

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

Программная инженерия

Направление подготовки
Направленность (профиль)

09.03.04 Программная инженерия
Инженерия сложных программных
систем

Год начала обучения

2026

Форма обучения

очная

Реализуется в семестрах

4,5

Разработано

Заведующая межинститутской базовой
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)

Целью освоения дисциплины «Программная инженерия» является формирование у обучающихся системы теоретических знаний и практических навыков в области систематического, дисциплинированного и измеримого подхода к разработке, эксплуатации и сопровождению программного обеспечения, а также подготовка к профессиональной деятельности в области проектирования, реализации и управления качеством программных продуктов с использованием современных методологий, архитектурных паттернов и технологических стеков

Задачи освоения дисциплины:

1. **Сформировать системное понимание** жизненного цикла программного обеспечения и методологий разработки.
2. **Научить собирать, анализировать и документировать** требования к программному продукту.
3. **Познакомить с современными архитектурными подходами**, включая паттерн PCMEF, и научить применять их на практике.
4. **Сформировать навыки проектирования** баз данных, объектно-реляционного отображения и реализации бизнес-логики.
5. **Освоить паттерны проектирования** GoF и методы рефакторинга для повышения качества кода.
6. **Научить тестировать** программное обеспечение (модульное, интеграционное) и оценивать его качество.
7. **Сформировать умение создавать** REST API, веб- и мобильные интерфейсы с использованием современных технологий (Spring Boot, React).
8. **Обеспечить понимание основ безопасности** программных систем (аутентификация, авторизация, JWT).
9. **Научить работать с системами контроля версий** (Git) и контейнеризации (Docker).
10. **Подготовить к самостоятельной профессиональной деятельности** по разработке и сопровождению программных продуктов

2. Место дисциплины (модуля) в структуре образовательной программы

Дисциплина «Программная инженерия» относится к обязательной части.

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
ОПК-3: Способен решать стандартные задачи профессиональной деятельности на основе	ИД-2 ОПК-3 Применяет инструменты документирования и визуального	Способен создавать техническую документацию в формате Markdown и разрабатывать UML-диаграммы (классов, последовательности, вариантов использования) с

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	моделирования (Markdown, PlantUML, Draw.io) для создания технической документации и диаграмм в процессе разработки программного обеспечения	использованием инструментов PlantUML и Draw.io для визуализации архитектуры программной системы.
	ИД-3 ОПК-3 Использует системы контроля версий (Git) и платформы совместной разработки (GitHub, GitLab) для организации командной работы и отслеживания изменений в коде и документации	Способен организовывать совместную разработку с использованием систем контроля версий Git и платформ GitHub/GitLab, выполняя операции с ветками, разрешение конфликтов и отслеживание изменений в коде и документации на протяжении всего жизненного цикла проекта.
	ИД-4 ОПК-3 Учитывает требования информационной безопасности при разработке: применяет методы защиты от типовых уязвимостей (SQL injection, XSS, CSRF), реализует аутентификацию и авторизацию	Способен обеспечивать информационную безопасность программного продукта путём реализации аутентификации на основе JWT-токенов, ролевой авторизации (USER, ADMIN) и применения методов защиты от типовых уязвимостей (SQL injection, XSS, CSRF).

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
ОПК-4: Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ИД-1 ОПК-4 Разрабатывает техническое задание на создание программного продукта в соответствии с требованиями заказчика, включая функциональные и нефункциональные требования, пользовательские истории и варианты использования	Способен разрабатывать техническое задание на программный продукт, включая функциональные и нефункциональные требования, пользовательские истории, варианты использования (Use Case) и критерии приемки, с использованием методов MoSCoW и FURPS+.
	ИД-2 ОПК-4 Оформляет проектную и техническую документацию (пояснительную записку, руководство пользователя, руководство администратора) в соответствии с действующими стандартами (ЕСПД, ISO/IEC 12207)	Способен оформлять пояснительную записку, руководство пользователя и руководство администратора в формате Markdown в соответствии с требованиями стандартов ЕСПД и ISO/IEC 12207, включая структурирование отчётов, оформление титульных листов и списков литературы.
ОПК-8: Способен осуществлять поиск, хранение, обработку и анализ информации из	ИД-1 ОПК-8 Осуществляет эффективный поиск профессиональной	Способен осуществлять эффективный поиск профессиональной информации в специализированных источниках (IEEE, ACM, Stack Overflow, GitHub) с

