

Документ подписан простой электронной подписью Информация о владельце: ФИО: Таскаев Сергей Валерьевич Должность: Ректор Дата подписания: 26.06.2026 10:50:37	МИНОБРНАУКИ РОССИИ Федеральное государственное бюджетное образовательное учреждение высшего образования «Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)	
Уникальный идентификатор средства для промежуточной аттестации по дисциплине "Проектирование и разработка распределенных программных систем" по направлению подготовки (специальности) "09.04.04 Программная инженерия" направленности (профилю) "Разработка программного обеспечения ФГБОУ ВО «ЧелГУ»"		стр. 1

**Фонд оценочных средств для промежуточной аттестации по дисциплине
Проектирование и разработка распределенных программных систем**

Направление подготовки (специальность)

09.04.04 Программная инженерия

Направленность (профиль)

Разработка программного обеспечения

Присваиваемая квалификация (степень)

магистр

Форма обучения

очная форма обучения

Год(ы) набора 2026

Челябинск 2026 г.

**09.04.04 Программная инженерия профиль Разработка программного обеспечения, дисциплина
Проектирование и разработка распределенных программных систем, 2026 год набора, очная форма
обучения**

Фонд оценочных средств дисциплины (модуля) одобрен и рекомендован:

Проректор по учебной работе утверждено 27.02.2026 А.А. Саламатов

Ученым советом института информационных технологий

Протокол заседания № 7 от 26.02.2026

Председатель Ученого совета
института информационных
технологий

согласовано

Ю.В. Петриченко

Заседанием кафедры информационных технологий и экономической информатики

Протокол заседания №7 от 26.02.2026

Заведующий кафедрой

согласовано

С.А. Скрипов

Автор (составитель)

В.А. Мельников

**Структура фондов оценочных средств соответствует приказу ректора ФГБОУ ВО «ЧелГУ» от 27
сентября 2022 № 573-1**



Содержание

1. Паспорт фонда оценочных средств	3
2. Перечень формируемых компетенций	4
3. Содержание оценочных средств по дисциплине	7
3.1. Виды оценочных средств	7
3.2. Содержание оценочных средств	11
4. Порядок проведения и критерии оценивания промежуточной аттестации	15
4.1. Порядок проведения промежуточной аттестации	15
4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств	15
4.3. Результаты промежуточной аттестации и уровни сформированности компетенций	15



1. Паспорт фонда оценочных средств

Направление подготовки: 09.04.04 Программная инженерия

Направленность: Разработка программного обеспечения

Дисциплина: Проектирование и разработка распределенных программных систем.

Семестры: 3

Форма промежуточной аттестации: экзамен

Для оценивания результатов обучения используется балльно-рейтинговая система.



2. Перечень формируемых компетенций

Изучение дисциплины «Проектирование и разработка распределенных программных систем» направлено на формирование компетенций, приведённых в 1.

Таблица 1. Результаты обучения по дисциплине.

Коды компетенции и согласно ФГОС (ОПОП ВО)	Содержание компетенций согласно ФГОС (ОПОП ВО)	Индикаторы достижения компетенции согласно ОПОП	Перечень планируемых результатов обучения по дисциплине
1	2	3	4
ОПК-2	Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	ОПК-2.1. Демонстрирует знание современных интеллектуальные технологии и технологий программирования, инструментальных средств, программно-технических платформ для решения профессиональных задач ОПК-2.2. Демонстрирует умения разрабатывать алгоритмические и программные решения с использованием интеллектуальных технологий и технологий программирования ОПК-2.3. Имеет практический опыт разработки программного обеспечения для решения профессиональных задач	Знать: принципы и шаблоны проектирования распределенных программных систем, программных интерфейсов Уметь: выполнять проектирование компонентов распределенных программных систем по заданным требованиям в рамках определенной предметной области Владеть: навыками разработки технических спецификаций на компоненты распределенных программных систем и интерфейсы
ОПК-5	Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем	ОПК-5.1. Знает виды, архитектуру современного программного и аппаратного обеспечение информационных и автоматизированных систем ОПК-5.2. Демонстрирует умения разрабатывать и модернизировать программное и аппаратное обеспечение исходя из заданных требований к информационным и автоматизированным системам ОПК-5.3. Имеет практический опыт разработки программного и аппаратного обеспечения	Знать: основные принципы и технологии промышленной разработки распределенных программных систем Уметь: разрабатывать распределенные программные системы с применением инструментов автоматизации сборки, интеграции, тестирования и развертывания ПО Владеть: навыками промышленной разработки распределенных программных систем



