

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Порохня Андрей Александрович
Должность: И.о. директора института перспективной инженерии
Дата подписания: 18.06.2026 12:13:40
Уникальный программный ключ:
990bea3aeec48c23bd8699e48288f8d1960c600

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

Директор института
перспективной инженерии,
кандидат технических наук,
доцент

Порохня А. А.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

Технологии хранения и обработки больших данных

название дисциплины (модуля)

Направленность (профиль)	09.04.02 Информационные системы и технологии
Профиль	Управление данными
Форма обучения	очная
Год начала обучения	2026
Реализуется в 3 семестре	

Введение

1. Назначение: Оценочные материалы (оценочные средства) предназначены для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся, и представляют собой совокупность контрольно-измерительных материалов и методов их использования для измерения уровня достижения обучающимися установленных результатов обучения.

2. ФОС является приложением к программе дисциплины «Технологии хранения и обработки больших данных»

3. Разработчик Николаев Е.И., доцент департамента цифровых, робототехнических систем и электроники

(ФИО, должность)

4. Проведена экспертиза ФОС.

Члены экспертной группы:

Председатель: Дроздова В. И., профессор департамента цифровых, робототехнических систем и электроники

(ФИО, должность)

Члены комиссии:

Шагрова Г. В., профессор департамента цифровых, робототехнических систем и электроники

(ФИО, должность)

Винокурский Д. Л., доцент департамента цифровых, робототехнических систем и электроники

(ФИО, должность)

Представитель организации-работодателя: Ю. В. Рокотов, директор ООО «КиберСофт»

(ФИО, должность)

Экспертное заключение: Оценочные материалы (оценочные средства) по дисциплине «Технологии хранения и обработки больших данных» составлены в соответствии с требованиями самостоятельно устанавливаемого Федерального государственного образовательного стандарта высшего образования по направлению 09.04.02 «Информационные системы и технологии» и позволяют оценить уровень сформированности компетенции ОПК-5.

5. Срок действия ФОС определяется сроком реализации образовательной программы.

1. Описание показателей и критериев оценивания на различных этапах их формирования, описание шкал оценивания

Уровни сформированности компетенций, индикаторов	Дескрипторы			
	Минимальный уровень не достигнут (Неудовлетворительно) 2 балла	Минимальный уровень (удовлетворительно) 3 балла	Средний уровень (хорошо) 4 балла	Высокий уровень (отлично) 5 баллов
Компетенция: ОПК-5				
Результаты обучения по дисциплине (модулю): Анализирует программное, техническое, организационное обеспечение информационных систем <i>Индикатор:</i> ИД-1ОПК-5	В рамках собеседования в полной степени раскрыто содержание вопроса для собеседования. В рамках коллоквиума даны ответы на все вопросы преподавателя. В рамках защиты лабораторных работ формулирует корректные ответы на вопросы для контроля.	В рамках собеседования в полной степени раскрыто содержание вопроса для собеседования. При ответе на вопросы коллоквиума демонстрирует пробелы в знании отдельных элементов теоретического материала. Не предоставил ответы на отдельные контрольные вопросы лабораторных работ	При собеседовании некорректно отвечает на вопрос. Демонстрирует незнание отдельных вопросов теоретического материала в рамках коллоквиума. Не ответил на большинство контрольных вопросов лабораторных работ.	Не отвечает (или отвечает некорректно) на вопросы собеседования. Не способен ответить по вопросам коллоквиума. Не предоставил ответы на контрольные вопросы лабораторных работ
Результаты обучения по дисциплине (модулю): Модернизирует программное, техническое, организационное обеспечение информационных систем <i>Индикатор:</i> ИД-2ОПК-5	Выполняет корректно учебное задание лабораторных работ. Разрабатывает и документирует приложения в рамках индивидуального задания к лабораторным работам. Самостоятельно проектирует архитектуру приложений в рамках выполнения лабораторных работ	Выполняет корректно учебное задание лабораторных работ. Разрабатывает приложения в рамках индивидуального задания к лабораторным работам. Проектирование архитектуры приложения реализует при помощи преподавателя.	Выполняет корректно учебное задание лабораторных работ. Не способен решать разработки приложения в рамках индивидуального задания к лабораторным работам. Проектирование архитектуры приложения реализует при помощи преподавателя.	Не выполняет учебное задание лабораторных работ. Не способен решать разработки приложения в рамках индивидуального задания к лабораторным работам. Не способен проектировать архитектуру приложений для решения задач.
Результаты обучения по дисциплине (модулю): Разрабатывает программное, техническое, организационное обеспечение	Владеет навыками по документированию приложений. Демонстрирует навыки уверенного использования интегрированных сред разработки.	Владеет навыками по документированию приложений. Демонстрирует навыки использования интегрированных сред разработки.	Не владеет навыками по документированию приложений. Не обладает навыками использования интегрированных сред разработки.	Не владеет навыками по документированию приложений. Не обладает навыками использования