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий	информации в специализированных источниках (IEEE, ACM, Scopus, Web of Science, Stack Overflow, GitHub) с использованием операторов расширенного поиска	использованием операторов расширенного поиска для решения задач проектирования и разработки программных систем.
ПК-2: Способен выполнять проектирование и разработку архитектуры программных систем среднего и крупного масштаба с применением современных паттернов и методов проектирования	ИД-1 ПК-2 Проектирует архитектуру программных систем с использованием современных архитектурных стилей и подходов	Способен проектировать архитектуру программных систем с использованием архитектурного паттерна PCMEF (Presentation-Control-Mediator-Entity-Foundation) и его адаптаций для различных траекторий (Desktop, Web, Mobile, Enterprise).
	ИД-2 ПК-2.2 Применяет архитектурный паттерн PCMEF и его адаптации для различных типов приложений	Способен адаптировать архитектурный паттерн PCMEF для различных типов приложений (деSKTOPных, веб- и мобильных), определяя распределение слоёв между клиентом и сервером и специфицируя интерфейсы взаимодействия (IService, IRepository).
	ИД-3 ПК-2 Документирует архитектурные решения с использованием современных подходов (Arc42, ADR)	Способен документировать архитектурные решения с использованием шаблона Arc42 и формата Architecture Decision Records (ADR), фиксируя контекст, обоснование, последствия и

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
		альтернативы принятых решений.
	ИД-4 ПК-2 Оценивает и сравнивает альтернативные архитектурные решения	Способен анализировать и сравнивать альтернативные архитектурные стили (MVC, Clean Architecture, микросервисная архитектура) и обосновывать выбор оптимального решения для конкретной предметной области.
	ИД-5 ПК-2 Проектирует распределенные системы с учетом CAP-теоремы	Способен учитывать ограничения CAP-теоремы при проектировании распределенных систем, выбирая компромиссы между согласованностью, доступностью и устойчивостью к разделению в зависимости от требований к системе.
ПК-8: Способен адаптировать и применять методы искусственного интеллекта и машинного обучения для решения профессиональных задач, включая интеграцию готовых моделей в программные продукты	ИД-1 ПК-8 Понимает основные концепции и методы искусственного интеллекта и машинного обучения	Способен объяснять основные концепции и методы искусственного интеллекта и машинного обучения (классификация, регрессия, кластеризация), а также определять применимость ML-решений для задач предметной области.
	ИД-2 ПК-8 Интегрирует готовые модели и API искусственного интеллекта (OpenAI, Yandex GPT, Google Cloud AI) в	Способен интегрировать готовые модели и API искусственного интеллекта в разрабатываемое программное обеспечение через REST API, обрабатывать результаты предсказаний и использовать их в бизнес-логике приложения.

Код, формулировка компетенции	Код, формулировка индикатора	Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций, индикаторов
	разрабатываемое программное обеспечение через REST API	
	ИД-3 ПК-8 Адаптирует существующие модели машинного обучения для конкретных задач предметной области методами дообучения (fine-tuning)	Способен выполнять дообучение (fine-tuning) существующих моделей машинного обучения для адаптации к специфическим задачам предметной области с учётом требований к данным и вычислительным ресурсам.

4. Объем учебной дисциплины и формы контроля *

Объем занятий: всего: 7 з.е. 252 ак.ч.	ОФО, в астр. часах
Контактная работа:	90
Лекции/из них практическая подготовка	36
Лабораторных работ/из них практическая подготовка	54
Практических занятий/из них практическая подготовка	
Самостоятельная работа	108
Формы контроля	
Экзамен	54
Зачет	
Зачет с оценкой	
Расчетно-графические работы	
Курсовой проект	±
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	Введение в программную инженерию. Жизненный цикл ПО 1. Определение программной инженерии, отличие от программирования 2. Исторический контекст: «Кризис программного обеспечения» 1968 г. 3. 6 тезисов о сущности ПИ (ПО в бизнес-контексте, сложность, моделирование) 4. Стадии жизненного цикла ПО: анализ, проектирование, реализация, тестирование, сопровождение 5. Обзор стандартов: ISO/IEC 12207, SWEBOOK 6. Сравнение программной и традиционной инженерии	ИД-1 ОПК-3 ИД-2 ОПК-3 ИД-1 ОПК-8	2		2	8
2	Модели жизненного цикла и методологии разработки 1. Каскадная модель (Waterfall) и V-модель 2. Итеративные и инкрементальные модели 3. Спиральная модель (Boehm, 1988) 4. Rational Unified Process (RUP) 5. Гибкие методологии: Agile-манифест, Scrum, Kanban, XP 6. Критерии выбора модели ЖЦ в зависимости от проекта	ИД-1 ОПК-3 ИД-2 ОПК-3 ИД-1 ОПК-8	2		2	8

