

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»

Структурное подразделение «Брикс кафедры (205)»

УТВЕРЖДЕНА:
на заседании кафедры
Протокол №6 от 25 февраля 2026 г.

Рабочая программа дисциплины

«ПРОГРАММНАЯ ИНЖЕНЕРИЯ / SOFTWARE ENGINEERING»

Направление: 09.04.02 Информационные системы и технологии

Цифровые технологии, сети и большие данные / Information technologies, networks and big data

Квалификация: Магистр

Форма обучения: очная

Документ подписан простой
электронной подписью
Составитель программы:
Дорофеев Андрей Сергеевич
Дата подписания: 11.06.2026

Документ подписан простой
электронной подписью
Утвердил: Киреенко Анна
Павловна
Дата подписания: 11.06.2026

Документ подписан простой
электронной подписью
Согласовал: Дорофеев Андрей
Сергеевич
Дата подписания: 11.06.2026

Год набора – 2026

Иркутск, 2026 г.

1 Перечень планируемых результатов обучения по дисциплине, соотнесённых с планируемыми результатами освоения образовательной программы

1.1 Дисциплина «Программная инженерия / Software Engineering» обеспечивает формирование следующих компетенций с учётом индикаторов их достижения

Код, наименование компетенции	Код индикатора компетенции
ОПК-1 Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	ОПК-1.5
ПК-3 Способен организовать управление ресурсами и процессами при проектировании информационных систем	ПК-3.9

1.2 В результате освоения дисциплины у обучающихся должны быть сформированы

Код индикатора	Содержание индикатора	Результат обучения
ОПК-1.5	Решает нестандартные профессиональные задачи разработки программного обеспечения, в том числе в новой или незнакомой среде с применением математических, естественнонаучных и профессиональных знаний	Знать основные понятия о видах и назначении программного обеспечения, технологиях и методах разработки программных продуктов, сведения о техническом сопровождении программных продуктов после внедрения; методы сопровождения проектов по разработке программного обеспечения. Уметь использовать методы разработки прикладного программного обеспечения для решения задач, возникающих в сфере профессиональной деятельности. Владеть практическими навыками решения задач программной инженерии; методами проектирования программного продукта; навыками естественнонаучного мышления, необходимыми для использования методов разработки прикладного программного обеспечения.
ПК-3.9	Анализирует сообщения об ошибках в программных средствах и проводит их отладку	Знать методы тестирования и отладки программного обеспечения до и на стадии внедрения. Уметь анализировать сообщения об ошибках во время отладки по в

		системной среде; принимать решение на основе анализа сообщений об ошибках Владеть практическими навыками и инструментами отладки и тестирования по в различных системных средах при различных условиях эксплуатации.
--	--	--

2 Место дисциплины в структуре ООП

Изучение дисциплины «Программная инженерия / Software Engineering» базируется на результатах освоения следующих дисциплин/практик: «Проектирование и менеджмент аналитических платформ / Design and management of data analytics platforms», «Технологическая платформа 1C:Enterprise Platform / 1C:Enterprise Platform», «Управление проектами / Project Management»

Дисциплина является предшествующей для дисциплин/практик: «Эффективное управление данными / Advanced Data Management», «Управление данными и проектирование баз данных / Database Design and Management», «Программирование искусственного интеллекта в приложениях / Artificial Intelligence Programming in Applications», «Конфигурирование и разработка решений 1C / 1C:Enterprise Applications Development»

3 Объем дисциплины

Объем дисциплины составляет – 3 ЗЕТ

Вид учебной работы	Трудоемкость в академических часах (Один академический час соответствует 45 минутам астрономического часа)	
	Всего	Семестр № 3
Общая трудоемкость дисциплины	108	108
Аудиторные занятия, в том числе:	48	48
лекции	16	16
лабораторные работы	0	0
практические/семинарские занятия	32	32
Самостоятельная работа (в т.ч. курсовое проектирование)	24	24
Трудоемкость промежуточной аттестации	36	36
Вид промежуточной аттестации (итогового контроля по дисциплине)	Экзамен	Экзамен

4 Структура и содержание дисциплины

4.1 Сводные данные по содержанию дисциплины

Семестр № 3

№ п/п	Наименование раздела и темы дисциплины	Виды контактной работы						СРС		Форма текущего контроля
		Лекции		ЛР		ПЗ(СЕМ)				
		№	Код.	№	Код.	№	Код.	№	Код.	