информационных систем <i>Индикатор:</i> ИД-З _{ОПК-5}	Владеет навыками тестирования, отладки и документирования информационных систем.		Владеет навыками тестирования, отладки и документирования информационных систем.	интегрированных сред разработки. Не владеет навыками тестирования, отладки и документирования информационных систем.
---	--	--	--	---

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
		Вопросы для собеседования	
1.		1. Развитие вычислительных технологий и стратегии их реализации 2. Зависимые и независимые витрины данных 3. Инкрементальный подход 4. Стратегии реализации 5. Инструментарий интеллектуального анализа данных 6. Базы данных NoSQL 7. Hadoop. История развития, особенности и инструменты 8. Интернет вещей 9. NoSQL и Big Data 10. История возникновения нереляционных подходов 11. Базовые принципы Big Data 12. Масштабируемость 13. Современное состояние Big Data 14. Правила 5V 15. Применение Big Data 16. Технологии Big Data 17. Последовательное чтение и прямой доступ 18. Модель программирования Big Data 19. Массивно параллельные вычислительные системы (MPP) 20. In-Memory базы данных 21. Системы MapReduce 22. BSP-системы 23. Системы Big Data и системы, основанные на транзакциях 24. Хранилище на основе упорядоченных столбцов 25. Хранилища ключ/значение 26. Документо-ориентированные базы данных 27. Графовые базы данных	ОПК-5

		28. Столбцовые наборы данных и РСУБД 29. Наборы столбцов, встроенные отображения, пары ключ/значение 30. Webtable 31. Архитектура распределенного хранилища HBase 32. Документо-ориентированное хранилище 33. Memory-Mapped файлы 34. Коллекции и индексы MongoDB 35. Горизонтальное масштабирование 36. Хранилище типа ключ/значение в Memcached и Redis 37. Частично согласованные нереляционные базы данных 38. Элементы кластера Hadoop 39. Дистрибутивы и экосистема Hadoop 40. Apache 41. Cloudera 42. Hortonworks 43. MapR 44. Компоненты YARN 45. YARN Framework 46. Приложения YARN 47. Недостатки YARN 48. YARN и MapReduce 49. Процесс взаимодействия с YARN 50. Свойства конфигурации YARN 51. Применения YARN 52. Обработка графов 53. Обработка данных в режиме реального времени 54. Технология BSP 55. Стратегии объединения 56. Соединение типа Map-side 57. Соединение типа Reduce-side 58. Срез данных в reduce-side-соединении 59. Сервер и инструментальные средства MongoDB 60. Области применения MongoDB	
--	--	---	--

		Задания закрытого типа	
2.		Какие СУБД являются реляционными? MS SQL Server MongoDB MySQL PostgreSQL	ОПК-5
3.		Какие СУБД являются документоориентированными? MongoDB MySQL PostgreSQL MS SQL Server	ОПК-5
4.		В каком формате представляются документы в MongoDB? JSON XML XLS HTTP	ОПК-5
5.		Как происходит создание базы данных при использовании консоли MongoDB? база данных создается при вызове функции записи документа (документов), вместе с коллекцией, содержащей документы база данных создается специальной командой; MongoDB не предполагает создания базы данных «на лету» создать базу данных можно только из клиентского приложения на языке Python	ОПК-5
6.		При описании подхода NoSQL используют правило 5V. Какие свойства присутствуют в правиле 5V? скорость (velocity) объем (volume) согласованность (consistency) атомарность (atomicity)	ОПК-5

