

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

«14» января 2026 г.



Приложение к рабочей программе учебной дисциплины

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ  
для проведения текущего контроля и промежуточной аттестации  
обучающихся по дисциплине  
«Проектирование программного обеспечения»**

**По направлению подготовки 09.03.04 Программная инженерия**

**Профиль подготовки Программное обеспечение систем и комплексов**

**Квалификация выпускника – бакалавр**

**Форма обучения очная**

**Курс ОФО – 3 курс**

Разработчик

Суворова Е.Ю.

канд. пех. наук, доцент кафедры  
информационных образовательных  
технологий и систем

Заведующий кафедрой

Д.А. Капустин

Протокол от «13» января 2026 г. № 11

Луганск, 2026

# 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

## 1.1. Область применения

Фонд оценочных средств (ФОС) – неотъемлемая часть рабочей программы дисциплины (модуля) Проектирование программного обеспечения 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 09.03.04 Программная инженерия, утвержденного приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г. № 920 (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-3. Владеет концепциями и атрибутами качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роли людей, процессов, методов, инструментов и технологий обеспечения качества	ПК-3.1. Знать концепции и атрибуты качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роли человека-оператора, процессов, методов, инструментов и технологий обеспечения качества ПК-3.2. Уметь определять оптимальные методы и технологии разработки компьютерных программ, обеспечивающие достаточный уровень качества программного обеспечения (надежности, безопасности, удобства использования), с учетом роли человека-оператора, процессов, методов, инструментов и технологий обеспечения качества ПК-3.3. Владеть навыками разработки компьютерных программ, с заданными потребительскими свойствами с учётом комплекса условий их эксплуатации

## 1.4. Этапы формирования компетенций и средства оценивания уровня их сформированности

Этапы формирования компетенций	Компетенции	Контрольно-оценочные средства / способ оценивания
Тема 1. Введение в проектирование и архитектуру программных систем	ПК-3	Выполнение лабораторных работ
Тема 2. Общие концепции проектирования	ПК-3	Выполнение лабораторных работ

Тема 3. Объектно-ориентированное проектирование с использованием UML	ПК-3	Выполнение лабораторных работ
Тема 4. Проектирование с использованием структурных диаграмм	ПК-3	Выполнение лабораторных работ
Тема 5. Проектирование поведения систем	ПК-3	Выполнение лабораторных работ
Тема 6. Проектирование взаимодействия систем	ПК-3	Выполнение лабораторных работ
Тема 7. Документирование проектных решений	ПК-3	Выполнение лабораторных работ
Тема 8. Архитектурные стили, шаблоны. Плюсы и минусы различных архитектурных стилей	ПК-3	Выполнение лабораторных работ
Тема 9. Архитектурные стили сетевых и распределенных программных систем	ПК-3	Выполнение лабораторных работ
<b>Текущая аттестация</b>	ПК-3	Контрольная работа
<b>Промежуточная аттестация</b>	ПК-3	Экзамен

### 1.5. Описание показателей формирования компетенций

Код компетенции	Результаты сформированности
ПК-3. Владеет концепциями и атрибутами качества программного обеспечения (надежности, безопасности, удобства использования), в том числе роли людей, процессов, методов, инструментов и технологий обеспечения качества	<p>Знать: фундаментальные основы теории и технологии программной инженерии; основные модели жизненного цикла ПО; основы компонентно-ориентированной разработки</p> <p>Уметь: отличать инкрементальную разработку от спиральной, выделять функциональные требования для реализации в первом и последующих прототипах ПО; разрабатывать программное обеспечение в соответствии с принципами программной инженерии</p> <p>Владеть: навыками разработки программных систем в соответствии с принципами программной инженерии; инструментами для создания UML-диаграмм</p>

### 1.6. Критерии оценивания компетенций на разных этапах их формирования

Вид учебной работы	Количество баллов		
5 семестр / 7-9 триместр			
	ОФО	О-ЗФО	ЗФО
Защита лабораторных работ	50 баллов		50 баллов
Выполнение заданий самостоятельной работы	30 баллов		30 баллов

экзамен (зачет)	20 баллов		20 баллов
<b>Итого за семестр:</b>	100 баллов		
<b>Всего</b>	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	<b>А</b> – 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>В</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	<b>С</b> – 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	<b>Д</b> – 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	<b>Е</b> – 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	

Неудовлетворительно	<b>21–49</b>	<b>FX</b> – неудовлетворительно – теоретическое содержание курса освоено частично; необходимые практические навыки работы не сформированы; большинство предусмотренных программой обучения учебных заданий не выполнено либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительной самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий	Не зачтено
Неудовлетворительно	<b>0–20</b>	<b>F</b> – 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	

