

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Порохня Андрей Алексеевич

Должность: И.о. директора института перспективной инженерии

Дата подписания: 19.06.2026 16:38:42

Уникальный программный ключ:

990bea3aeec848c23bd8699e48288f8d1960c600

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

УТВЕРЖДАЮ

И.о. директора института

А.А. Порохня

Ф.И.О.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Проектирование архитектуры программных систем

Направление подготовки
Направленность (профиль)

09.03.04 Программная инженерия
Инженерия сложных программных
систем

Год начала обучения

2026

Форма обучения

очная

Реализуется в семестре

6

Разработано

Заведующая межинститутской кафедрой
департамента цифровых, робототехнических
систем и электроники

(должность разработчика)

Новикова Елена Николаевна

Ф.И.О.

Ставрополь 2026 г.

1. Цель и задачи освоения дисциплины (модуля)

Цели дисциплины: Сформировать у студентов системную компетенцию в области проектирования, оценки и эволюции архитектуры программных систем среднего и крупного масштаба, позволяющую:

- анализировать нефункциональные требования и связывать их с архитектурными стилями;
- применять современные архитектурные подходы (Clean Architecture, DDD, CQRS/Event Sourcing, микросервисы, EDA);
- проводить архитектурную оценку (ATAM) и управлять архитектурным техническим долгом;
- выполнять миграцию между архитектурными стилями с использованием паттернов (Strangler, Feature Toggles, версионирование API).

Задачи освоения дисциплины:

1. Освоить метод FURPS+ и сценарии качества для вывода архитектурных требований (производительность, масштабируемость, отказоустойчивость, безопасность).
2. Научиться применять архитектурный метод анализа компромиссов (ATAM) для выявления рисков и обоснования решений.
3. Сформировать умение выбирать между монолитом, микросервисами, serverless и событийно-ориентированной архитектурой на основе количественных и качественных критериев.
4. Освоить трансформацию из PCMEF (изученного в 5 семестре) в Clean Architecture с обеспечением независимости от фреймворков и внешних деталей.
5. Научиться выделять ограниченные контексты, проектировать агрегаты, value objects, доменные события и репозитории.
6. Освоить разделение командной и запросной моделей, хранение событий, построение проекций и восстановление состояния.
7. Овладеть паттернами постепенной замены модулей (Strangler), управлением функциональными флагами (Feature Toggles) и версионированием API.
8. Научиться реализовывать паттерн Saga (хореография и оркестрация) с компенсирующими транзакциями, понимать компромиссы CAP-теоремы.
9. Управление архитектурным техническим долгом. Сформировать навыки инвентаризации, оценки стоимости и приоритизации архитектурных проблем, а также планирования рефакторингов.
10. Документирование и защита архитектурных решений

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Проектирование архитектуры программных систем» относится к части, формируемой участниками образовательных отношений.

3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесённых с планируемыми результатами освоения образовательной программы

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
-------------------------------	------------------------------	---

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
ПК-2 Способен выполнять проектирование и разработку архитектуры программных систем среднего и крупного масштаба с применением современных паттернов и методов проектирования	ИД-1пк-2: Выделяет и анализирует функциональные и нефункциональные требования (FURPS+), выбирает архитектурный стиль (монолит, микросервисы, EDA, Clean Architecture) на основе сценариев качества	Выделяет и анализирует нефункциональные требования с применением метода FURPS+, формулирует сценарии качества (стимул → источник → среда → ответ → мера) и на их основе выбирает архитектурный стиль (монолит, микросервисы, событийно-ориентированный, Clean Architecture) для программной системы среднего и крупного масштаба.
	ИД-2пк-2: Документирует архитектуру с использованием нотаций C4 и UML, фиксирует архитектурные решения в формате ADR (Architecture Decision Record)	Документирует архитектуру программной системы с использованием нотаций C4 (уровни Context, Containers, Components) и UML, а также фиксирует ключевые архитектурные решения в формате ADR (Architecture Decision Record) с описанием контекста, последствий и рассмотренных альтернатив.
	ИД-3пк-2: Применяет порождающие, структурные и поведенческие паттерны GoF, а также архитектурные паттерны (CQRS, Event Sourcing, Saga, Circuit Breaker)	Применяет порождающие, структурные и поведенческие паттерны GoF, а также архитектурные паттерны (Clean Architecture, CQRS, Event Sourcing, Saga (хореография и оркестрация), Circuit Breaker) при проектировании и реализации архитектуры программных систем среднего и крупного масштаба.