7.		При описании подхода NoSQL используют правило 5V. Какие свойства присутствуют в правиле 5V? достоверность (veracity) объем (volume) согласованность (consistency) значимость (value)	ОПК-5
8.		Какое утверждение о формате JSON корректно? это текстовый формат это бинарный формат это табличный формат это мультимедия-контент	ОПК-5
9.		Какое утверждение об объекте JSON корректно? это неупорядоченное множество пар имя/значение, заключённое в фигурные скобки { } это неупорядоченное множество пар имя/значение, заключённое в квадратные скобки [] это класс, определяемый на языке Javascript	ОПК-5
10.		Какой из объектов JSON определен корректно? {“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”} /*“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”*/ [“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”] class P{“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”} это класс, определяемый на языке Javascript	ОПК-5
11.		Какой из объектов JSON определен корректно? {“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”} <--“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”--!> [“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”] class P{“name” : “Nik Corner”, “salary” : 2500, “address” : “No data”}	ОПК-5

12.		В консоли MongoDB необходимо выбрать базу users в качестве текущей. Какую команду консоли следует использовать? use users use database users using users select users	ОПК-5
13.		какая команда консоли позволяет разработчику получить справку по функционалу консоли MongoDB? help use select what	ОПК-5
14.		В консоли MongoDB требуется узнать имя текущей используемой базы данных. Какую команду надо использовать для этого? db.getName() db.help() db.use() db.select()	ОПК-5
15.		Какая команда консоли MongoDB позволяет вывести список баз данных? show dbs dbs list list print databases	ОПК-5
16.		Какая команда консоли MongoDB позволяет вывести список коллекций текущей базы данных? show collections print collections collections	ОПК-5

		db.list()	
17.		Какая команда используется для запуска консоли MongoDB? mongo startmongo mongodb start mongodb	ОПК-5
18.		Какая команда используется для импорта данных из внешнего источника в базу данных MongoDB? mongoimport importdata importmongo dataimport import	ОПК-5
19.		Какие стратегии для работы с MongoDB доступны разработчику? работа с MongoDB посредством консоли работа с MongoDB с использованием специального графического приложения Compass работа с MongoDB из текстового редактора	ОПК-5
20.		Необходимо добавить документ в коллекцию persons текущей базы данных. Какая команда является корректной? db.persons.insert ({"name" : "No name"}) db.persons ({"name" : "No name"}) db.persons.create({"name" : "No name"}) db.insert({"name" : "No name"})	ОПК-5
21.		Необходимо добавить один документ в коллекцию persons текущей базы данных. Какие команды являются корректными? db.persons.insert({"name" : "No name"}) db.persons({"name" : "No name"})	ОПК-5

		db.persons.insertOne({“name” : “No name”}) db.insert({“name” : “No name”})	
22.		Необходимо добавить один документ в коллекцию persons текущей базы данных. Какие команды являются корректными? db.persons.insert({“name” : “No name”}) db.persons({“name” : “No name”}) db.persons.set({“name” : “No name”}) db.insert({“name” : “No name”})	ОПК-5
23.		Необходимо добавить несколько документов в коллекцию persons текущей базы данных. Какие команды являются корректными? db.persons.insert ([{“name” : “No name”}, {“name” : “Max”}]) db.persons.select ([{“name” : “No name”}, {“name” : “Max”}]) db.persons.insertMany ([{“name” : “No name”}, {“name” : “Max”}]) db.persons.insert ({“name” : “No name”}, {“name” : “Max”})	ОПК-5
24.		Необходимо выбрать все документы из коллекции zoo текущей базы данных. Какую команду консоли следует использовать? db.zoo.find() db.zoo.select() db.zoo.findAndSelect() db.find()	ОПК-5
25.		Необходимо выбрать все документы из коллекции cats текущей базы данных. Какую команду консоли следует использовать? db.cats.find() db. cats.select() db. cats.findAndSelect() db.find(“cats”)	ОПК-5
26.		Необходимо выбрать все документы из коллекции person текущей базы данных. Какую команду консоли следует использовать?	ОПК-5

		db.person.find() db.person.select() db.person.findAndSelect() db.find()	
27.		Необходимо выбрать один документ из коллекции zoo текущей базы данных, удовлетворяющий критерию поиска. Какую команду консоли следует использовать? db.person.findOne({name: "Tiger"}) db.person.find({name: "Tiger"}) db.person.One({name: "Tiger"}) db.person.selectOne({name: "Tiger"})	ОПК-5
28.		Необходимо выбрать один документ из коллекции zoo текущей базы данных, удовлетворяющий критерию поиска. Какую команду консоли следует использовать? db.person.findOne({age: 30}) db.person.find({age: 30}) db.person.One({age: 30}) db.person.selectOne({age: 30})	
29.		В коллекции goods содержатся документы: {type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} Были выполнены следующие команды: db.goods.insertOne({type: "A", product: "Сыр"}) db.goods.insert ([{name: "No name"}, {type: "B", name: "Духи"}]) db.goods.getcount() Что будет выведено в консоль? 6 Error 2	

