

**Комплект оценочных материалов по дисциплине
«Архитектура распределенных систем автоматизации»**

Задания закрытого типа

Задания закрытого типа на выбор правильного ответа

Выберите один правильный ответ

1. Что будет выведено при выполнении кода?

```
import java.util.Arrays;
import java.util.List;
public class Main {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(1, 2, 3, 4);
        numbers.replaceAll(n -> n * 2);
        numbers.stream().filter(n -> n % 4 = 0)
                .forEach(System.out::print);
    }
}
```

- A) 248
- Б) 4
- В) 8
- Г) 48

Правильный ответ: Г

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

2. Какое значение вернет показанный ниже метод на языке Java?

```
1 public static int test() {
2     return true ? null : 42;
3 }
```

Ответ:

- A) Код не скомпилируется
- Б) 42
- В) будет брошено исключение `NullPointerException`
- Г) `null`

Правильный ответ: В

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

3. Для описания метода, потенциального способного быть источником `FileNotFoundException` в сигнатуре метода используют следующее описание:

- A) `throws FileNotFoundException`
- Б) `throw FileNotFoundException`
- В) `raise FileNotFoundException`

Правильный ответ: А

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

4. Ключевое слово _____ используется в Java для описания того факта, что класс реализует интерфейс

- A) `extends`
- Б) `implements`
- В) `inherits`
- Г) `derives`

Правильный ответ: Б

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

5. Укажите наиболее популярный вид системы сборки для Spring приложений

- A) Gradle
- Б) Webpack
- В) Maven
- Г) Ant

Правильный ответ: В

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

Выберите несколько правильных ответов

6. Укажите возможные источники хранения описания бинов во фреймворке Spring

- A) Файл `application.properties`
- Б) Java аннотации
- В) YAML файл
- Г) Файл `xml`

Правильный ответ: А, Б, Г

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

7. Укажите корректные способы включения бина в контекст фреймворка Spring

- А) посредством аннотации `@Bean`;
- Б) посредством директивы `#include`
- В) посредством стереотипных аннотаций;
- Г) посредством директивы `#define`
- Д) программно (метод `getBean()`)

Правильный ответ: А, В, Д.

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

8. Перечислите свойства Java класса, чтобы он мог быть отнесен к JavaBeans

- А) Класс должен реализовать методы рефлексии
- Б) Класс должен иметь конструктор без параметров, с модификатором доступа `public`.
- В) Свойства класса должны быть доступны через `get`-еры, `set`-еры.
- Г) Класс должен быть сериализуем.
- Д) Класс должен иметь переопределенные методы `equals()`, `hashCode()` и `toString()`.

Правильный ответ: Б, В, Г, Д.

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

Задания закрытого типа на установление соответствия

Установите правильное соответствие.

Каждому элементу левого столбца соответствует только один элемент правого столбца.

1. Поставьте соответствие каждому определению функционального интерфейса его название в Java.

Определение функционального интерфейса	Название функционального интерфейса
1) Функциональный интерфейс, который принимает один параметр на вход и не возвращает никаких выходных данных	А) Function
2) Функциональный интерфейс, который возвращает значение, а аргументы могут отсутствовать	Б) Predicate
3) Функциональный интерфейс, который принимает аргумент и возвращает логическое значение	В) Consumer
4) Функциональный интерфейс, который принимает один аргумент и возвращает значение	Г) Supplier

Правильный ответ:

1	2	3	4
В	Г	Б	А

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

2. Поставьте соответствие каждому названию параметризации в левой колонке пример параметризации из правой колонки.