		информационных и автоматизированных систем для решения профессиональных задач	
ПК-4	Способность проектировать распределенные информационные системы, их компоненты и протоколы их взаимодействия, разрабатывать требования к программному обеспечению, определять цели и ключевые сценарии для архитектуры программного обеспечения; обосновывать выбор технологий и средств разработки программного обеспечения	ПК-4.1. Демонстрирует знание архитектуры и принципов проектирования распределенных систем, компонентов и интерфейсов, методов сбора и анализа требований к ИС ПК-4.2. Демонстрирует умения определять требования к разработке и сценарии использования ИС, выполнять проектирование компонентов распределенных информационных систем ПК-4.3. Имеет практический опыт разработки технических спецификаций на компоненты распределенных программных систем и протоколы взаимодействия	Знать: понятия архитектуры ПО, основные архитектурные стили и принципы проектирования ПО основные способы организации взаимодействия между модулями приложения, их особенности, назначение и границы применимости Уметь: проектировать архитектуру распределенных программных систем и распределенных хранилищ данных исходя из бизнес-проблем и проектных ограничений анализировать потребности заказчика и пользователей, формулировать требования к архитектуре и функциональности системе, анализировать и формализовывать требования Владеть: навыками разработки архитектурных спецификаций для создаваемых распределенных программных систем; обоснованно применять на практике архитектурные принципы и концептуальные модели взаимодействия и организации вычислений в распределенных системах
ПК-3	Способность проводить анализ требований к архитектуре программного обеспечения, осуществлять выбор и моделирование архитектуры единой информационной системы, осуществлять документирование программного обеспечения, контролировать реализацию и тестирование программного обеспечения	ПК-3.1. Демонстрирует знание основных особенностей процесса проектирования программных систем, типы черт программных систем (поведенческие, структурные), классификацию моделей UML ПК-3.2. Демонстрирует умения выделять функциональные требования к разрабатываемой системе, определять поведенческие и структурные черты проектируемого ПО, строить модели проектируемого продукта с	Знать: основы тестирования и методы оценки качества распределенных программных систем Уметь: проводить тестирование, определять метрики качества распределенных программных систем Владеть: навыками решения задач обеспечения качества распределенных программных систем



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю) «Проектирование и разработка распределенных программных систем» по направлению подготовки 09.04.04 «Программная инженерия» направленности «Разработка программного обеспечения» ФГБОУ ВО «ЧелГУ»

стр. 6

		помощью различного типа диаграмм UML ПК-3.3. Имеет практический опыт проектирования структуры программных систем, навык анализа предметной области, спецификации поведенческих и структурных черт разрабатываемой информационной системы	
--	--	--	--



3. Содержание оценочных средств по дисциплине

3.1. Виды оценочных средств

Таблица 2. Виды оценочных средств.

№ п/п	Код компетенции/ планируемые результаты обучения	Контролируемые темы/ разделы	Наименование оценочного средства для текущего контроля	Наименование оценочного средства на промежуточной аттестации/№ задания
1	ОПК-2.1. Демонстрирует знание современных интеллектуальные технологии и технологий программирования, инструментальных средств, программно-технических платформ для решения профессиональных задач Знать: принципы и шаблоны проектирования распределенных программных систем, программных интерфейсов	Введение в распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
2	ОПК-2.2. Демонстрирует умения разрабатывать алгоритмические и программные решения с использованием интеллектуальных технологий и технологий программирования Уметь: выполнять проектирование компонентов распределенных программных систем по заданным требованиям в рамках определенной предметной области	Введение в распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
3	ОПК-2.3. Имеет практический опыт разработки программного обеспечения для решения профессиональных задач Владеть: навыками разработки технических спецификаций на компоненты распределенных программных систем и интерфейсы	Введение в распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем	Практические задания	Задания теста № 1-18 Практические задания 1-4



		Концепция REST сервисов. Графовый API		
4	ОПК-5.1. Знает виды, архитектуру современного программного и аппаратного обеспечение информационных и автоматизированных систем Знать: основные принципы и технологии промышленной разработки распределенных программных систем	Введение в распределенные программные системы Протоколы распределенных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
5	ОПК-5.2. Демонстрирует умения разрабатывать и модернизировать программное и аппаратное обеспечение исходя из заданных требований к информационным и автоматизированным системам Уметь: разрабатывать распределенные программные системы с применением инструментов автоматизации сборки, интеграции, тестирования и развертывания ПО	Введение в распределенные программные системы Протоколы распределенных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
6	ОПК-5.3. Имеет практический опыт разработки программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач Владеть: навыками промышленной разработки распределенных программных систем	Введение в распределенные программные системы Протоколы распределенных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
7	ПК-4.1. Демонстрирует знание архитектуры и принципов проектирования распределенных систем, компонентов и интерфейсов,	Введение в распределенные программные системы Протоколы распределенных	Практические задания	Задания теста № 1-18 Практические задания 1-4