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФГБОУ ВО «ЛГПУ»  
ИНСТИТУТ ФИЗИКО-МАТЕМАТИЧЕСКОГО ОБРАЗОВАНИЯ, ИНФОРМАЦИОННЫХ  
И ОБСЛУЖИВАЮЩИХ ТЕХНОЛОГИЙ  
20\_\_ – 20\_\_ учебный год

Направление подготовки (специальность): 09.03.04 Программная инженерия

Профиль подготовки Программное обеспечение систем и комплексов

курс / форма обучения: 3 курс, ОФО

Семестр 5

Учебная дисциплина: Проектирование программного обеспечения

**ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1**

1. Общие свойства и различия пакетов и подсистем.
2. Какие этапы включает в себя процесс верификации и валидации ПО.
3. Что такое шаблон проектирования "Одиночка" (Singleton)

Утверждено на заседании кафедры информационных образовательных технологий и систем

Протокол №\_\_ от \_\_\_\_ г.

Заведующий кафедрой ИОТС \_\_\_\_\_ Капустин Д.А.

(подпись)

Экзаменатор

\_\_\_\_\_

(подпись)

доцент Суворова Е.Ю.

(должность, ФИО преподавателя)

## **2. КОНТРОЛЬНО-ОЦЕНОЧНЫЕ СРЕДСТВА**

### **2.1. Оценочные средства текущего контроля (типовые)**

Вопросы для текущего контроля:

1. Место, цели и задачи этапа проектирования в процессе разработки программных систем.
2. Типы моделей и основные задачи, решаемые с их использованием.
3. Основные методологии проектирования.
4. Принципы построения объектной модели (по Гр. Бучу). Определения. Основные элементы объектной модели: объект, индивидуальность, поведение объекта, состояние объекта, класс, атрибут, операция, компонент, интерфейс, пакет и подсистема.
5. Виды связей между элементами моделей: соединение, ассоциация, агрегация, композиция, зависимость, реализация, обобщение.
6. Полиморфизм. Направление ассоциации, мощности полюсов. Характеристики полюсов ассоциаций с мощностью «\*». Квалификаторы. N-арные ассоциации ( $N > 2$ ), классы ассоциаций. Определения и примеры.
7. Общие свойства и различия пакетов и подсистем.
8. Архитектурный анализ, его цели, содержание, исполнители и рабочие продукты.
9. Соглашения моделирования. Механизмы анализа. Идентификация ключевых абстракций.
10. Архитектурные уровни. Понятие образца и способ его описания. Образцы анализа: «уровни», «модель-представление-управление», «каналы и фильтры». Примеры.

### **2.2. Оценочные средства для промежуточной аттестации**

Вопросы для проведения аттестации

1. Жизненный цикл ПО. Процессы жизненного цикла согласно ISO 12207. Модель жизненного цикла, стадия, контрольная точка (веха). Примеры моделей: водопадная, основанная на формальных преобразованиях, пошаговая итерационная, спиральная. Сравнение разных моделей между собой, их достоинства и недостатки.
2. Понятие архитектуры программного обеспечения. Архитектурные представления. Модель «4+1».
3. Принципы построения объектной модели (по Гр. Бучу). Определения. Основные элементы объектной модели:
4. Виды связей между элементами моделей: соединение, ассоциация, агрегация, композиция, зависимость, реализация, обобщение. Полиморфизм. Направление ассоциации, мощности полюсов.
5. Варианты использования. Описание варианта использования. Виды сценариев вариантов использования. Диаграммы вариантов использования, их элементы и связи между элементами. Примеры.

6. Диаграммы взаимодействия UML, их элементы и связи между элементами. Виды диаграмм взаимодействия и область их применения. Примеры.

7. Диаграммы классов UML, их элементы и связи между элементами. Область применения. Примеры.

8. Диаграммы состояний, их элементы и связи между элементами. Область применения. Примеры.

9. Диаграммы деятельности, их элементы и связи между элементами. Область применения. Примеры.

10. Диаграммы компонентов и диаграммы размещения, их элементы и связи между элементами. Область применения. Примеры.

11. Определение бизнес-процесса. Моделирование бизнес-процессов, его цели, содержание, исполнители и рабочие продукты.

12. Модель бизнес процессов, ее элементы, связи, диаграммы. Примеры. Переход от модели бизнес-процессов к модели вариантов использования.

13. Модель бизнес анализа, ее элементы, связи, диаграммы. Примеры. Переход от модели бизнес-анализа к модели анализа системы.

14. Определение требования. Виды требований. Описание требований.

15. Процесс определения требований, его цели, содержание, исполнители и рабочие продукты.

16. Архитектурный анализ, его цели, содержание, исполнители и рабочие продукты. Соглашения моделирования. Механизмы анализа.

17. Идентификация ключевых абстракций. Архитектурные уровни. Понятие образца и способ его описания. Образцы анализа: «уровни», «модель-представление-управление», «каналы и фильтры». Примеры.