			Час.		Час.		Час.		Час.	
1	2	3	4	5	6	7	8	9	10	11
1	Программная инженерия, основные принципы и понятия	1	2					3, 4	8	Устный опрос, Проработка отдельных разделов теоретического курса
2	Жизненный цикл и процесс разработки ПО	2	2			1	4	1, 2	2	Проработка отдельных разделов теоретического курса, Устный опрос, Отчет
3	Методологии разработки ПО	3	2			2	6	2, 4	2	
4	Анализ предметной области и требования к ПО	4	4			3, 4	10	1, 2	3	
5	Архитектура программного обеспечения. Принципы создания пользовательского интерфейса	5	4			5	8	2, 4, 4	5	
6	Методы верификации и тестирования программ и систем	6	2			6	4	2, 4	2	
	Промежуточная аттестация								36	Экзамен
	Всего		16				32		58	

4.2 Краткое содержание разделов и тем занятий

Семестр № 3

№	Тема	Краткое содержание
1	Программная инженерия, основные принципы и понятия	Предпосылки и история. Программная инженерия – что это такое? Основные понятия дисциплины.
2	Жизненный цикл и процесс разработки ПО	Начало стандартизации жизненного цикла по. Стандарт ISO/IEC – процессы жизненного цикла ПП. Модель жизненного цикла программного продукта.
3	Методологии разработки ПО	Методологии разработки ПО: SCRUM, KANBAN, DSDM, MSF, RUP, AUP.
4	Анализ предметной области и требования к ПО	Анализ предметной области: цели и задачи. Модели предметной области. Формальные определения. Классификация моделей.

		Методология IDEF0, синтаксис IDEF0-моделей. Определение требований к программному обеспечению. Источники требований. Методы выявления требований. Модели жизненного цикла разработки и этапы работы с требованиями. Классификация требований. Функциональные требования.
5	Архитектура программного обеспечения. Принципы создания пользовательского интерфейса	Языки описания архитектуры. Принципы создания пользовательского интерфейса Факторы влияния: функциональность, эстетичный внешний вид, производительность. Разработка интерфейса как процесс: проектирование, реализация, тестирование.
6	Методы верификации и тестирования программ и систем	Формальные методы верификации. Логикоалгебраические модели. Исполнимые модели. Динамические методы. Мониторинг. Тестирование. Синтетические методы.

4.3 Перечень лабораторных работ

Лабораторных работ не предусмотрено

4.4 Перечень практических занятий

Семестр № 3

№	Темы практических (семинарских) занятий	Кол-во академических часов
1	Модели жизненного цикла MSF, RUP, XP	4
2	Декомпозиция подсистем на модули. Модульность	6
3	Анализ предметной области и требования к ПО	4
4	Использование UML для описания требований к системе	6
5	Создание пользовательского интерфейса	8
6	Методы верификации программ и систем	4

4.5 Самостоятельная работа

Семестр № 3

№	Вид СРС	Кол-во академических часов
1	Оформление отчетов по лабораторным и практическим работам	4
2	Подготовка к практическим занятиям	1
3	Подготовка к экзамену	6
4	Проработка разделов теоретического материала	11

В ходе проведения занятий по дисциплине используются следующие интерактивные методы обучения: В ходе проведения лекций и практических занятий используются следующие интерактивные методы обучения: взаимная проверка выполненных заданий,

тестирование приложений; демонстрация работы с CASE средствами путем синхронной работы преподавателя (показывает на экране с использованием проекционной техники) и обучающихся, которые повторяют действия преподавателя (при необходимости обучающиеся задают вопросы).

5 Перечень учебно-методического обеспечения дисциплины

5.1 Методические указания для обучающихся по освоению дисциплины

5.1.1 Методические указания для обучающихся по практическим занятиям

Неотъемлемой частью изучения дисциплины является выполнение практических заданий, основной целью которых является выработка умений проектирования, разработки, тестирования ПО. Совокупность методических указаний по выполнению заданий практических работ и СРС, а также материалов конспекта лекций обеспечивают достаточный объем информации для успешного освоения дисциплины.

5.1.2 Методические указания для обучающихся по самостоятельной работе:

1) Самостоятельное изучение отдельных разделов дисциплины

Цель: получение более глубоких знаний и навыков по основным разделам дисциплины. Обучающимся необходимо найти информацию по теме с использованием основной и дополнительной литературы, изучить теоретический материал, разобраться с примерами. Контроль осуществляется включением в экзаменационные вопросы изученных самостоятельно тем.

2) Подготовка к практическим занятиям.

Цель: работа с методическими указаниями к выполнению работы, повторение материала для защиты работы.

3) Подготовка отчетов и защите работы.

Цель: закрепление полученных в ходе изучения дисциплины знаний, умений и навыков.

4) Подготовка к экзамену.

Цель: закрепление полученных в ходе изучения дисциплины знаний, умений и навыков по определенному перечню вопросов для самоподготовки, включающему теоретическую и практическую части.