	методов сбора и анализа требований к ИС Знать: понятия архитектуры ПО, основные архитектурные стили и принципы проектирования ПО основные способы организации взаимодействия между модулями приложения, их особенности, назначение и границы применимости	программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API		
8	ПК-4.2. Демонстрирует умения определять требования к разработке и сценарии использования ИС, выполнять проектирование компонентов распределенных информационных систем Уметь: проектировать архитектуру распределенных программных систем и распределенных хранилищ данных исходя из бизнес-проблем и проектных ограничений анализировать потребности заказчика и пользователей, формулировать требования к архитектуре и функциональности системе, анализировать и формализовывать требования	Введение в распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
9	ПК-4.3. Имеет практический опыт разработки технических спецификаций на компоненты распределенных программных систем и протоколы взаимодействия Владеть: навыками разработки архитектурных спецификаций для создаваемых распределенных программных систем; обоснованно применять на практике архитектурные принципы и концептуальные модели взаимодействия и организации вычислений в распределенных системах	Введение в распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
10	ПК-3.1. Демонстрирует	Введение в	Практические задания	Задания теста № 1-18



	знание основных особенностей процесса проектирования программных систем, типы черт программных систем (поведенческие, структурные), классификацию моделей UML Знать: основы тестирования и методы оценки качества распределенных программных систем	распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API		Практические задания 1-4
11	ПК-3.2. Демонстрирует умения выделять функциональные требования к разрабатываемой системе, определять поведенческие и структурные черты проектируемого ПО, строить модели проектируемого продукта с помощью различного типа диаграмм UML Уметь: проводить тестирование, определять метрики качества распределенных программных систем	Введение в распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4
12	ПК-3.3. Имеет практический опыт проектирования структуры программных систем, навык анализа предметной области, спецификации поведенческих и структурных черт разрабатываемой информационной системы Владеть: навыками решения задач обеспечения качества распределенных программных систем	Введение в распределенные программные системы Протоколы распределенных программных систем Клиент-серверная концепция. Удаленный вызов процедур и методов. Очереди вычислений Сервис-ориентированная архитектура распределенных вычислительных систем Концепция REST сервисов. Графовый API	Практические задания	Задания теста № 1-18 Практические задания 1-4

Типовые задания, критерии и показатели оценивания в рамках текущего контроля представлены в рабочей программе дисциплины (модуля). Полные комплекты оценочных средств и контрольно-измерительных материалов хранятся на кафедре.



3.2. Содержание оценочных средств

База тестовых вопросов

№ п/п	Формулировка вопроса	Варианты ответов
1.	Укажите наиболее децентрализованный тип РАСПРЕДЕЛЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ из представленных ниже видов вычислительных систем:	A) Кластерные системы B) Peer-to-peer системы C) Рабочая станция D) NUMA
2.	Система, в которой каждый компонент взаимодействует с другими компонентами исключительно на основе информации об их интерфейсе, без привязки к особенностям их внутренней реализации, называется...	A) слабосвязанной системой B) объектно-ориентированной системой C) клиент-серверной системой D) сильносвязанной системой E) одноранговой системой F) распределенной системой
3.	Отметьте систему, которая обеспечивает преобразование доменного имени сервера в IP-адрес.	A) DNS B) NTP C) SNMP D) DHCP E) XMPP
4.	Распределенная система обмена сообщениями, разработанная корпорацией LinkedIn в 2011 г. для организации связи внутренних подсистем при переходе платформы от монолитной инфраструктуры к децентрализованной называлась...	A) Kafka B) RabbitMQ C) Redis D) CouchDB E) AMQP
5.	При использовании API сообщений имя предоставляемого сервиса определяется...	A) Сервером, на основе анализа семантики переданного сообщения B) Сервером, на основе имени процедуры, указанном в запросе клиента C) Сервером, на основе использованного при обращении глагола протокола HTTP и идентификатора ресурса D) Сервером, посредством анализа WSDL-сообщения клиента E) Сервером, на основе анализа последовательности и типов аргументов RPC-вызова
6.	При использовании REST API имя предоставляемого сервиса определяется...	A) Сервером, на основе анализа семантики переданного сообщения B) Сервером, на основе имени процедуры, указанном в запросе клиента C) Сервером, на основе использованного при обращении глагола протокола HTTP и идентификатора ресурса D) Сервером, посредством анализа WSDL-сообщения клиента E) Сервером, на основе анализа последовательности и типов аргументов