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
	ИД-4пк-2: Реализует многослойную архитектуру (Clean Architecture, Hexagonal, PCMEF), обеспечивая независимость от фреймворков через Dependency Inversion	Реализует многослойную архитектуру (Clean Architecture, Hexagonal, PCMEF), обеспечивая независимость бизнес-логики от фреймворков и инфраструктуры через применение принципа инверсии зависимостей (Dependency Inversion) и паттерна «Порт и адаптер».
ПК-6 Владеет методами работы с данными и информационными системами: способен проектировать базы данных, выполнять анализ и обработку данных, интегрировать ПО с внешними системами	ИД-1пк-6: Проектирует реляционные схемы (нормализация до 3НФ) и нереляционные структуры данных, управляет миграциями БД (Liquibase, Flyway)	Проектирует реляционные схемы баз данных с нормализацией до 3НФ и нереляционные структуры данных (документно-ориентированные, колоночные), а также управляет миграциями схем баз данных с использованием инструментов Liquibase или Flyway.
	ИД-2пк-6: Выполняет интеграцию с внешними API, брокерами сообщений (Kafka, RabbitMQ) и legacy-системами, обеспечивая идемпотентность и отказоустойчивость	Выполняет интеграцию программного обеспечения с внешними API (REST/gRPC), брокерами сообщений (Kafka, RabbitMQ) и legacy-системами, обеспечивая идемпотентность обработки и отказоустойчивость взаимодействия.
	ИД-3пк-6: Строит ETL/ELT-пайплайны	Строит ETL/ELT-пайплайны обработки данных с использованием SQL, Pandas,

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
	обработки данных с использованием SQL, Pandas, Spark, Airflow, реализует проверки качества данных (Great Expectations)	Apache Spark и Apache Airflow, а также реализует проверки качества данных (полнота, уникальность, валидность) с применением Great Expectations

4. Объем учебной дисциплины и формы контроля *

Объем занятий: всего: 4 з.е. 144 ак.ч.	ОФО, в ак. часах
Контактная работа:	54
Лекции/из них практическая подготовка	18
Лабораторных работ/из них практическая подготовка	36
Практических занятий/из них практическая подготовка	
Самостоятельная работа	90
Формы контроля	
Экзамен	
Зачет	
Зачет с оценкой	±
Расчетно-графические работы	
Курсовые работы	
Контрольные работы	

* Дисциплина (модуль) предусматривает применение электронного обучения, дистанционных образовательных технологий

5. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием количества часов и видов занятий

№	Раздел (тема) дисциплины и краткое содержание	Формируемые компетенции, индикаторы	очная форма			
			Контактная работа обучающихся с преподавателем /из них в форме практической подготовки, часов			Самостоятельная работа, часов
			Лекции	Практические занятия	Лабораторные работы	
1	Тема 1. Понятие архитектуры ПО. Нефункциональные требования как драйвер архитектуры. Определение архитектуры ПО. Роль архитектора. Атрибуты качества: производительность, масштабируемость, отказоустойчивость, безопасность, поддерживаемость, тестируемость. Метод FURPS+ (Functional, Usability, Reliability, Performance, Supportability). Сценарии качества: стимул → источник → среда → ответ → мера. Практические примеры: как требования к производительности влияют на выбор между монолитом и микросервисами.	ИД-1пк-2, ИД-1пк-6	2		4	10