		3 1 5 4	
		Задания открытого типа	
30.		Какая библиотека python предназначена для управления наборами данных?	ОПК-5
31.		Какая библиотека python предназначена для выполнения векторно-матричных вычислений, математических вычислений?	ОПК-5
32.		Какая библиотека python предназначена для построения графиков?	ОПК-5
33.		В коллекции city содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.city.find({type: "Town"})?	ОПК-5
34.		В коллекции places содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.places.find({name: "Town"})?	ОПК-5
35.		В коллекции city содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.city.find({type: "Village"})?	ОПК-5
36.		В коллекции city содержатся документы: {type: "Village", name: "Woodburn"} {type: "Town", name: "Bookland"} {type: "Town", name: "Prokland"} Сколько документов вернет запрос db.city.find()?	ОПК-5

37.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос <code>db.goods.find()</code>?	ОПК-5
38.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос <code>db.goods.find({ type : "B" })</code>?	ОПК-5
39.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос <code>db.goods.findOne({ type : "B" })</code>?	ОПК-5
40.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос <code>db.goods.find({ type : "B" }).limit(1)</code>?	ОПК-5

41.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос <code>db.goods.find({ type : "B" }).limit(3)?</code>	ОПК-5
42.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"} {type: "B", name: "Салфетка"} {type: "C", name: "NoName"}</pre> Сколько документов вернет запрос <code>db.goods.find({ type : "B" }).limit(0)?</code>	ОПК-5
43.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"}</pre> Сколько документов вернет запрос <code>db.goods.find().limit(0)?</code>	ОПК-5
44.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"}</pre> Сколько документов вернет запрос <code>db.goods.find({ type : "B" }).limit(2)?</code>	ОПК-5
45.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко"} {type: "B", name: "Спрей"}</pre>	ОПК-5

		Сколько документов вернет запрос <code>db.goods.find({ type : "A" }).limit(2)?</code>	
46.		В коллекции <code>goods</code> содержатся документы: <code>{type: "A", name: "Хлеб"}</code> <code>{type: "A", name: "Молоко"}</code> <code>{type: "B", name: "Спрей"}</code> Сколько документов вернет запрос <code>db.goods.find({ type : "A" }).limit(0)?</code>	ОПК-5
47.		В коллекции <code>goods</code> содержатся документы: <code>{type: "A", name: "Хлеб"}</code> <code>{type: "A", name: "Молоко"}</code> <code>{type: "B", name: "Спрей"}</code> Что вернет запрос <code>db.goods.find({ type : "A" }).count()?</code>	ОПК-5
48.		В коллекции <code>goods</code> содержатся документы: <code>{type: "A", name: "Хлеб"}</code> <code>{type: "A", name: "Молоко"}</code> <code>{type: "B", name: "Спрей"}</code> Что вернет запрос <code>db.goods.find({ type : "B" }).count()?</code>	ОПК-5
49.		В коллекции <code>goods</code> содержатся документы: <code>{type: "A", name: "Хлеб"}</code> <code>{type: "A", name: "Молоко"}</code> <code>{type: "B", name: "Спрей"}</code> Были выполнены следующие команды: <code>db.goods.insertOne({type: "A", product: "Сыр"})</code> <code>db.goods.insert ([{name: "No name"}, {type: "B", name: "Спрей"}])</code> <code>db.goods.find().count()</code> Что будет выведено в консоль?	ОПК-5
50.		В коллекции <code>goods</code> содержатся документы: <code>{type: "A", name: "Хлеб"}</code> <code>{type: "A", name: "Молоко"}</code>	ОПК-5