		RPC-вызова
7.	При использовании RPC API имя предоставляемого сервиса определяется...	А) Сервером, на основе анализа семантики переданного сообщения В) Сервером, на основе имени процедуры, указанном в запросе клиента С) Сервером, на основе использованного при обращении глагола протокола HTTP и идентификатора ресурса D) Сервером, посредством анализа WSDL-сообщения клиента Е) Сервером, на основе анализа последовательности и типов аргументов RPC-вызова
8.	При использовании графового API, сообщение содержит...	А) Структурированный запрос на получение либо модификацию объектов и их конкретных полей В) Стандартный HTTP метод, URI определяющий ресурс и данные для запроса С) Идентификатор события, задачи или темы и структурированное содержимое сообщения D) WSDL-сообщение Е) Имя процедуры и параметры ее вызова
9.	Отметьте вид вычислительных систем, обеспечивающих наибольшую степень распределенности и децентрализации управления.	А) Грид В) Облачные системы С) NUMA D) Кластер Е) P2P
10.	Основной отличительной особенностью технологии _удаленного вызова методов_ от технологии _удаленного вызова процедур_ является:	А) Явная работа (чтение и модификация) с состоянием удаленного объекта. В) Взаимодействие с удаленными сущностями посредством сетевого интерфейса, а не посредством механизма вызова локальных процедур. С) Передача параметров по значению, а не по ссылке. D) Использование механизма сокетов для установления соединения. Е) Отличительных особенностей нет, т.к. данные термины являются синонимами.
11.	Клиент, реализующий уровень пользовательского интерфейса и уровень бизнес-логики называется...	А) толстым клиентом В) тонким клиентом С) очередью сообщений D) терминалом Е) браузером
12.	Закончите	А) сервис-ориентированная



	определение: "Парадигма организации и использования распределенных функциональных возможностей, которые могут предоставляться различными владельцами - это ..."	архитектура В) облачные вычисления С) вычислительные кластера D) тонкие клиенты
13.	Примером использования трехзвенной клиент-серверной архитектуры является	A) подключение по удаленному рабочему столу к удаленной вычислительной системе B) получение файла с удаленного сервера по протоколу FTP C) осуществление покупки в интернет-магазине, реализованным на PHP D) внесение информации о поставщике в удаленную базу данных посредством SQL-запроса
14.	Укажите наиболее ДЕЦЕНТРАЛИЗОВАННЫЙ тип распределенных вычислительных систем из представленных ниже видов вычислительных систем:	A) Кластер B) Облачные вычислительные системы C) Векторные вычислительные системы D) SMP
15.	Отметьте механизм, использующийся для организации асинхронного обмена сообщениями между независимыми процессами при разработке прикладных распределенных вычислительных систем.	A) очереди B) сигналы C) семафоры D) протоколы E) функторы
16.	Абстрактный объект, предложенный в BSD Unix, и определяемый в рамках протокола TCP/IP комбинацией IP-адреса и номера порта называется ...	A) сокетом B) веб-сервисом C) удаленной процедурой D) удаленным объектом E) REST-службой
17.	Отметьте, какая концепция предоставления вычислительных ресурсов основывалась на методологии обеспечения доступности и использования вычислительной мощности способом, аналогичным предоставлению воды и электроэнергии в современных коммунальных сетях.	A) грид-вычисления B) NUMA C) потоковые вычисления D) одноранговые сети E) Gnutella
18.	Разработчик веб-сервиса решил предоставить информацию об API сервиса. В нем указано следующее: Сервис предоставляет доступ к следующим конечным точкам: /USERS/{ID} * GET - предоставить информацию о пользователе * PUT - обновить информацию о пользователе * DELETE – удалить информацию о пользователе /USERS * POST - создать нового пользователя Выберите стиль, в котором данный API веб-сервиса разработан.	A) REST API B) Графовый API C) API Сообщений D) RPC API

Практические задания:

№ п/п	Формулировка задания
-------	----------------------



1.	На языке высокого уровня (Java, C#, Python и др. – на выбор обучающегося) реализовать сетевое клиент-серверное приложение – чат (в виде консольного либо диалогового приложения) на основе технологии сокетов. Клиентская программа предоставляет пользователю интерфейс ввода имени пользователя и сообщений, отправляет сообщения серверу и получает от сервера сообщения от остальных пользователей. Соответственно, серверная программа принимает от пользователей сообщения и рассылает их по всем участникам чата, с указанием имени пользователя, который отправил сообщение.
2.	На языке высокого уровня (Java, C#, Python и др. – на выбор обучающегося) реализовать два сетевых клиент-серверных приложения: 1. «Сортировщик слов». Клиентская программа предоставляет пользователю интерфейс ввода текстового блока, отправляет его серверу, получает и отображает пользователю все слова, представленные в тексте в алфавитном порядке без повторов. Соответственно, серверная часть приложения принимает текстовый блок, разбивает его на слова, формирует перечень использованных слов в алфавитном порядке и возвращает клиенту информацию в виде списка слов (каждое слово – с новой строки). 2. «Доска объявлений». Если клиент передает команду «LIST», то ему выводятся все объявления (текстовые строки), добавленные пользователями на доску объявлений через «;». Если клиент передает другую текстовую строку S, она сохраняется во внешнем файле (или базе данных) и пользователю возвращается подтверждение формата: Message added: “S”. Если клиент вводит пустую строку, то соединение разрывается.
3.	На языке высокого уровня (Java, C#, Python, Go и др. – на выбор обучающегося) реализовать gRPC веб-сервис и клиента для него, которые бы обеспечивали функционирование социальной сети. Сервис должен предоставлять API для просмотра ленты сообщений, в которую пользователи могут загружать сообщения из своих клиентов. Другие клиенты могут «лайкать» чужие посты и оставлять к ним комментарии.
4.	На языке высокого уровня (Java, C#, Python, Go и др. – на выбор обучающегося) реализовать REST веб-сервис, принимающий на обработку CSV-файлы и реализующий их асинхронную (отложенную) обработку. Отложенное выполнение задач необходимо для того, чтобы обеспечить быстрый отклик в веб-приложениях и должно применяться во всех потенциально длительных операциях (посылка писем, обработка файлов и т.д.) Веб-сервис должен предоставлять веб-API, обеспечивающий загрузку в сервис таблицы в виде CSV-файла для ее обработки и получения результата в виде N элементов из данного файла с максимальным значением значения данных по определенному полю данного CSV-файла (TOP-N). Первая строка CSV-файла содержит заголовки соответствующих столбцов таблицы.



4. Порядок проведения и критерии оценивания промежуточной аттестации

4.1. Порядок проведения промежуточной аттестации

Экзамен проводится в виде тестирования. Студент должен ответить на вопросы закрытого типа, которые предполагают выбор вариантов ответа. Всего 20 тестовых вопросов. Продолжительность теста – 35 минут.

4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств

4.2.1. Критерии оценивания теста

Тест формируется в системе электронного обучения MOODLE. Максимальный балл за тест — 100 баллов.

Оценка	Отлично/ Зачтено	Хорошо/ зачтено	Удовлетворительно/зачтено	Неудовлетворительно/ незачтено
Баллы	100-86 баллов	85-76 баллов	75-60 баллов	59-0 баллов
Уровень освоения проверяемых компетенций	высокий	средний	базовый	недостаточный

4.2.2. Критерии оценивания практического задания:

15 баллов: задание полностью выполнено без ошибок

1-14 баллов: задание выполнено частично или выполнено с ошибками, которые были исправлены студентом через некоторое время (2 попытка сдачи работы)

0 баллов: задание не выполнено

4.3. Результаты промежуточной аттестации и уровни сформированности компетенций

На экзамене происходит оценивание учебной деятельности обучающихся по дисциплине на основе полученных оценок за контрольно-рейтинговые мероприятия текущего контроля и промежуточной аттестации. При оценивании результатов учебной деятельности обучающегося по дисциплине используется балльно-рейтинговая система оценивания результатов учебной деятельности обучающихся:

Отлично/зачтено: Величина рейтинга обучающегося по дисциплине 85...100 %

Хорошо/зачтено: Величина рейтинга обучающегося по дисциплине 75...84 %

Удовлетворительно/зачтено: Величина рейтинга обучающегося по дисциплине 60...74 %

Неудовлетворительно/не зачтено: Величина рейтинга обучающегося по дисциплине 0...59 %.

Допускается выставление оценки на основе текущего рейтинга (автоматом)

Особенности проведения процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья обозначены в рабочей программе дисциплины (модуля).



Уровни сформированности компетенций определяется следующим образом:

1. Высокий уровень сформированности компетенций соответствует оценке зачтено:

- предполагает формирование компетенций на высоком уровне;
- знание теоретических разделов изучаемой дисциплины на уровне не ниже оценки

удовлетворительно;

- студент умеет применять на практике знания, полученные в рамках изучения дисциплины

- формируются навыки использования теоретических и практических разделов дисциплины для решения задач профессиональной деятельности;

2. Низкий уровень соответствует оценке незачтено.