3	Инструментарий программной инженерии 1. Классификация инструментов: управление проектом, моделирование, IDE, СКВ 2. Инструменты управления проектом: MS Project, GanttPRO, Jira 3. Инструменты моделирования: PlantUML, Draw.io, Enterprise Architect 4. Интегрированные среды разработки: IntelliJ IDEA, Eclipse, VS Code 5. Системы контроля версий: Git, GitHub/GitLab, GitFlow 6. Интеграция инструментов в единую среду разработки	ИД-1 ОПК-3 ИД-2 ОПК-3 ИД-3 ОПК-3 ИД-1 ПК-2	2		2	8
4	Управление требованиями 1. Роль требований в успехе проекта (47% неудач — неясные требования) 2. Классификация требований: функциональные, нефункциональные, бизнес-требования 3. Процесс управления требованиями: выявление, анализ, документирование, валидация 4. Техники выявления: интервью, анкетирование, мозговой штурм, прототипирование 5. Спецификация требований: SRS, User Stories, Use Cases 6. Методы приоритизации: MoSCoW, FURPS+ 7. Трассировка	ИД-1 ОПК-4 ИД-1 ПК-2	2		2	8
5	Язык моделирования UML (часть 1) 1. Назначение моделирования в программной инженерии 2. Структурное vs объектно-ориентированное моделирование 3. Диаграммы вариантов использования (Use Case): акторы, прецеденты, отношения include/extend 4. Диаграммы классов (Class Diagram): классы, атрибуты, методы, отношения (ассоциация, агрегация, композиция, наследование) 5. Кратности связей	ИД-2 ОПК-3 ИД-1 ПК-2	2		2	8

6	Язык моделирования UML (часть 2) 1. Диаграммы последовательности (Sequence Diagram): линии жизни, сообщения, фрагменты (alt, opt, loop) 2. Диаграммы деятельности (Activity Diagram): действия, потоки, ветвления 3. Диаграммы состояний (State Machine Diagram): состояния, переходы, события 4. Диаграммы компонентов и развертывания 5. Практические рекомендации по выбору диаграмм	ИД-2 ОПК-3 ИД-1 ПК-2	2		2	8
7	Введение в архитектуру ПО. Паттерн PCMEF 1. Что такое архитектура ПО? Роль архитектора 2. Принципы архитектуры: модульность, слабое зацепление, высокая связность, тестируемость 3. Обзор архитектурных стилей: Layered, MVC, Client-Server, Microservices 4. Паттерн PCMEF (Presentation-Control-Mediator-Entity-Foundation) 5. Структура PCMEF: ответственность каждого слоя 6. Правило зависимостей: только сверху вниз	ИД-1ПК-2.1, ИД-ПК-2.2	2		2	8
8	Адаптация PCMEF для различных траекторий 1. Классический PCMEF (монолит) для десктопных приложений 2. Распределенный PCMEF для веб-приложений (тонкий клиент) 3. Клиент-серверная адаптация для мобильных приложений 4. Дополнительные слои: State Management, API Client, Local Cache 5. Сравнение реализаций для разных траекторий 6. Выбор траектории в зависимости от требований	ИД-1 ПК-2 ИД-2 ПК-2 ИД-4 ПК-2	2		2	8