- | | |
|--------------------------------|---|
| 1) Параметризованный класс | А) <pre>public interface List<E> { void add(E x); Iterator<E> iterator(); }</pre> |
| 2) Обобщенный класс | Б) <pre>Gen<Integer> iOb;</pre> |
| 3) Параметризованный интерфейс | В) <pre>class Gen<T> { T ob; Gen(T o) { ob = o; } }</pre> |
| 4) Обобщенный интерфейс | Г) <pre>List<String> p=new ArrayList<>()</pre> |

Правильный ответ:

1	2	3	4
Б	В	А	Г

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

3. Укажите соответствие описания области видимости Spring бина и ее краткого названия

- | | |
|--|----------------|
| 1) Возвращает один и тот же экземпляр бина на каждый запрос контейнера Spring IoC. | А) request |
| 2) Возвращает один и тот же экземпляр бина на каждый запрос контейнера Spring IoC. | Б) session |
| 3) Возвращает один и тот же экземпляр бина на каждый запрос контейнера Spring IoC. | В) singleton |
| 4) Создает и возвращает экземпляр бина для каждой HTTP сессии | Г) prototype |
| 5) Бин с данной областью видимости создается один раз для всего сервлет-контекста. Это полезно для данных, которые должны быть общими для всех пользователей и сессий в приложении | Д) Application |

Правильный ответ:

1	2	3	4	5
В	Г	А	Б	Д

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

4. Установите соответствие описания слоев информационной системы и их названий

- | | |
|---|--------------------------|
| 1) Всё, что связано с взаимодействием с пользователем: нажатие кнопок, движение мыши, отрисовка изображения, вывод результатов поиска и т.д | A) Слой бизнес-логики |
| 2) Правила, алгоритмы реакции приложения на действия пользователя или на внутренние события, правила обработки данных | Б) Слой доступа к данным |
| 3) Хранение, выборка, модификация и удаление данных, связанных с решаемой приложением прикладной задачей | В) Слой представления |

Правильный ответ:

1	2	3
В	А	Б

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

5. Установите соответствие описания видов архитектур информационных систем и их названий

- | | |
|--|---------------------|
| 1) Распределённая архитектура, когда слои представления и бизнес-логики находятся на клиенте, а также там реализуется часть вычислений. Сервер отвечает только за хранение и управление файлами | А) Монолитная |
| 2) Обеспечивает многопользовательскую работу с данными, имеет более высокую надёжность, так как на клиенте находится лишь слой представления и часть слоя бизнес-логики | Б) Файл-серверная |
| 3) Если все функциональные возможности ИС реализованы в виде одного, а не нескольких серверных компонентов, и поддерживаются единой базой данных. | В) Микросервисная |
| 4) Если весь набор функциональных возможностей ИС представлен не одним, а несколькими backend'ами, которые взаимодействуют между собой и каждый из них имеет свою ограниченную по контексту область действия | Г) Клиент-серверная |

Правильный ответ:

1	2	3	4
Б	Г	А	В

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

6. Установите соответствие описания видов протоколов взаимодействия микросервисов и примеров таких протоколов

- 1) Асинхронный протокол. Код клиента или отправителя сообщения не ждут ответ — сообщения отправляются аналогично передаче в очередь любого брокера. Такой тип применяет протокол А) HTTP
- 2) Синхронный протокол. При отправке запроса клиент ожидает ответ от службы — задачи выполняются только после того, как сервер пришлет ответ. Б) AMQP

Правильный ответ:

1	2
Б	А

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

Задания закрытого типа на установление правильной последовательности

Установите правильную последовательность. Запишите правильную последовательность букв слева направо.

1. В каком порядке располагаются методы в Java Stream API?

- А) Обработчик
- Б) Сохранение результата
- В) Источник данных

Правильный ответ: В, А, Б

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

2. Расположите прав доступа к полям и методам класса наиболее строгого к наиболее открытому?

- А) private
- Б) public
- В) package-private
- Г) protected

Правильный ответ: А, В, Г, Б

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

3. Задан список `peoples` объектов класса `Person`. В каком порядке будут выведены объекты, если для сортировки используется следующий оператор `peoples.sort((p1, p2) -> p1.getAge() - p2.getAge());`

В каком порядке будут выведены люди со следующими возрастaми?