6 Фонд оценочных средств для контроля текущей успеваемости и проведения промежуточной аттестации по дисциплине

6.1 Оценочные средства для проведения текущего контроля

6.1.1 семестр 3 | Устный опрос

Описание процедуры.

Устный опрос заключается в ответах на вопросы, предложенные преподавателям по теме, связанной с защищаемой работой.

Критерии оценивания.

Устный опрос считается пройденным успешно при удовлетворительных ответах на заданные вопросы, владении основным материалом курса (раздела курса).

6.1.2 семестр 3 | Отчет

Описание процедуры.

Отчет по практической работы включает описание выполненного задания с иллюстрациями (модели, схемы, декомпозиция, диаграммы и т.п.) согласно условия задачи.

Критерии оценивания.

Для зачета работы требуется подробное описание результатов выполненной работы в виде отчета. Оценивается полнота, качество, грамотное принятие решений.

6.1.3 семестр 3 | Проработка отдельных разделов теоретического курса

Описание процедуры.

Отдельные разделы дисциплины требуют дополнительного изучения материала из дополнительных источников с целью более глубокого понимания дисциплины.

Критерии оценивания.

Оценивание данного вида СРС производится, например, при защите работ, при проведении промежуточной аттестации по дисциплине. Рассматриваются полнота и качество изученного материала.

6.2 Оценочные средства для проведения промежуточной аттестации

6.2.1 Критерии и средства (методы) оценивания индикаторов достижения компетенции в рамках промежуточной аттестации

Индикатор достижения компетенции	Критерии оценивания	Средства (методы) оценивания промежуточной аттестации
ОПК-1.5	Демонстрирует знание основ разработки программного обеспечения любого вида и уровня, методологии и инструментов программной инженерии, технического сопровождения проектов по разработке ПО. Умеет решать нестандартные профессиональные задачи, в том числе в новой или незнакомой среде с применением методов программной инженерии. Владеет навыками теоретического и экспериментального исследования объектов профессиональной деятельности с применением методов программной инженерии	Устное собеседование и/или практические задания и/или тест.
ПК-3.9	Демонстрирует знание методы тестирования и отладки программного обеспечения до и на стадии внедрения. Умеет анализировать сообщения об	Практические задания и/или тест.

	ошибках во время отладки ПО в системной среде; принимать решение на основе анализа сообщений об ошибках. Владеет отладки и тестирования ПО в различных системных средах при различных условиях эксплуатации.	
--	--	--

6.2.2 Типовые оценочные средства промежуточной аттестации

6.2.2.1 Семестр 3, Типовые оценочные средства для проведения экзамена по дисциплине

6.2.2.1.1 Описание процедуры

Время экзамена – 60 мин. Экзамен проводится при наличии выполненных практических работ. Экзаменационный билет содержит два теоретических вопроса и одно практическое задание. Практические задания предлагаются в следующих формах:

- Анализ предметной области и требования к ПО
- Использование UML для описания требований к системе
- Декомпозиция подсистем на модули.
- Создание пользовательского интерфейса.
- Методы верификации программ и систем.

Вопросы к экзамену:

1. Предпосылки и история программной инженерии.
2. Профессиональные и этические требования.
3. Стандарты программной инженерии.
4. Начало стандартизации жизненного цикла программного обеспечения.
5. Модель жизненного цикла программного продукта.
6. Модели жизненного цикла MSF, RUP, XP.
7. Методологии разработки ПО: SCRUM, KANBAN, DSDM, MSF, RUP, AUP.
8. Анализ предметной области: цели и задачи.
9. Модели предметной области.
10. Формальные определения.
11. Классификация моделей.
12. Методология IDEF0, синтаксис IDEF0 моделей.
13. Определение требований к программному обеспечению.
14. Источники требований.
15. Методы выявления требований.
16. Модели жизненного цикла разработки и этапы работы с требованиями.
17. Классификация требований.
18. Функциональные требования.
19. Способы и методы их описания.
20. Нефункциональные требования.
21. Критерии качества требований.
22. Особенности этапа проектирования.
23. Структурирование системы.
24. Моделирование управления.
25. Декомпозиция подсистем на модули.
26. Модульность.

27. Особенности процесса синтеза программных систем.
28. Особенности этапа проектирования.
29. Структурирование системы.
30. Моделирование управления.
31. Декомпозиция подсистем на модули.
32. Модульность.
33. Языки описания архитектуры.
34. Факторы влияния: функциональность, эстетичный внешний вид, производительность.
35. Разработка интерфейса как процесс: проектирование, реализация, тестирование.
36. Формальные методы верификации.
37. Логико-алгебраические модели.
38. Исполнимые модели.
38. Динамические методы.
39. Мониторинг.
40. Тестирование.
41. Мониторинг формальных свойств ПО.
42. Основные отличия от других видов управления проектами.
43. Основные методы разработки программного обеспечения.
44. Сопутствующие процессы при управлении проектом.
45. Планирование, отслеживание и контроль за проектом.

