

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ
ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛУГАНСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ЛГПУ»)**

Структурное подразделение Институт физико-математического
образования, информационных и обслуживающих технологий
Кафедра информационных образовательных технологий и систем

УТВЕРЖДАЮ

Врио директора ИФМОИОТ

Е.А. Журавлева

2026 г.

Приложение к рабочей программе учебной дисциплины

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
для проведения текущего контроля и промежуточной аттестации
обучающихся по дисциплине
«Методология программной инженерии»**

По направлению подготовки 09.04.04 Программная инженерия
Профиль подготовки Программное обеспечение систем и комплексов
Квалификация выпускника – магистр
Форма обучения очная, заочная
Курс ОФО – 1 курс, ЗФО – 1 курс

Разработчик
Суворова Е.Ю.
канд. пед. наук, доцент кафедры
информационных технологий и систем

Заведующий кафедрой
Д.А. Капустин

Протокол от «13» августа 2026 г. № 11

Луганск, 2026

1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

1.1. Область применения

Фонд оценочных средств (ФОС) – неотъемлемая часть рабочей программы дисциплины (модуля) Методология программной инженерии и предназначен для контроля и оценки образовательных достижений студентов, освоивших программу дисциплины (модуля).

1.2. Цели и задачи фонда оценочных средств

Цель ФОС – установить соответствие уровня подготовки обучающегося требованиям ФГОС ВО бакалавриат / специалитет / магистратура по направлению подготовки 09.04.04 Программная инженерия, утвержденного приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г. № 932 (с изменениями и дополнениями).

1.3. Перечень компетенций, формируемых в процессе освоения основной образовательной программы

Процесс освоения дисциплины направлен на формирование следующих компетенций и индикаторов их достижения:

Код по ФГОС ВО	Индикатор достижения
Профессиональные	
ПК-2. Владение методами программной реализации распределенных информационных систем	ПК-2.1. Знать методы программной реализации распределенных информационных систем ПК-2.2. Уметь использовать методы программной реализации распределенных информационных систем ПК-2.3. Владеть навыками использования методов программной реализации распределенных информационных систем

1.4. Этапы формирования компетенций и средства оценивания уровня их сформированности

Этапы формирования компетенций	Компетенции	Контрольно-оценочные средства / способ оценивания
Тема 1. Введение в программную инженерию и жизненный цикл ПО.	ПК-2	Выполнение лабораторных работ
Тема 2. Модели жизненного цикла для разработки программных систем.	ПК-2	Выполнение лабораторных работ
Тема 3. Методы определения требований в программной инженерии.	ПК-2	Выполнение лабораторных работ
Тема 4. Методы анализа и построения моделей ПрО.	ПК-2	Выполнение лабораторных работ

Тема 5. Методы проектирования программных систем.	ПК-2	Выполнение лабораторных работ
Тема 6. Инженерия приложений и инженерия предметной области.	ПК-2	Выполнение лабораторных работ
Тема 7. Методы верификации и тестирования программ и систем.	ПК-2	Выполнение лабораторных работ
Тема 8. Методы интеграции, преобразования и изменения компонентов и данных.	ПК-2	Выполнение лабораторных работ
Тема 9. Модели качества и надежности в программной инженерии.	ПК-2	Выполнение лабораторных работ
Тема 10. Методы управления проектом, риском и конфигурацией.	ПК-2	Выполнение лабораторных работ
Текущая аттестация	ПК-2	Контрольная работа
Промежуточная аттестация	ПК-2	Экзамен

1.5. Описание показателей формирования компетенций

Код компетенции	Результаты сформированности
ПК-2. Владение методами программной реализации распределенных информационных систем	ПК-2.1. Знает методы программной реализации распределенных информационных систем ПК-2.2. Умеет использовать методы программной реализации распределенных информационных систем ПК-2.3. Владеет навыками использования методов программной реализации распределенных информационных систем

1.6. Критерии оценивания компетенций на разных этапах их формирования

Вид учебной работы	Количество баллов		
1-2 семестр / 1-3 триместр			
	ОФО	О-ЗФО	ЗФО
Оформление отчетов по лабораторным работам	30 баллов		
Работа на лабораторных занятиях	30 баллов		
Выполнение тестовых заданий	-		
Выполнение заданий самостоятельной работы	10 баллов		
экзамен	30 баллов		
Итого за семестр:	100 баллов		
Всего	100 баллов		

Накопительная система оценивания по 100-балльной шкале

Четырехбал- льная система оценивания экзамена	100- балльная шкала	Буквенная шкала, соответствующая 100- балльной шкале	Система оценивания зачета
Отлично	90–100	A – отлично – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом сформированы; все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	Зачтено
Хорошо	83–89	B – очень хорошо – теоретическое содержание курса освоено полностью, без пробелов; необходимые практические навыки работы с освоенным материалом в основном сформированы; все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному	
Хорошо	75–82	C – хорошо – теоретическое содержание курса освоено полностью; некоторые практические навыки работы с освоенным материалом сформированы недостаточно; все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	
Удовлетво- рительно	63–74	D – удовлетворительно – теоретическое содержание дисциплины освоено частично, но пробелы не носят существенного характера; необходимые практические навыки работы с освоенным материалом в основном сформированы; большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, содержат ошибки	
Удовлетво- рительно	50–62	E – посредственно – теоретическое содержание курса освоено частично; некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному	
Неудовлетво- рительно	21–49	FX – неудовлетворительно – теоретическое содержание курса освоено частично; необходимые практические навыки работы не сформированы; большинство предусмотренных программой обучения учебных заданий не выполнено либо качество	Не зачтено