- А) 25
- Б) 18
- В) 22
- Г) 21

Правильный ответ: Б, Г, В, А

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

4. Расположите этапы жизненного цикла `String` бина в хронологическом порядке

- А) удаление бина;
- Б) Пре-инициализация ;
- В) Установка свойств из конфигурации бина;
- Г) Инициализация;
- Д) Пост-инициализация;
- Е) Инстанцирование объекта.

Правильный ответ: Е, В, Б, Г, Д, А

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

5. Расположите в правильном порядке составные части HTTP запроса.

- А) заголовок запроса (`header fields`) — характеризуют тело сообщения,
- Б) тело сообщения (`body`)

В) строка запроса (request line) — определяет метод, путь к ресурсу и версию протокола

Правильный ответ: В, А, Б.

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

6. Расположите в правильном порядке составные части HTTP ответ.

А) тело сообщения (body)

Б) строка статуса ответа (status line) — кол ответа, версия протокола

В) заголовки ответа (header fields) — характеризуют тело ответа,

Правильный ответ: Б, В, А.

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3)..

Задания открытого типа

Задания открытого типа на дополнение

Напишите пропущенное слово (словосочетание).

1. В языке Java при описании класса тело класса заключается в _____

Правильный ответ: фигурные скобки

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

2. В языке Java при описании класса тело класса заключается в _____

Правильный ответ: фигурные скобки

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

3. Метод _____ в Java трансформирует элементы исходного потока, создавая новые.

Правильный ответ: map

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

4. Файл описания проекта в Maven называется _____

Правильный ответ: pom.xml

Правильный ответ: В

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

5. Бины фреймворк Spring создает в _____ - специальном пространстве имен для управления бинами.

Правильный ответ: контексте

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

6. Java класс, содержащий описание конфигурации Spring должен быть помечен аннотацией _____

Правильный ответ: @configuration (также правильно configuration)

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

7. Классы, созданием экземпляров которых и установкой в них зависимостей управляет контейнер фреймворка Spring называют _____
Правильный ответ: бины (также правильно beans)
Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

Задания открытого типа с кратким свободным ответом

Напишите пропущенное слово (словосочетание).

1. Java интерфейс, содержащий только один абстрактный метод называется _____

Правильный ответ: функциональный интерфейс.

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

2. При использовании ссылок на методы в Java для разделен имени класса или объекта от имени метода используется (-ются) символ(ы) _____

Правильный ответ: :: (два двоеточия) / два двоеточия / ::

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

3. В лямбда функциях в Java для отделения списка параметров от тела функции используется (-ются) символ (ы) _____.

Правильный ответ: -> (тире и знак больше) / -> / тире и знак больше

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

4. Для создания пользовательского интерфейса в Java используется ключевое слово _____.

Правильный ответ: @interface

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

5. REST полностью построен на основе _____ протокола

Правильный ответ: HTTP

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

6. Сервис _____ используется для автоматизации документирования REST API сервисов

Правильный ответ: swagger

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

7. При использовании REST протокола рекомендуется метод _____ использовать для получения ресурса, метод _____ - чтобы создать новый ресурс, метод _____ - для внесения изменений или обновления существующего ресурса, метод _____, чтобы избавиться от ресурса.

Правильный ответ: GET, POST, PUT, DELETE

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

8. _____ это необходимый этап проверка данных переданных пользователем не серверную сторону приложения при помощи HTTP – запроса.

Правильный ответ: валидация (допускается validation)

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

Задания открытого типа с развернутым ответом

1. Напишите компьютерную программу на языке Java, демонстрирующую создание персонажей сказки «Курочка ряба» на основе порождающего паттерна «Фабричный метод».

Задачи:

- проанализировать сюжет художественного произведения
- выявить основных персонажей произведения, которые могут быть представлены как классы
- разработать программную реализацию основных классов
- разработать сервисные классы на основе паттерна «Фабричный метод» для создания экземпляров классов основных персонажей.
- разработать программную реализация основной программы, демонстрирующую создание объектов с использованием паттерна «Фабричный метод»

Время выполнения – 40 мин.