9	Проектирование баз данных и ORM 1. Реляционная модель данных: таблицы, атрибуты, ключи 2. Нормализация: 1НФ, 2НФ, 3НФ (примеры и проверка) 3. Типы связей и их реализация: 1:1, 1:N, M:N 4. Объектно-реляционное отображение (ORM): концепция 5. JPA и Hibernate: основные аннотации (@Entity, @Table, @Id, @GeneratedValue, @Column) 6. Стратегии отображения наследования: SINGLE_TABLE, JOINED, TABLE_PER_CLASS	ИД-1 ПК-2 ИД-1 ПК-8	2		2	8
	Зачет с оценкой					
	ИТОГО за 5 семестр		18		18	72
1	Паттерны проектирования GoF (часть 1) 1. Что такое паттерны проектирования? История GoF (Gang of Four) 2. Классификация паттернов: порождающие, структурные, поведенческие (23 паттерна) 3. Порождающие паттерны: Singleton, Factory Method, Abstract Factory, Builder, Prototype 4. Примеры применения в контексте РСМЕФ (Singleton для соединения с БД, Factory для репозитория) 5. Критерии выбора паттернов	ИД-1 ПК-2 ИД-1 ПК-8	2		4	2
2	Паттерны проектирования GoF (часть 2) 1. Структурные паттерны: Adapter, Facade, Proxy, Decorator, Composite 2. Поведенческие паттерны: Observer, Strategy, Command, Mediator, Chain of Responsibility 3. Связь паттерна Mediator со слоем Mediator в РСМЕФ 4. Паттерн Observer в реактивных интерфейсах (Mobile, React) 5. Паттерн Strategy для алгоритмов расчета (скидки, ценообразование) 6. Паттерн Command для инкапсуляции запросов	ИД-1 ПК-2 ИД-1 ПК-8	2		4	2

3	Детальное проектирование. Диаграммы последовательности 1. От архитектуры к детальному проектированию 2. Диаграммы последовательности как инструмент детализации 3. Выявление недостающих методов и уточнение сигнатур 4. Обнаружение нарушений архитектуры PCMEF 5. Уточнение диаграммы классов проектирования 6. Спецификация методов: сигнатуры, параметры, исключения, JavaDoc	ИД-1 ПК-2 ИД-3 ПК-2 ИД-2 ОПК-4	2		4	2
4	Реализация ядра системы (Entity, Foundation, Mediator, Control) 1. Структура проекта в соответствии с PCMEF 2. Реализация слоя Entity: JPA-сущности, бизнес-методы (богатая модель) 3. Реализация слоя Foundation: репозитории, CRUD-операции, кастомные запросы (@Query) 4. Реализация слоя Mediator: сервисы с бизнес-логикой, управление транзакциями (@Transactional) 5. Реализация слоя Control: REST-контроллеры, валидация DTO, обработка исключений (@ControllerAdvice) 6. Инверсия управления и внедрение зависимостей (DI)	ИД-2 ПК-2 ИД-4 ОПК-3	2		4	2
5	Модульное и интеграционное тестирование 1. Роль тестирования в обеспечении качества 2. Уровни тестирования: модульное, интеграционное, системное 3. JUnit 5: аннотации, assertions, параметризованные тесты 4. Mockito: создание заглушек и моков для изолированного тестирования сервисов 5. Разработка через тестирование (TDD): цикл red-green-refactor 6. Интеграционное тестирование с TestContainers (реальная PostgreSQL в Docker) 7. Покрытие кода тестами (JaCoCo, цель >40%)	ИД-1 ПК-2 ИД-4 ОПК-3	2		4	2

6	Качество кода и рефакторинг 1. Понятие качества программного обеспечения 2. Метрики качества кода: сложность, связность, зацепление 3. «Запахи кода» (Code Smells): длинные методы, большие классы, дублирование 4. Инструменты статического анализа: SonarQube, Checkstyle, PMD 5. Паттерны рефакторинга: Data Mapper, Identity Map, Lazy Load, Unit of Work 6. Безопасный рефакторинг: роль тестов	ИД-1 ПК-2	2	4	2
7	REST API и документация 1. REST архитектурный стиль: ресурсы, методы, статусы 2. Принципы проектирования API: идемпотентность, безопасность, версионирование 3. Документирование API: OpenAPI/Swagger 4. Реализация REST API в Spring: @RestController, @RequestMapping, @GetMapping, @PostMapping, @PutMapping, @DeleteMapping 5. Обработка ошибок в REST API: @ControllerAdvice, @ExceptionHandler 6. Тестирование API: Postman, интеграционные тесты	ИД-2 ПК-2 ИД-2 ОПК-4	2	4	2
8	Безопасность и аутентификация в веб-приложениях 1. Основы безопасности: аутентификация vs авторизация 2. JWT (JSON Web Tokens): структура, принцип работы (access + refresh токены) 3. Хеширование паролей: BCrypt 4. Spring Security: настройка SecurityConfig, JWT-фильтры 5. Авторизация на основе ролей: @PreAuthorize, ROLE_USER, ROLE_ADMIN 6. Защита от типовых атак: SQL injection, XSS, CSRF, CORS	ИД-4 ОПК-3 ИД-2 ПК-2	2	4	2

