

Министерство науки и высшего образования Российской Федерации  
**Муромский институт (филиал)**  
федерального государственного бюджетного образовательного учреждения высшего образования  
**«Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»**  
(МИ ВлГУ)

Кафедра *ПИИ*

«УТВЕРЖДАЮ»  
Заместитель директора по УР  
Д.Е. Андрианов  
\_\_\_\_\_ 25.05.2021

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

*Проектирование программного обеспечения*

**Направление подготовки**

*09.03.04 Программная инженерия*

**Профиль подготовки**

*Программная инженерия*

Семестр	Трудоемкость, час./зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
<b>5</b>	<b>144 / 4</b>	<b>24</b>		<b>28</b>	<b>4,4</b>	<b>0,35</b>	<b>56,75</b>	<b>60,6</b>	<b>Экз.(26,65)</b>
<b>Итого</b>	<b>144 / 4</b>	<b>24</b>		<b>28</b>	<b>4,4</b>	<b>0,35</b>	<b>56,75</b>	<b>60,6</b>	<b>26,65</b>

Муром, 2021 г.

## 1. Цель освоения дисциплины

Цель дисциплины: подготовка бакалавров, способных решать задачи проектирования, анализа, документирования и сопровождения архитектуры программных систем

## 2. Место дисциплины в структуре ОПОП ВО

Курс базируется на знаниях, полученных студентами в процессе изучения дисциплин: «Основы алгоритмизации и программирования», «Объектно-ориентированное программирование», «Организация баз данных». На дисциплине «Проектирование программного обеспечения» базируется изучение дисциплин: «Тестирование программного обеспечения», «Разработка корпоративных приложений», «Протоколы и интерфейсы информационных систем», «Документирование, сертификация и стандартизация программного обеспечения», «Разработка приложений для мобильных операционных систем», «Корпоративные информационные системы».

## 3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;	ОПК-6.3 Понимает методы проектирования программного обеспечения	Знает современные технологии разработки и документирования ПО (ОПК-6.3) Умеет определять показатели атрибутов качества ПО и использовать формальные методы конструирования программного обеспечения (ОПК-6.3) Владеет инструментальными средствами, обеспечивающими процесс проектирования и документирования программного обеспечения (ОПК-6.3)	задачи, вопросы к устному опросу

## 4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 4 зачетные единицы, 144 часа.

### 4.1. Форма обучения: очная

Уровень базового образования: среднее общее.

Срок обучения 4г.

#### 4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником						Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)	
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация			Контроль
1	Жизненный цикл ПО. Разработка требований к ПО	5	12		4					40	устный опрос, отчет по лабораторной работе
2	Инструментальные средства моделирования систем	5	12		24					20,6	устный опрос, отчет по лабораторной работе
Всего за семестр		144	24		28			4,4	0,35	60,6	Экз.(26,65)
Итого		144	24		28			4,4	0,35	60,6	26,65

#### 4.1.2. Содержание дисциплины

##### 4.1.2.1. Перечень лекций

##### Семестр 5

*Раздел 1. Жизненный цикл ПО. Разработка требований к ПО*

##### Лекция 1.

Роль требований в задаче внедрения ПО. Уровни требований. Системные требования и требования к программному обеспечению (2 часа).

##### Лекция 2.

Функциональные, нефункциональные требования и характеристики продукта. Методологии и стандарты, регламентирующие работу с требованиями. Свойства требований (2 часа).

##### Лекция 3.

Процесс анализа требований. Анализ требований, бизнес-анализ, анализ проблемной области. Методологии бизнес-анализа. Требования и архитектура ПО. Анализ требований и другие рабочие потоки программной инженерии (2 часа).

#### **Лекция 4.**

Источники требований. Стратегии выявления требований. Прототипирование (2 часа).

#### **Лекция 5.**

Видение продукта и границы проекта. Концепция в ГОСТ РФ. Видение в RUP. Видение / рамки в MSF. Видение продукта и границы проекта. Акторы и варианты использования. Глоссарий. Спецификация варианта использования (2 часа).

#### **Лекция 6.**

Документирование требований. Структура ТЗ. Описание требований к системе. Документирование требований в RUP, MSF. Верификация и валидация (2 часа).

*Раздел 2. Инструментальные средства моделирования систем*

#### **Лекция 7.**

Модели UML, поясняющие функциональность системы. Диаграмма вариантов использования. Диаграмма действий, диаграмма состояний. Диаграммы UML, поясняющие внутреннее устройство системы. Диаграмма потоков данных. Цели прототипирования. Классификация прототипов. Горизонтальный и вертикальный прототипы. Одноразовый и эволюционные прототипы. Бумажный прототип. Раскадровка. Иллюстрированные сценарии прецедентов. Ориентиры (2 часа).