2	Тема 2. Оценка архитектуры: АТАМ и архитектурные компромиссы Метод АТАМ: 4 этапа (представление архитектуры, выявление подходов, генерация сценариев, анализ рисков). Сценарии атрибутов качества для производительности, масштабируемости, безопасности. Архитектурные компромиссы: консистентность vs доступность (CAP), производительность vs безопасность, простота vs гибкость. Анализ рисков: идентификация, оценка вероятности и последствий, смягчение. Ролевая игра: архитектор vs стейкхолдеры.	ИД-1пк-2, ИД-3пк-2	2		4	10
3	Тема 3. Сравнительный анализ архитектурных стилей: достоинства, недостатки и критерии выбора Классификация архитектурных стилей: монолитный, микросервисный, событийно-ориентированный, serverless. Детальный разбор каждого стиля: характеристики, достоинства, недостатки, типичные сценарии использования. Таблица сравнения: сложность разработки, масштабируемость, отказоустойчивость, ТСО, время выхода на рынок. CAP-теорема и её влияние на выбор стиля. Практические критерии выбора: размер команды, требования к масштабируемости, бюджет, требования к latency.	ИД-1пк-2, ИД-2пк-6	2		4	10

4	Тема 4. Clean Architecture vs PCMEF: сравнение, независимость от фреймворков и адаптация под разные траектории Clean Architecture: слои Entities, Use Cases, Interface Adapters, Frameworks/Drivers. PCMEF: слои Presentation, Control, Model, Entity, Foundation. Сравнительная таблица: количество слоёв, направление зависимостей, тестируемость, независимость. Dependency Inversion Principle: модули верхних уровней не зависят от модулей нижних уровней. Паттерн «Порт и адаптер» (Hexagonal Architecture) как родственная концепция. Как Clean Architecture упрощает смену БД, UI, внешних сервисов.	ИД-3пк-2, ИД-4пк-2	2		4	10
5	Тема 5. Domain-Driven Design: стратегический и тактический дизайн Стратегический DDD: выделение контекстов, построение контекстной карты. Паттерны взаимодействия: Shared Kernel, Customer-Supplier, Conformist, Anti-Corruption Layer, Open Host Service, Published Language. Тактический DDD: Entity (идентичность), Value Object (неизменяемый), Aggregate (транзакционная граница), Aggregate Root. Domain Events: публикация, обработка, идемпотентность. Репозитории и фабрики для работы с агрегатами. Границы агрегатов: что должно быть внутри одной транзакции. связан с Clean Architecture и микросервисами?	ИД-2пк-2, ИД-3пк-2, ИД-1пк-6	2		4	10

6	Тема 6. CQRS и Event Sourcing: разделение Command/Query, хранение событий, проекции CQRS: Command side (изменение состояния) и Query side (чтение). Пример: диспетчерский дашборд (сложные чтения) vs создание заказа (бизнес-логика). Event Sourcing: Event Store, сохранение событий, восстановление состояния. Снапшоты (snapshots) для ускорения восстановления. Проекции: синхронные и асинхронные, перестроение (rebuild). Компромиссы: eventual consistency, сложность, идиempотентность.	ИД-3пк-2, ИД-1пк-6, ИД-3пк-6	2		4	10
7	Тема 7. Архитектура распределённых транзакций: Saga (хореография и оркестрация) Распределённые транзакции: проблема. Saga: последовательность локальных транзакций с компенсациями. Хореография: децентрализованное управление через события. Оркестрация: централизованный оркестратор (Saga Orchestrator). Компенсирующие транзакции: примеры (отмена заказа, возврат средств). 2PC: блокирующие операции, снижение доступности. CAP-теорема: consistency, availability, partition tolerance. Выбор на практике.	ИД-3пк-2, ИД-2пк-6	2		4	10
8	Тема 8. Архитектурная эволюция и миграции: Strangler pattern, Feature Toggles, версионирование API Strangler pattern: reverse проху, поэтапное вытеснение, ACL. Parallel run: сравнение результатов работы старой и новой версий. Feature Toggles: типы (release toggle, experiment toggle, ops toggle), процентное включение (canary), конфигурация. Версионирование API: URI path, custom header, content negotiation, media type. Обратная совместимость: добавление полей, необязательные параметры, депрекация, sunset. План миграции: фазы, риски, компромиссы.	ИД-2пк-2, ИД-4пк-2, ИД-2пк-6	2		4	10

