проверьте результат запроса: наличие таблицы или ошибку, если таблица не существует.

4. Верните сообщение о наличии таблицы или сообщение об ошибке, если таблица отсутствует.

Ответ:

- Если таблица существует, выдаётся сообщение: «Таблица 'название\_таблицы' существует в базе данных.»

- Если таблица не существует, выдаётся сообщение: «Ошибка: Таблица 'название\_таблицы' отсутствует в базе данных.»

## 13.Задание: Проверка количества записей в таблице

Решение:

Шаги:

1. Убедитесь, что соединение с базой данных установлено успешно.

2. Выполните запрос к базе данных для подсчета количества записей в указанной таблице.

3. Проверьте результат запроса: количество записей или ошибку, если таблица не существует.

4. Верните сообщение о количестве записей в таблице или сообщение об ошибке, если таблица отсутствует.

Ответ:

- Если таблица существует и содержит записи, выдаётся сообщение с количеством записей: «Таблица 'название\_таблицы' содержит 'N' записей.»

- Если таблица существует, но не содержит записей, выдаётся сообщение: «Таблица 'название\_таблицы' пуста.»

- Если таблица не существует, выдаётся сообщение об ошибке: «Ошибка: Таблица 'название\_таблицы' отсутствует в базе данных.»

## 14.Задание: Проверка правильности структуры таблицы

Решение:

Шаги:

1. Убедитесь, что соединение с базой данных установлено успешно.

2. Выполните запрос к базе данных для получения описания структуры таблицы.

3. Сравните полученную структуру с ожидаемой схемой таблицы.

4. Верните сообщение о соответствии или несоответствии структуры таблицы ожидаемой схеме.

Ответ:

- Если структура таблицы соответствует ожидаемой схеме, выдаётся сообщение: «Структура таблицы 'название\_таблицы' соответствует ожидаемой схеме.»

- Если структура таблицы не соответствует ожидаемой схеме, выдаётся сообщение с указанием различий.

- Если таблица не существует, выдаётся сообщение об ошибке: «Ошибка: Таблица 'название\_таблицы' отсутствует в базе данных.»

#### 15.Задание: Проверка наличия индексов в таблице

Решение:

Шаги:

1. Убедитесь, что соединение с базой данных установлено успешно.

2. Выполните запрос к базе данных для получения списка индексов в указанной таблице.

3. Проверьте результат запроса: наличие индексов или их отсутствие.

4. Верните сообщение о наличии индексов в таблице или сообщение об их отсутствии.

Ответ:

- Если в таблице существуют индексы, выдаётся сообщение с перечислением индексов: «Таблица 'название\_таблицы' содержит следующие индексы: [индекс1, индекс2, ...]».

- Если в таблице нет индексов, выдаётся сообщение: «Таблица 'название\_таблицы' не содержит индексов.»

- Если таблица не существует, выдаётся сообщение об ошибке: «Ошибка: Таблица 'название\_таблицы' отсутствует в базе данных.»

## 6. ЛИСТ СОГЛАСОВАНИЯ

Составил:

О.И. Подулыбина, ст.преп.



---

(подпись)

Заведующий кафедрой

Н.Б.Стрекалова, д.п.н., доцент



---

(подпись)

Заведующий выпускающей кафедрой

Н.Б.Стрекалова, д.п.н., доцент



---

(подпись)