#### **Лекция 8.**

Принципы проектирования классов - SOLID (2 часа).

#### **Лекция 9.**

Методы моделирования бизнес-процессов (2 часа).

#### **Лекция 10.**

Функциональные диаграммы, диаграммы потоков данных (2 часа).

#### **Лекция 11.**

Диаграммы последовательностей, кооперации, деятельности (2 часа).

#### **Лекция 12.**

Диаграммы развёртывания (2 часа).

### **4.1.2.2. Перечень практических занятий**

Не планируется.

### **4.1.2.3. Перечень лабораторных работ**

#### **Семестр 5**

*Раздел 1. Жизненный цикл ПО. Разработка требований к ПО*

#### **Лабораторная 1.**

Разработка требований. Диаграмма вариантов использования, спецификация процессов (4 часа).

*Раздел 2. Инструментальные средства моделирования систем*

#### **Лабораторная 2.**

Методы моделирования бизнес-процессов. Функциональное моделирование (4 часа).

#### **Лабораторная 3.**

Диаграмма состояний (4 часа).

#### **Лабораторная 4.**

Диаграммы классов, объектов (4 часа).

#### **Лабораторная 5.**

Диаграммы последовательностей, кооперации (4 часа).

#### **Лабораторная 6.**

Диаграммы деятельности, развертывания (4 часа).

#### **Лабораторная 7.**

Выбор методов тестирования программного средства (4 часа).

### **4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы**

Перечень тем, вынесенных на самостоятельное изучение:

1. Понятие программной архитектуры.
2. Зарождение концепции архитектуры программных систем.
3. Ретроспектива программных архитектур.
4. История развития языков описания архитектур.
5. Архитектура системы: основные определения.
6. Атрибут качества архитектуры системы.
7. Бизнес-архитектура и архитектура информации.
8. Управление и контроль процесса разработки архитектуры.
9. Варианты использования и проектирование архитектуры системы.
10. Интегрированная концепция и уровни абстракции.
11. Содержание стандартов IEEE 1016-1998 и IEEE 1471-2000.
12. Функциональные и нефункциональные требования.
13. Категории и обзор возможностей CASE-средств.
14. Инструментальные средства и мониторинг технологий.
15. Управление ресурсами программной системы.
16. Методика Test Case Oriented Design.
17. Выбор методов тестирования. Стратегии проектирования тестовых наборов.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

#### **4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР**

Не планируется.

#### **4.1.2.6. Примерный перечень тем курсовых работ (проектов)**

Не планируется.

## 4.2 Форма обучения: заочная

Уровень базового образования: среднее общее.

Срок обучения 5л.

Семестр	Трудоёмкость, час./зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
7	180 / 5	8		12	4	0,6	24,6	146,75	Экз.(8,65)
<b>Итого</b>	<b>180 / 5</b>	<b>8</b>		<b>12</b>	<b>4</b>	<b>0,6</b>	<b>24,6</b>	<b>146,75</b>	<b>8,65</b>

### 4.2.1. Структура дисциплины

№ п/п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Жизненный цикл ПО. Разработка требований к ПО	7	6		4					96	устный опрос, отчет по лабораторной работе
2	Инструментальные средства моделирования систем	7	2		8					50,75	устный опрос
Всего за семестр		180	8		12	+		4	0,6	146,75	Экз.(8,65)
<b>Итого</b>		<b>180</b>	<b>8</b>		<b>12</b>			<b>4</b>	<b>0,6</b>	<b>146,75</b>	<b>8,65</b>

### 4.2.2. Содержание дисциплины

#### 4.2.2.1. Перечень лекций

##### Семестр 7

Раздел 1. Жизненный цикл ПО. Разработка требований к ПО

##### Лекция 1.

Роль требований в задаче внедрения ПО. Уровни требований. Системные требования и требования к программному обеспечению (2 часа).

## **Лекция 2.**

Функциональные, нефункциональные требования и характеристики продукта. Методологии и стандарты, регламентирующие работу с требованиями. Свойства требований (2 часа).

## **Лекция 3.**

Процесс анализа требований. Анализ требований, бизнес-анализ, анализ проблемной области. Методологии бизнес-анализа. Требования и архитектура ПО. Анализ требований и другие рабочие потоки программной инженерии (2 часа).