9	Тема 9. Архитектурный технический долг и рефакторинг Архитектурный технический долг: определение, виды (намеренный/ненамеренный, стратегический/тактический). Признаки: циклические зависимости, нарушение слоёв, дублирование логики, God Object, отсутствие интерфейсов, жёсткие связи. Инструменты выявления: анализ кода, метрики (SonarQube, JDepend, PlantUML для анализа зависимостей). Оценка: время исправления, стоимость владения, влияние на команду, риск инцидентов. Приоритизация: $\text{severity} \times \text{cost of delay} / \text{effort}$. Стратегии: рефакторинг (постоянный), выделение микросервисов (Strangler), отказ от устаревших модулей.	ИД-4пк-2, ИД-1пк-6				
			2		4	10
	Зачет с оценкой					10
	ИТОГО за 6 семестр		18		36	90
	ИТОГО		18		36	58

6. Фонд оценочных средств по дисциплине (модулю)

Фонд оценочных средств (ФОС) по дисциплине (модулю) базируется на перечне осваиваемых компетенций с указанием индикаторов. ФОС обеспечивает объективный контроль достижения запланированных результатов обучения. ФОС включает в себя:

- описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;

- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций (включаются в методические указания по тем видам работ, которые предусмотрены учебным планом и предусматривают оценку сформированности компетенций);

- типовые оценочные средства, необходимые для оценки знаний, умений и уровня сформированности компетенций.

ФОС является приложением к данной программе дисциплины.

7. Методические указания для обучающихся по освоению дисциплины

Приступая к работе, каждый студент должен принимать во внимание следующие положения.

Дисциплина построена по тематическому принципу, каждая тема представляет собой логически завершённый раздел.

Практические занятия проводятся с целью закрепления усвоенной информации, приобретения навыков ее применения при решении практических задач в соответствующей предметной области.

Самостоятельная работа студентов направлена на самостоятельное изучение дополнительного материала, подготовку к лабораторным занятиям, а также выполнения

всех видов самостоятельной работы.

Для успешного освоения дисциплины, необходимо выполнить все виды самостоятельной работы, используя рекомендуемые источники информации.

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля)

8.1.1. Перечень основной литературы:

1. Эванс Э. Предметно-ориентированное проектирование (DDD). Структуризация сложных программных систем. – М.: Вильямс, 2020.
2. Клеппман М. Высоконагруженные приложения. Программирование, масштабирование, поддержка. – СПб.: Питер, 2023.
3. Ньюмен С. Создание микросервисов. – СПб.: Питер, 2024.
4. Мартин Р. Clean Architecture: A Craftsman's Guide to Software Structure and Design. – СПб.: Питер, 2018. (Оригинал на английском языке)
5. Гагарина Л.Г. Введение в архитектуру программного обеспечения. – М.: ФОРУМ, 2020.

8.1.2. Перечень дополнительной литературы:

1. Рихтер Ч. CLR via C#. – СПб.: Питер, 2019.
2. Фримен Э., Робсон Э. Head First. Паттерны проектирования. – СПб.: Питер, 2022.
3. Стилмен М., Грин Д. Программирование на C++. – М.: Бином, 2018.
4. Дейтел Х.М. Как программировать на C++. – М.: Бином, 2019.
5. Хабибуллин И.Ш. Разработка мобильных приложений на Java. – СПб.: БХВ-Петербург, 2020.
6. Самардак А.С. Программирование на C#. – Томск: ТУСУР, 2021.
7. Зиборов В.В. Язык SQL для профессионалов. – СПб.: Наука и техника, 2020.
8. Айзекс С. Программирование на Python. – М.: Лаборатория знаний, 2021.
9. Федоров Д.Ю. Разработка веб-приложений на Django. – М.: ДМК Пресс, 2020.
10. Морган Б. Профессиональная разработка на React. – СПб.: Питер, 2021.

8.2. Перечень учебно-методического обеспечения самостоятельной работы обучающихся по дисциплине

8.3. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Znanium.com (znanium.com) - Крупнейшая российская электронно-библиотечная система (ЭБС). Фонд насчитывает более 90 тыс. документов, включая эксклюзивные издания ведущих российских издательств (ИНФРА-М, Вузовский учебник, Форум и др.). Доступ предоставляется через вузы-участники.
2. e.lanbook.com (ЭБС «Лань») - Российская платформа, предоставляющая доступ к учебной и научной литературе. Имеет контракты с ведущими вузами РФ на 2025-2026 гг. .
3. eLibrary.ru (**Научная электронная библиотека**) - Крупнейший российский информационно-аналитический портал в области науки, технологии, медицины и образования. Содержит рефераты и полные тексты научных статей и журналов, индексируемых в РИНЦ.