		их выполнения оценено числом баллов, близким к минимальному; при дополнительной самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий	
Неудовлетворительно	0–20	F – неудовлетворительно – теоретическое содержание курса не освоено; необходимые практические навыки работы не сформированы; все выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к какому-либо значимому повышению качества выполнения учебных заданий	

2. КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА

2.1. Оценочные средства текущего контроля (типовые)

Вопросы для текущего контроля:

1. Как оцениваются задачи в Agile?
2. Что такое Use Case?
3. Что такое пользовательские сценарии (User Scenarios)?
4. Как проводится анализ требований?
5. Что такое модульное тестирование (Unit Testing)?
6. Что такое интеграционное тестирование?
7. Что такое диаграмма Ганта?
8. Что такое критический путь в проекте?
9. Что такое CMMI (Capability Maturity Model Integration)?
10. Какие уровни зрелости существуют в CMMI?
11. Какие виды документации существуют в программной инженерии?
12. Что такое техническое задание (ТЗ)?
13. Что такое Continuous Deployment (CD)?
14. Что такое безопасность ПО?
15. Какие принципы лежат в основе GDPR?
16. Какие инструменты используются для управления задачами?
17. Что такое Jira?
- 18.

2.2. Оценочные средства для промежуточной аттестации

Вопросы для проведения аттестации

1. Что такое программная инженерия?
2. Каковы основные цели программной инженерии?
3. Назовите основные принципы программной инженерии.
4. Что такое жизненный цикл программного обеспечения?
5. Какие этапы входят в жизненный цикл ПО?
6. Что такое методология разработки ПО?
7. Какие существуют подходы к разработке ПО?
8. В чем разница между программной инженерией и программированием?
9. Что такое качество программного обеспечения?
10. Какие факторы влияют на качество ПО?
11. Что такое каскадная модель (Waterfall)?
12. Какие преимущества и недостатки у каскадной модели?
13. Что такое итеративная модель разработки?
14. В чем отличие инкрементальной модели от итеративной?
15. Что такое спиральная модель?
16. Какие риски учитываются в спиральной модели?
17. Что такое V-модель?
18. Какие этапы входят в V-модель?
19. Что такое Agile-подход?
20. Какие принципы лежат в основе Agile?
21. Что такое Scrum?

22. Какие роли существуют в Scrum?
23. Что такое спринт в Scrum?
24. Какие артефакты используются в Scrum?
25. Что такое Kanban?
26. В чем отличие Kanban от Scrum?
27. Что такое пользовательские истории (User Stories)?
28. Как оцениваются задачи в Agile?
29. Что такое ежедневный стендап (Daily Standup)?
30. Какие метрики используются в Agile?
31. Что такое DevOps?
32. Какие инструменты используются для CI/CD?
33. Что такое контейнеризация?
34. В чем преимущества использования Docker?
35. Что такое оркестрация контейнеров?
36. Какие задачи решает Kubernetes?
37. Что такое инфраструктура как код (IaC)?
38. Какие инструменты используются для IaC?
39. Что такое требования к ПО?
40. Какие типы требований существуют?
41. Что такое функциональные требования?
42. Что такое нефункциональные требования?
43. Как документируются требования?
44. Что такое Use Case?
45. Что такое пользовательские сценарии (User Scenarios)?
46. Как проводится анализ требований?
47. Что такое трассируемость требований?
48. Какие инструменты используются для управления требованиями?
49. Что такое архитектура ПО?
50. Какие существуют стили архитектуры ПО?
51. Что такое микросервисная архитектура?
52. В чем преимущества микросервисной архитектуры?
53. Какие принципы лежат в основе REST?
54. Что такое тестирование ПО?
55. Какие виды тестирования вы знаете?
56. Что такое модульное тестирование (Unit Testing)?
57. Что такое интеграционное тестирование?
58. Что такое системное тестирование?
59. Что такое регрессионное тестирование?
60. Что такое нагрузочное тестирование?
61. Какие инструменты используются для автоматизации тестирования?
62. Что такое TDD (Test-Driven Development)?
63. В чем отличие TDD от BDD (Behavior-Driven Development)?
64. Какие метрики используются для оценки прогресса проекта?
- 65.)?
66. Какие уровни зрелости существуют в CMMI?
67. Какие виды документации существуют в программной инженерии?
68. Что такое техническое задание (ТЗ)?

- 69. Что такое спецификация требований?
- 70. Что такое архитектурная документация?
- 71. Что такое пользовательская документация?
- 72. Какие инструменты используются для документирования ПО?
- 73. Что такое Swagger/OpenAPI?
- 74. Как документируется API?
- 75. Что такое README файл?