*Раздел 2. Инструментальные средства моделирования систем*

## **Лекция 4.**

Методы моделирования бизнес-процессов. Функциональные диаграммы, диаграммы потоков данных (2 часа).

### **4.2.2.2. Перечень практических занятий**

Не планируется.

### **4.2.2.3. Перечень лабораторных работ**

#### **Семестр 7**

*Раздел 1. Жизненный цикл ПО. Разработка требований к ПО*

#### **Лабораторная 1.**

Диаграмма вариантов использования, спецификация процессов (4 часа).

*Раздел 2. Инструментальные средства моделирования систем*

#### **Лабораторная 2.**

Методы моделирования бизнес-процессов. Функциональное моделирование (4 часа).

#### **Лабораторная 3.**

Диаграмма состояний (4 часа).

### **4.2.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы**

Перечень тем, вынесенных на самостоятельное изучение:

1. Понятие программной архитектуры.
2. Зарождение концепции архитектуры программных систем.
3. Ретроспектива программных архитектур.
4. История развития языков описания архитектур.
5. Архитектура предприятия: основные определения.
6. Интегрированная концепция и уровни абстракции.
7. Элементы архитектуры предприятия.
8. Бизнес-архитектура и архитектура информации.
9. Управление и контроль процесса разработки архитектуры.
10. Инструментальные средства и мониторинг технологий.
11. Содержание стандартов IEEE 1016-1998 и IEEE 1471-2000.
12. Функциональные и нефункциональные требования.
13. Варианты использования и проектирование архитектуры системы.
14. Атрибут качества архитектуры системы.
15. Обнаружение и предотвращение неисправности.
16. Управление ресурсами программной системы.
17. Методика Test Case Oriented Design.
18. Выбор методов тестирования. Стратегии проектирования тестовых наборов.
19. Видение продукта и границы проекта. Концепция в ГОСТ РФ. Видение в RUP.

Видение / рамки в MSF. Видение продукта и границы проекта. Акторы и варианты использования. Глоссарий. Спецификация варианта использования.

20. Модели UML, поясняющие функциональность системы. Диаграмма вариантов использования. Диаграмма действий, диаграмма состояний. Диаграммы UML, поясняющие внутреннее устройство системы. Диаграмма потоков данных. Цели прототипирования.

Классификация прототипов. Горизонтальный и вертикальный прототипы. Одноразовый и эволюционные прототипы. Бумажный прототип. Раскадровка. Иллюстрированные сценарии прецедентов. Ориентиры.

21. Документирование требований. Структура ТЗ. Описание требований к системе. Документирование требований в RUP, MSF. Верификация и валидация.

22. Принципы проектирования классов - SOLID.

23. Методы моделирования бизнес-процессов.

24. Функциональные диаграммы, диаграммы потоков данных.

25. Диаграммы последовательностей, кооперации, деятельности.

26. Диаграммы развёртывания.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

#### **4.2.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР**

1. Диаграммы классов, объектов.

2. Диаграммы последовательностей, кооперации.

3. Диаграммы деятельности, развертывания.

4. Выбор методов тестирования программного средства.

#### **4.2.2.6. Примерный перечень тем курсовых работ (проектов)**

Не планируется.

### **5. Образовательные технологии**

В процессе изучения дисциплины применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении лабораторных работ применяется имитационный или симуляционный подход, когда преподавателем разбирается на конкретном примере проблемная ситуация, все шаги решения задачи студентам демонстрируются при помощи мультимедийной техники. Затем студенты самостоятельно решают аналогичные задания, осваивая знания концепции и стратегии архитектурного проектирования и конструирования программного продукта, основы теории организации и применения баз данных, понимание основ групповой динамики, психологии и профессионального поведения, специфичных для программной инженерии и понимание особенностей эволюционной деятельности, как с технической точки зрения, так и с точки зрения бизнеса.

Во время выполнения лабораторных работ каждый студент выполняет индивидуальное задание, тем самым формируется умение конструировать программное обеспечение, разрабатывать основные программные документы.

### **6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.**

Фонды оценочных материалов (средств) приведены в приложении.