4. Consultant.ru (**КонсультантПлюс**) - Справочная правовая система, включенная в программу информационной поддержки российского образования. Доступ предоставляется вузам на льготной основе .
5. GitFlic.ru - **Отечественная** платформа для совместной разработки, альтернатива GitHub/GitLab. Предназначена для хранения репозитория, CI/CD и управления проектами. Является лидером среди российских DevOps-платформ .
6. **Yandex Cloud** - Российская облачная платформа (IaaS/PaaS). Предоставляет управляемые сервисы баз данных (PostgreSQL, Redis, Kafka), объектное хранилище (S3-совместимое) и Managed Kubernetes. Полностью соответствует требованиям к хостингу ПО на территории РФ (152-ФЗ)

9. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем

При чтении лекций используется компьютерная техника, демонстрации презентационных мультимедийных материалов. На семинарских и практических занятиях студенты представляют презентации, подготовленные ими в часы самостоятельной работы.

Информационные справочные системы:

Информационно-справочные и информационно-правовые системы, используемые при изучении дисциплины:

Программное обеспечение:

Системы контроля версий и CI/CD

GitFlic или GitLab (внутренний сервер).

GitFlic — рекомендованная отечественная платформа для хранения кода и CI/CD .

Среды разработки (IDE)

IntelliJ IDEA (Community Edition).

Visual Studio Code.

Примечание: Оба продукта распространяются свободно, их экосистема плагинов (PlantUML, Mermaid) продолжает работать без ограничений.

Базы данных и Брокеры сообщений

PostgreSQL (СУБД).

Redis (Кэш-сервер).

Apache Kafka или RabbitMQ (Брокеры сообщений).

Контейнеризация и Облака

Docker CE (версия для Linux).

Важно: Docker Desktop для Windows/Mac официально недоступен в РФ. Альтернативы: Использование Docker внутри виртуальной машины Linux (через VirtualBox) или легализованные аналоги, такие как Yandex Container Registry и сервисы Yandex Cloud для оркестрации .

Yandex Cloud (Облачная платформа).

Предоставляет Managed Kubernetes, Managed Databases (PostgreSQL, Redis, MongoDB) как сервис, что полностью заменяет необходимость ручной установки инфраструктуры на нелицензионном ПО .

Тестирование и Утилиты

JUnit 5, Mockito, TestContainers (Java-библиотеки, доступ к репозиториям Maven Central открыт).

Postman (Доступен, но для сложных сценариев можно использовать Insomnia или Swagger UI).

10. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Лекционные занятия	Учебная аудитория для проведения учебных занятий, оснащенная мультимедийным оборудованием и техническими средствами обучения.
Лабораторные работы	Учебная аудитория для проведения учебных занятий, оснащенная мультимедийным оборудованием и техническими средствами обучения.
Самостоятельная работа	Помещение для самостоятельной работы обучающихся оснащенное компьютерной техникой с возможностью подключения к сети "Интернет" и возможностью доступа к электронной информационно-образовательной среде университета

11. Особенности освоения дисциплины (модуля) лицами с ограниченными возможностями здоровья

Обучающимся с ограниченными возможностями здоровья предоставляются специальные учебники, учебные пособия и дидактические материалы, специальные технические средства обучения коллективного и индивидуального пользования, услуги ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, а также услуги сурдопереводчиков и тифлосурдопереводчиков.

Освоение дисциплины (модуля) обучающимися с ограниченными возможностями здоровья может быть организовано совместно с другими обучающимися, а также в отдельных группах.

Освоение дисциплины (модуля) обучающимися с ограниченными возможностями здоровья осуществляется с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья.