Критерии оценивания:

- наличие в программном коде описаний классов, представляющих персонажей художественного произведения;
- наличие в программном коде описаний классов, реализующих паттерн «фабричный метод»;
- наличие в программном коде фрагментов создания экземпляров классов на основе созданных классов паттерна «фабричный метод»;
- работоспособность представленной программы.

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

2. Напишите компьютерную программу на языке Java, демонстрирующую иерархию персонажей русской народной сказки. В программной реализации должно присутствовать наследование классов.

Задачи:

- проанализировать сюжет художественного произведения
- выбрать художественное произведение;
- выявить основных персонажей произведения, которые могут быть представлены как классы;
- проанализировать иерархические связи между основными классами;

- на основе анализа выявить исходные и производные классы для реализации отношений наследования
 - разработать программную реализацию отношений наследования;
 - разработать программную реализацию основной программы, демонстрирующую создание объектов на основе отношений наследования.
- Время выполнения – 40 мин.

Критерии оценивания:

- наличие исходных классов и производных классов
- наличие в программном коде описаний классов, представляющих персонажей художественного произведения;
- наличие в программном коде описаний отношений наследования классов;
- наличие в программном коде фрагментов создания экземпляров классов;
- работоспособность представленной программы.

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

3. Напишите компьютерную программу на языке Java, демонстрирующую организацию взаимодействия персонажей сказки на основе поведенческого паттерна “Цепочка ответственности”.

Задачи:

- проанализировать сюжет художественного произведения
- выбрать художественное произведение;
- выявить основных персонажей произведения, которые могут быть представлены как классы;
- проанализировать взаимодействие персонажей и его развитие во времени;
- на основе анализа разработать методы выполнения команд
- разработать программную реализацию цепочки классов;
- разработать программную реализацию основной программы, демонстрирующую обработку команд с помощью паттерна «цепочка ответственности»

Время выполнения – 40 мин.

Критерии оценивания:

- наличие описаний классов
- наличие списка объектов для реализации паттерна «цепочка ответственности»;
- работоспособность представленной программы.

Компетенции (индикаторы): ОПК-12 (ОПК-12.1, ОПК-12.2, ОПК-12.3).

4. Напишите компьютерную программу на языке Java, демонстрирующую организацию взаимодействия персонажей сказки на основе поведенческого паттерна “Наблюдатель”.

Задачи:

- проанализировать сюжет художественного произведения
- выбрать художественное произведение;
- выявить основных персонажей произведения, которые могут быть представлены как классы;
- проанализировать взаимодействие персонажей и его развитие во времени;
- на основе анализа разработать методы выполнения команд
- разработать программную реализацию цепочки классов;
- разработать программную реализацию основной программы, демонстрирующую обработку команд с помощью паттерна «цепочка ответственности»

Время выполнения – 40 мин.

Критерии оценивания:

- наличие описаний классов
- наличие списка объектов для реализации паттерна «цепочка ответственности»;
- работоспособность представленной программы.

Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

5. Опишите основные особенности структуры HTTP-запроса

Задачи:

- Проанализировать структуру HTTP-запроса;
- Описать поля строки статуса
- Дать описание строк запроса.

Время выполнения – 20 мин

Критерии оценивания:

- наличие описания структуры HTTP-запроса
- наличие описания строки статуса;
- описание структуры заголовков и тела запроса.

Ожидаемый результат:

HTTP-запрос состоит из нескольких ключевых компонентов, каждый из которых играет важную роль в процессе передачи данных между клиентом и сервером. Вот основные компоненты HTTP-запроса:

1. Метод (Method)

Метод указывает, какое действие клиент хочет совершить над ресурсом.

Основные методы включают:

- GET – получение ресурса.
- POST – создание нового ресурса.