Пример задания:

- 1) Модель жизненного цикла программного продукта.
- 2) Тестирование ПО.
- 3) Провести анализ предметной области «Аттестация обучающихся» с использованием UML.

6.2.2.1.2 Критерии оценивания

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Наличие глубоких знаний в объеме пройденного программного материала, правильные и уверенные действия по применению полученных знаний на практике, грамотное и логически стройное изложение материала при ответе, знание дополнительно	Наличие достаточно полных знаний программного материала, незначительные ошибки при ответах, правильное применение знаний на практике, четкое изложение материала. Умеет описывать требования к разрабатываемому ПО, проектировать его	Наличие частичного, не полного знания пройденного программного материала, дополняемое наводящими вопросами, незначительные ошибки при ответах, правильное применение знаний на практике, нечеткое изложение материала, наличие ошибок в практическом задании. Умеет проектировать	Наличие грубых ошибок при изложении пройденного программного материала, отсутствие ответов на наводящие вопросы, неумение применять знания на практике. Не умеет грамотно проектировать ПО, разбить задачу на модули, организовать процесс верификации.

рекомендованной литературы. Знает основные методы проектирования и разработки программного обеспечения, методы верификации ПО. Умеет описать требования к ПО, провести декомпозицию на модули, организовать тестирование.	и строить процесс тестирования программного обеспечения.	программное обеспечение, но допускает ошибки; процесс тестирования строится с недочетами.	
---	--	---	--

7 Основная учебная литература

1. Черткова Е. А. Программная инженерия. Визуальное моделирование программных систем [Электронный ресурс] : учебник для среднего профессионального образования / Е. А. Черткова, 2024. - 146.

[Сайт] – URL: <https://urait.ru/bcode/539955>

2. Белугина С. В. Разработка программных модулей программного обеспечения для компьютерных систем. Прикладное программирование : учебное пособие / С. В. Белугина, 2020. - 312 с

[Сайт] – URL: <https://e.lanbook.com/book/133920>

3. Маран, М. М. Программная инженерия : учебное пособие для вузов / М. М. Маран. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 196 с. — ISBN 978-5-8114-9323-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/189470>

[Сайт] – URL: <https://e.lanbook.com/book/189470>

4. Минакова, О. В. Программная инженерия. Основные принципы, методы и инструменты : учебник для вузов / О. В. Минакова. — Санкт-Петербург : Лань, 2024. — 212 с. — ISBN 978-5-507-49278-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/414989>

[Сайт] – URL: <https://e.lanbook.com/book/414989>

8 Дополнительная учебная литература и справочная

1. Программная инженерия : учебник по направлению подготовки 231000 "Программная инженерия" / Антипов В. А., Бубнов А. А., Пылькин А. Н. [и др.], 2014. - 281.

2. Лаврищева Е. М. Программная инженерия. Парадигмы, технологии и CASE-средства : учебник для вузов / Е. М. Лаврищева, 2016. - 280.

3. Лаврищева Е. М. Технология программирования и программная инженерия : учебник для вузов по направлению "Прикладная математика и физика", а также по другим математическим и естественнонаучным направлениям / Е. М. Лаврищева, 2017. - 431.
4. Минакова О. В. Программная инженерия. Основы принципы, методы и инструменты : учебник для вузов / О. В. Минакова, 2024. - 212.
5. Маран М. М. Программная инженерия : учебное пособие / М. М. Маран, 2018. - 194.
6. Вендров А. М. Проектирование программного обеспечения экономических информационных систем [UML, CASE] : учеб. для экон. вузов по специальностям "Прикладная информатика (по обл.)" и "Прикладная математика и информатика" / А. М. Вендров, 2000. - 347.

9 Ресурсы сети Интернет

1. <http://library.istu.edu/>
2. <https://e.lanbook.com/>

10 Профессиональные базы данных

1. <http://new.fips.ru/>
2. <http://www1.fips.ru/>

11 Перечень информационных технологий, лицензионных и свободно распространяемых специализированных программных средств, информационных справочных систем

1. Лицензионное программное обеспечение Системное программное обеспечение
2. Лицензионное программное обеспечение Пакет прикладных офисных программ
3. Лицензионное программное обеспечение Интернет-браузер
4. Свободно распространяемое программное обеспечение Star UML, пакет для UML-моделирования
5. Свободно распространяемое программное обеспечение Любая среда разработка приложений (свободно-распространяемая)
6. Свободно распространяемое программное обеспечение draw.io

12 Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения лекционных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.
2. Учебная аудитория для проведения лабораторных/практических (семинарских) занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска.

Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.