18. Проектирование архитектуры системы, его цели и содержание, исполнители и рабочие продукты. Выявление проектных классов, пакетов, подсистем и интерфейсов. Проектные механизмы и механизмы реализации. Примеры.

19. Понятие образца и способ его описания. Образцы проектирования (абстрактная фабрика, фабричный метод, адаптер, компоновщик, мост, фасад, заместитель, цепочка обязанностей, итератор, стратегия, декоратор), Их классификация. Примеры использования образцов проектирования.

20. Проектирование структуры потоков управления и конфигурации системы. Цели, содержание, исполнители и рабочие продукты. Примеры.

21. Проектирование классов, его цели и содержание, исполнители и рабочие продукты. Примеры.

22. Отображение объектной модели в реляционную модель. Примеры.

23. Язык OCL. Основные понятия, элементы выражений, навигация по ассоциациям, коллекции, операции над коллекциями, в том числе let и iterate. Примеры. Наследование ограничений.

24. Технология создания программного обеспечения. Основные определения. Rational Unified Process (RUP). Основные принципы RUP. Общее представление RUP. Стадии жизненного цикла RUP и их содержание. Процессы жизненного цикла RUP и их содержание.

25. Что включает в себя процесс проектирования программного обеспечения?

26. Какие этапы присутствуют в процессе проектирования ПО?
27. Что такое функциональные требования к программному обеспечению?
28. Чем отличаются функциональные и нефункциональные требования к ПО?
29. Какие методы анализа требований к ПО вы знаете?
30. Что такое диаграмма прецедентов в UML и как она используется при проектировании ПО?
31. Какие шаблоны проектирования вы можете назвать?
32. В чем заключается паттерн проектирования "Фабричный метод"?
33. Какие основные принципы SOLID вы знаете и как они применяются при проектировании ПО?
34. Что такое принцип единственной ответственности?
35. Какие существуют виды связей между классами в UML и что они означают?
36. Какие преимущества и недостатки имеют объектно-ориентированные методы проектирования ПО?
37. Что такое диаграмма классов в UML и как она используется при проектировании ПО?
38. Что такое шаблон проектирования "Одиночка" (Singleton)?
39. Какие основные шаблоны проектирования используются при проектировании пользовательского интерфейса?
40. Какие шаблоны проектирования применяются при работе с базами данных?
41. Что такое многопоточность в программировании и как она учитывается при проектировании ПО?
42. Какие методы тестирования ПО используются при проектировании?
43. Что такое модульное тестирование и как оно применяется при разработке ПО?
44. Какие принципы безопасности следует учитывать при проектировании ПО?
45. Какие методы и инструменты используются для моделирования и проектирования ПО?
46. Какие существуют модели жизненного цикла разработки ПО и в чем их отличия?
47. Что такое архитектурные стили и как они влияют на проектирование ПО?
48. Какие факторы следует учитывать при выборе архитектуры программного обеспечения?
49. Что такое MVP (Минимально жизнеспособный продукт) и как он используется при проектировании ПО?
50. Какие этапы включает в себя процесс разработки программного обеспечения с использованием Agile методологии?
51. Какие принципы лежат в основе Agile методологии разработки ПО?
52. Какие инструменты и технологии широко используются при проектировании и разработке ПО?

53. Что такое Continuous Integration (непрерывная интеграция) и как она применяется в проектировании ПО?
54. Что такое Continuous Deployment (непрерывное развертывание) и как оно связано с проектированием ПО?
55. Какие основные этапы и методы управления конфигурацией ПО существуют?
56. Какие метрики используются для оценки качества проектирования ПО?
57. Что такое архитектурный дизайн и как он отличается от детального дизайна ПО?
58. Какие основные принципы проектирования пользовательского интерфейса вы можете назвать?
59. Что такое UX-дизайн и как он влияет на проектирование ПО?
60. Какие методы тестирования пользовательского интерфейса используются при проектировании ПО?
61. Какие факторы влияют на выбор технологического стека при проектировании ПО?
62. Каким образом архитектурные решения влияют на масштабируемость и производительность ПО?
63. Какие методы обеспечения безопасности информации используются при проектировании ПО?
64. Какие этапы включает в себя процесс тестирования программного обеспечения?
65. Что такое модульное тестирование и как оно отличается от интеграционного тестирования?
66. Какие инструменты используются для автоматизации тестирования ПО?
67. Какие виды тестирования производительности используются при проектировании ПО?
68. Каким образом проектирование для масштабируемости отличается от проектирования для производительности ПО?
69. Какие методы обеспечения безопасности информации используются при проектировании ПО?
70. Что такое дизайн пользовательского опыта (UX design) и как он влияет на проектирование ПО?