9	Проектирование пользовательских интерфейсов. React 1. Принципы UI/UX-дизайна: согласованность, обратная связь, снисходительность 2. Архитектура современных веб-приложений: клиент-сервер, SPA 3. React: компонентный подход, JSX, хуки (useState, useEffect) 4. Маршрутизация с React Router DOM 5. Управление состоянием: Context API, Redux Toolkit 6. Валидация форм: React Hook Form + Zod 7. Адаптивная вёрстка: Bootstrap/Tailwind CSS	ИД-2 ПК-2 ИД-2 ОПК-4				
	Экзамен	54				
	ИТОГО за 6 семестр		18		36	36
	ИТОГО		16		16	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. **Лаврищева, Е. М.** Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561885>
2. **Черткова, Е. А.** Программная инженерия. Визуальное моделирование программных систем : учебник для вузов / Е. А. Черткова. — 3-е изд., пер. и доп. — Москва : Издательство Юрайт, 2026. — 146 с. — (Высшее образование). — ISBN 978-5-534-18094-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/563828>
3. **Щербак, А. В.** Тестирование программного обеспечения : учебник для вузов / А. В. Щербак. — Москва : Издательство Юрайт, 2025. — 145 с. — (Высшее образование). — ISBN 978-5-534-19291-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/580604>

8.1.2. Перечень дополнительной литературы:

1. **Казарин, О. В.** Основы информационной безопасности: надежность и безопасность программного обеспечения : учебник для вузов / О. В. Казарин, И. Б. Шубинский. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 352 с. — (Высшее образование). — ISBN 978-5-534-19386-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/580669>
2. **Гутгарц, Р. Д.** Проектирование автоматизированных систем обработки информации и управления : учебник для вузов / Р. Д. Гутгарц. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 351 с. — (Высшее образование). — ISBN 978-5-534-15761-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/565010>
3. **Мейер, Б.** Основы объектно-ориентированного проектирования : учебник / Б. Мейер. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 751 с. — (Премиум-версия IPR SMART). — ISBN 978-5-4497-0885-4
4. **Стружкин, Н. П.** Базы данных: проектирование : учебник для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2026. — 477 с. — (Высшее образование). — ISBN 978-5-534-00229-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/580000>

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

1. Методические указания по выполнению лабораторных работ по дисциплине «Программная инженерия» для студентов направления подготовки 09.03.04 «Программная инженерия» (разработаны на кафедре).
2. Методические рекомендации по организации самостоятельной работы студентов по дисциплине «Программная инженерия» (разработаны на кафедре).

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

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:

Электронные библиотечные системы (ЭБС):

ЭБС «Юрайт» — <https://urait.ru>

ЭБС IPR SMART — <https://www.iprbookshop.ru>

Научная электронная библиотека eLibrary.ru — <https://www.elibrary.ru>

Электронная библиотека МГТУ им. Н.Э. Баумана — <https://library.bmstu.ru>

Профессиональные информационные ресурсы:

Stack Overflow — <https://stackoverflow.com>

GitHub — <https://github.com>

Хабр — <https://habr.com>

Docker Documentation — <https://docs.docker.com>

Spring Documentation — <https://spring.io/projects/spring-framework>

React Documentation — <https://react.dev>

PlantUML — <https://plantuml.com>

Baeldung — <https://www.baeldung.com>

Программное обеспечение:

реды разработки и редакторы кода:

IntelliJ IDEA Community Edition

Visual Studio Code

Системы управления базами данных:

PostgreSQL

DBeaver Community Edition

pgAdmin

Инструменты контейнеризации и виртуализации:

Docker Desktop Community Edition

VirtualBox (опционально)

Системы контроля версий и сборки:

Git

Maven

Gradle

Инструменты тестирования и анализа качества кода:

JUnit 5

Mockito

JaCoCo

SonarQube Community Edition

Postman

Инструменты моделирования и документирования:

PlantUML

Draw.io (diagrams.net)

Среды выполнения и фреймворки:

Node.js

Vite

Операционные системы:

Microsoft Windows (лицензионная)

Linux (Ubuntu) — свободно распространяемое ПО

9.2. Информационные справочные системы

КонсультантПлюс — <http://www.consultant.ru>

Гарант — <http://www.garant.ru>

Росстандарт — <https://www.rst.gov.ru>

Федеральный портал проектов нормативных правовых актов — <https://regulation.gov.ru>

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.