В целях доступности получения высшего образования по образовательной программе лицами с ограниченными возможностями здоровья при освоении дисциплины (модуля) обеспечивается:

1) для лиц с ограниченными возможностями здоровья по зрению:

- присутствие ассистента, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе, записывая под диктовку),
- письменные задания, а также инструкции о порядке их выполнения оформляются увеличенным шрифтом,
- специальные учебники, учебные пособия и дидактические материалы (имеющие крупный шрифт или аудиофайлы),
- индивидуальное равномерное освещение не менее 300 люкс,
- при необходимости студенту для выполнения задания предоставляется увеличивающее устройство;

2) для лиц с ограниченными возможностями здоровья по слуху:

- присутствие ассистента, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе, записывая под диктовку),
- обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости обучающемуся предоставляется звукоусиливающая аппаратура индивидуального пользования;
- обеспечивается надлежащими звуковыми средствами воспроизведения информации;

3) для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата (в том числе с тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;
- по желанию студента задания могут выполняться в устной форме.

12. Особенности реализации дисциплины с применением дистанционных образовательных технологий и электронного обучения

Согласно части 1 статьи 16 Федерального закона от 29 декабря 2012 г. № 273-ФЗ «Об образовании в Российской Федерации» под *электронным обучением* понимается организация образовательной деятельности с применением содержащейся в базах данных и используемой при реализации образовательных программ информации и обеспечивающих ее обработку информационных технологий, технических средств, а также информационно-телекоммуникационных сетей, обеспечивающих передачу по линиям связи указанной информации, взаимодействие обучающихся и педагогических работников. Под *дистанционными образовательными технологиями* понимаются образовательные технологии, реализуемые в основном с применением информационно-телекоммуникационных сетей при опосредованном (на расстоянии) взаимодействии обучающихся и педагогических работников.

Реализация дисциплины может быть осуществлена с применением дистанционных образовательных технологий и электронного обучения полностью или частично. Компоненты УМК дисциплины (рабочая программа дисциплины, оценочные и методические материалы, формы аттестации), реализуемой с применением дистанционных образовательных технологий и электронного обучения, содержат указание на их использование.

При организации образовательной деятельности с применением дистанционных образовательных технологий и электронного обучения могут предусматриваться асинхронный и синхронный способы осуществления взаимодействия участников образовательных отношений посредством информационно-телекоммуникационной сети «Интернет».

При применении дистанционных образовательных технологий и электронного обучения в расписании по дисциплине указываются: способы осуществления взаимодействия участников образовательных отношений посредством информационно-телекоммуникационной сети «Интернет» (ВКС-видеоконференцсвязь, ЭТ – электронное тестирование); ссылки на электронную информационно-образовательную среду СКФУ, на образовательные платформы и ресурсы иных организаций, к которым предоставляется открытый доступ через информационно-телекоммуникационную сеть «Интернет»; для синхронного обучения - время проведения онлайн-занятий и преподаватели; для асинхронного обучения - авторы онлайн-курсов.

При организации промежуточной аттестации с применением дистанционных образовательных технологий и электронного обучения используются Методические рекомендации по применению технических средств, обеспечивающих объективность результатов при проведении промежуточной и государственной итоговой аттестации по образовательным программам высшего образования - программам бакалавриата, программам специалитета и программам магистратуры с применением дистанционных образовательных технологий (Письмо Минобрнауки России от 07.12.2020 г. № МН-19/1573-АН "О направлении методических рекомендаций").

Реализация дисциплины с применением электронного обучения и дистанционных образовательных технологий осуществляется с использованием электронной информационно-образовательной среды СКФУ, к которой обеспечен доступ обучающихся через информационно-телекоммуникационную сеть «Интернет», или с использованием

ресурсов иных организаций, в том числе платформ, предоставляющих сервисы для проведения видеоконференций, онлайн-встреч и дистанционного обучения (МТС-Линк), а также с использованием возможностей социальных сетей для осуществления коммуникации обучающихся и преподавателей.

Учебно-методическое обеспечение дисциплины, реализуемой с применением электронного обучения и дистанционных образовательных технологий, включает представленные в электронном виде рабочую программу, учебно-методические пособия или курс лекций, методические указания к выполнению различных видов учебной деятельности обучающихся, предусмотренных дисциплиной, и прочие учебно-методические материалы, размещенные в информационно-образовательной среде СКФУ.