		<pre>{type: "B", name: "Спрей"}</pre> Были выполнены следующие команды: <pre>db.goods.insertOne({type: "A", product: "Сыр"}) db.goods.insert ([{name: "No name"}, {type: "B", name: "Духи"}]) db.goods.find({ type: "B"}).count()</pre> Что будет выведено в консоль?	
51.		В коллекции goods содержатся документы: <pre>{type: "A", name: "Хлеб"} {type: "A", name: "Молоко", date : "07.2019"} {type: "B", name: "Спрей"}</pre> Были выполнены следующие команды: <pre>db.goods.insertOne({type: "A", product: "Сыр"}) db.goods.insert ([{name: "No name"}, {type: "B", name: "Спрей"}]) db.goods.find({ type: "A"}).count()</pre> Что будет выведено в консоль?	ОПК-5
52.		В коллекции goods содержатся документы: <pre>{ name : "Хлеб" , date : "2020"} {type : "A", name: "Молоко" } {type : "B", name: "Спрей", date : "2019"} {type : "B", name: "Салфетка" , date : "2018"} {type : "C", name: "NoName" , date : "2020"} Сколько документов вернет запрос db.goods.find({ date : "2020" }).limit(0)?</pre>	ОПК-5
53.		В коллекции goods содержатся документы: <pre>{ name : "Хлеб" , date : "2020"} {type : "A", name: "Молоко" } {type : "B", name: "Спрей", date : "2019"} {type : "B", name: "Салфетка" , date : "2020"} {type : "C", name: "NoName" , date : "2020"}</pre>	ОПК-5

		Сколько документов вернет запрос <code>db.goods.find({ date : "2020" }).limit(0)?</code>	
54.		В коллекции <code>goods</code> содержатся документы: <pre>{type: "A", name: "Хлеб"}</pre> <pre>{type: "A", name: "Молоко"}</pre> <pre>{type: "B", name: "Спрей"}</pre> Были выполнены следующие команды: <pre>db.goods.insertOne({type: "A", product: "Сыр"})</pre> <pre>db.goods.insert ([{name: "No name"}, {type: "C", name: "Духи"}])</pre> <pre>db.goods.find({ type: "C"}).count()</pre> Что будет выведено в консоль?	ОПК-5
55.		В коллекции <code>goods</code> содержатся документы: <pre>{type: "A", name: "Хлеб"}</pre> <pre>{type: "A", name: "Молоко"}</pre> <pre>{type: "B", name: "Спрей"}</pre> <pre>{type: "B", name: "Молоко"}</pre> Были выполнены следующие команды: <pre>db.goods.insertOne({type: "A", product: "Сыр"})</pre> <pre>db.goods.insert ([{name: "No name"}, {type: "C", name: "Духи"}])</pre> <pre>db.goods.find({ name: "Молоко"}).count()</pre> Что будет выведено в консоль?	ОПК-5

2. Описание шкалы оценивания

Рейтинговая система оценки не предусмотрена для студентов, обучающихся на образовательных программах уровня высшего образования магистратуры.

3. Критерии оценивания компетенций заданий открытого типа

Оценка «**отлично**» выставляется студенту, если являются верными результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; представленный документ не содержит ошибок и решены все задачи. Компетенция ОПК-5 освоена на высоком уровне.

Оценка «**хорошо**» выставляется студенту, если являются верными результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; отчетный документ содержит незначительные ошибки, не влияющие на общий результат и не представлены отдельные задачи, не влияющие на общий результат. Компетенция ОПК-5 освоена на среднем уровне.

Оценка «**удовлетворительно**» выставляется студенту, если в основном верными являются результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; отчетный документ содержит ошибки, не позволяющие оценить качество разработки; представлены не все данные и задачи решены не в полном объеме. Компетенция ОПК-5 освоена на минимальном уровне.

Оценка «**неудовлетворительно**» выставляется студенту, если являются неверными результаты демонстрации знаний об основных платформах, технологиях и инструментальных программно-аппаратных средствах реализации информационных систем, применения современных технологий для реализации информационных систем, выбора платформ для реализации информационных систем, анализа и выбора инструментальных программно-аппаратных средств для реализации информационных систем; представленный документ содержит много грубых ошибок, не представлены ключевые данные и задачи решены недостаточно полно. Компетенция ОПК-5 не сформирована.

4. Критерии оценивания компетенций заданий закрытого типа

Оценка «**отлично**» выставляется студенту, если он правильно ответил на 88-100% от общего количества заданий. Компетенция ОПК-5 освоена на высоком уровне.

Оценка «**хорошо**» выставляется студенту, если он правильно ответил на 72-87% от общего количества заданий. Компетенция ОПК-5 освоена на среднем уровне.

Оценка «**удовлетворительно**» выставляется студенту, если он правильно ответил на 53-71% от общего количества заданий. Компетенция ОПК-5 освоена на минимальном уровне.

Оценка «**неудовлетворительно**» выставляется студенту, выставляется студенту, если он правильно ответил на менее 53% от общего количества заданий. Компетенция ОПК-5 не сформирована.