- PUT – обновление существующего ресурса.
- DELETE – удаление ресурса.
- HEAD – аналогичен GET, но без тела ответа.
- OPTIONS – запрашивает доступные методы для данного URL.
- PATCH – частичное изменение ресурса.

2. URL (Uniform Resource Locator)

Это адрес ресурса, к которому отправляется запрос. URL содержит доменное имя сервера, путь до ресурса и параметры запроса (если есть).

Например:

`https://example.com/api/users?id=123`

3. Версия протокола (Protocol Version)

Указывает версию используемого протокола HTTP. Обычно это HTTP/1.1 или HTTP/2.

4. Заголовки (Headers)

Заголовки содержат дополнительную информацию о запросе и клиентах. Они могут включать такие данные, как:

- Host: Домен, к которому обращается клиент.
- User-Agent: Информация о браузере или приложении, отправившем запрос.
- Content-Type: Тип содержимого в теле запроса (например, `application/json`).
- Authorization: Данные аутентификации (например, токены доступа).
- Accept: Форматы данных, которые клиент готов принять.
- Cookie: Cookies, переданные клиенту ранее.

5. Тело запроса (Request Body)

Содержит данные, которые отправляются на сервер. В основном используется при методах POST, PUT и PATCH. Может быть пустым для других методов.

Пример полного HTTP-запроса:

```
POST /api/users HTTP/1.1
Host: example.com
Content-Type: application/json
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

{
  "name": "John Doe",
  "email": "johndoe@example.com"
}
```

Как эти компоненты влияют на обработку запроса сервером?

Метод определяет, какую операцию сервер должен выполнить над ресурсом. Сервер обрабатывает запросы по-разному в зависимости от метода: для получения данных использует GET, а для их изменения – POST, PUT или PATCH.

URL помогает серверу определить конкретный ресурс, с которым необходимо работать. Путь и параметры указывают на то, какой именно ресурс требуется обработать.

Протокол влияет на то, каким образом осуществляется передача данных. Различные версии HTTP имеют свои особенности обработки запросов и ответов.

Заголовки предоставляют дополнительные метаданные, которые помогают серверу правильно интерпретировать запрос. Например, заголовок Content-Type сообщает серверу формат данных в теле запроса, а заголовок Authorization передает данные для проверки прав доступа.

Тело запроса содержит данные, которые должны быть обработаны сервером. Эти данные могут использоваться для создания новых ресурсов (POST), обновления существующих (PUT) или выполнения других операций.

Таким образом, все компоненты HTTP-запроса работают вместе, обеспечивая корректную передачу информации между клиентом и сервером. Компетенции (индикаторы): ПК-1 (ПК-1.1, ПК-1.2, ПК-1.3).

6. Опишите структуру HTTP – ответа

Задачи:

- Проанализировать структуру HTTP-ответа;
- Описать поля строки статуса
- Дать описание строк ответа.

Время выполнения – 20 мин

Критерии оценивания:

- наличие описания структуры HTTP-ответа
- наличие описания строки статуса;
- описание структуры заголовков и тела ответа.

Ожидаемый результат:

HTTP-ответ – это сообщение, которое сервер отправляет клиенту в ответ на HTTP-запрос. Структура HTTP-ответа состоит из нескольких частей, каждая из которых выполняет свою функцию. Рассмотрим основные компоненты HTTP-ответа:

1. Строка статуса (Status Line)

Первая строка ответа, которая включает три части:

- Версия протокола: Версия HTTP, использованная для ответа (обычно HTTP/1.1 или HTTP/2).

- Код состояния: Числовой код, который указывает результат обработки запроса. Наиболее распространенные коды:
 - 200 OK – успешная обработка запроса.
 - 404 Not Found – запрошенный ресурс не найден.
 - 500 Internal Server Error – внутренняя ошибка сервера.
- Фраза состояния: Описание кода состояния на естественном языке (например, «OK», «Not Found»).

Пример строки статуса:
HTTP/1.1 200 OK

2. Заголовки (Headers)

Заголовки содержат дополнительную информацию об ответе и сервере. Некоторые важные заголовки:

- Content-Type: Определяет формат данных в теле ответа (например, text/html, application/json).
- Content-Length: Размер тела ответа в байтах.
- Date: Дата и время отправки ответа.
- Server: Информация о программном обеспечении сервера.
- Set-Cookie: Устанавливает cookie для клиента.
- Cache-Control: Управляет кэшированием ответа.

Пример заголовков:

```
Content-Type: text/html
Date: Mon, 25 Dec 2023 12:34:56 GMT
Server: Apache/2.4.41 (Ubuntu)
Set-Cookie: session_id=abc123; Path=/; HttpOnly
Cache-Control: max-age=3600
```

3. Пустая строка

После заголовков идет пустая строка, которая отделяет заголовки от тела ответа.

4. Тело ответа (Response Body)

Содержит данные, которые были запрошены клиентом. Тело ответа может быть пустым (например, для запросов типа HEAD или некоторых ошибок), либо содержать HTML-документ, JSON-данные, изображение и т.п.

Пример тела ответа:

```
{
  "message": "Success!",
  "data": {
    "id": 123,
    "name": "John Doe"
  }
}
```

Полный пример HTTP-ответа:

```
HTTP/1.1 200 OK
Content-Type: application/json
Date: Mon, 25 Dec 2023 12:34:56 GMT
Server: Apache/2.4.41 (Ubuntu)
Set-Cookie: session_id=abc123; Path=/; HttpOnly
Cache-Control: max-age=3600
```

```
{
  "message": "Success!",
  "data": {
    "id": 123,
    "name": "John Doe"
  }
}
```

Как структура HTTP-ответа влияет на обработку ответа клиентом?

Строка статуса предоставляет клиенту основную информацию о результате обработки запроса. Код состояния и фраза состояния позволяют клиенту понять, был ли запрос успешным или произошла ошибка.

Заголовки передают дополнительную информацию, необходимую для правильной интерпретации тела ответа. Например, заголовок Content-Type сообщает клиенту, в каком формате представлены данные, а заголовок Set-Cookie устанавливает cookies для последующих запросов.

Пустая строка служит разделителем между заголовками и телом ответа, позволяя клиенту легко различать эти две части сообщения.

Тело ответа содержит сами данные, которые были запрошены клиентом. Клиент анализирует тело ответа в соответствии с типом данных, указанным в заголовке Content-Type.

Таким образом, структура HTTP-ответа обеспечивает эффективную и стандартизированную передачу данных между сервером и клиентом, позволяя им обмениваться информацией в понятной форме.

Компетенции (индикаторы): ПК-2 (ПК-2.1, ПК-2.2, ПК-2.3).

Экспертное заключение

Представленный комплект оценочных материалов по дисциплине «Архитектура распределенных систем автоматизации» соответствует требованиям ФГОС ВО.

Предлагаемые оценочные материалы адекватны целям и задачам реализации основной профессиональной образовательной программы по направлению подготовки 15.04.04 Автоматизация технологических процессов и производств.

Виды оценочных средств, включенные в представленный фонд, отвечают основным принципам формирования ФОС.

Разработанные и представленные для экспертизы оценочные материалы рекомендуются к использованию в процессе подготовки обучающихся по указанному направлению.

Председатель учебно-методической
комиссии института компьютерных
систем и информационных
технологий



Ветрова Н. Н.

Лист изменений и дополнений

№ п/п	Виды дополнений и изменений	Дата и номер протокола заседания кафедры (кафедр), на котором были рассмотрены и одобрены изменения и дополнения	Подпись (с расшифровкой) заведующего кафедрой (заведующих кафедрами)
1	В фонд оценочных средств добавлен комплект оценочных материалов	25.02.2025 г., №14	 А.В. Колесников