### **7. Учебно-методическое и информационное обеспечение дисциплины.**

#### **7.1. Основная учебно-методическая литература по дисциплине**

1. Введение в программные системы и их разработку : учебное пособие / С.В. Назаров [и др.].. — Москва, Саратов : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 649 с. — ISBN 978-5-4497-0312-5. — Текст : электронный // IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/89429.html> (дата обращения: 26.11.2022). — Режим доступа: для авторизир. пользователей - <https://www.iprbookshop.ru/89429.html>

2. Иванова, О. Г. Методы и средства проектирования информационных систем и технологий. Основы UML : учебное пособие / О. Г. Иванова, Ю. Ю. Громов. — Тамбов :

Тамбовский государственный технический университет, ЭБС АСВ, 2020. — 80 с. — ISBN 978-5-8265-2308-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/115768.html> (дата обращения: 12.12.2022). — Режим доступа: для авторизир. пользователей - <https://www.iprbookshop.ru/115768.html>

3. Доррер, Г. А. Методология программной инженерии : учебное пособие / Г. А. Доррер. — Красноярск : Сибирский государственный университет науки и технологий имени академика М.Ф. Решетнева, 2021. — 190 с. — URL: <https://www.iprbookshop.ru/116641.html> - <https://www.iprbookshop.ru/116641.html>

4. Мейер, Б. Объектно-ориентированное программирование и программная инженерия / Б. Мейер. — М. : ИНТУИТ, Ай Пи Эр Медиа, 2019. — 285 с. — URL: <https://www.iprbookshop.ru/79706.html> - <https://www.iprbookshop.ru/79706.html>

5. Суханов, М. Б. Программная инженерия : учебное пособие / М. Б. Суханов. — СПб. : Санкт-Петербургский государственный университет промышленных технологий и дизайна, 2018. — 146 с. — URL: <https://www.iprbookshop.ru/102465.html> - <https://www.iprbookshop.ru/102465.html>

## **7.2. Дополнительная учебно-методическая литература по дисциплине**

1. Малышева Е.Н. Проектирование информационных систем. Раздел 5. Индустриальное проектирование информационных систем. Объектно-ориентированная Case-технология проектирования информационных систем [Электронный ресурс]: учебное пособие/ Малышева Е.Н.— Электрон. текстовые данные.— Кемерово: Кемеровский государственный институт культуры, 2009.— 70 с.— Режим доступа: <http://www.iprbookshop.ru/22067>.— ЭБС «IPRbooks», по паролю - <https://www.iprbookshop.ru/22067.html>

2. Бабич А.В. Введение в UML : учебное пособие / Бабич А.В.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2022. — 198 с. — ISBN 978-5-4497-1637-8. — Текст : электронный // IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/120473.html> (дата обращения: 26.11.2022). — Режим доступа: для авторизир. пользователей - <https://www.iprbookshop.ru/120473.html>

3. Носова Л.С. Case-технологии и язык UML : учебно-методическое пособие / Носова Л.С.. — Челябинск, Саратов : Южно-Уральский институт управления и экономики, Ай Пи Эр Медиа, 2019. — 67 с. — ISBN 978-5-4486-0670-0. — Текст : электронный // IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/81479.html> (дата обращения: 26.11.2022). — Режим доступа: для авторизир. пользователей. - DOI: <https://doi.org/10.23682/81479> - <https://www.iprbookshop.ru/81479.html>

4. Савельев А.О. Проектирование и разработка веб-приложений на основе технологий Microsoft : учебное пособие / Савельев А.О., Алексеев А.А.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2022. — 418 с. — ISBN 978-5-4497-1650-7. — Текст : электронный // IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/120486.html> (дата обращения: 26.11.2022). — Режим доступа: для авторизир. пользователей - <https://www.iprbookshop.ru/120486.html>

## **7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем**

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института ([www.mivlgu.ru/iop](http://www.mivlgu.ru/iop)), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

Информационно-образовательный портал МИ ВлГУ - <https://www.mivlgu.ru/iop/>

Электронная библиотечная система iprBooks.ru - <http://www.iprbooks.ru>

Электронная библиотека «ЭВРИКА» - <https://evrika.mivlgu.ru/>

Электронная библиотека ВлГУ - <https://dspace.www1.vlsu.ru/>

Программное обеспечение:

LibreOffice (Mozilla Public License v2.0)

Microsoft Visual Studio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

Microsoft Visio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

#### **7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины**

[iprbookshop.ru](http://iprbookshop.ru)

[mivlgu.ru](http://mivlgu.ru)

[iprbooks.ru](http://iprbooks.ru)

[mivlgu.ru/iop](http://mivlgu.ru/iop)

#### **8. Материально-техническое обеспечение дисциплины**

Лаборатория системного и прикладного программирования

6 шт. компьютеров Intel Core i5, 3500 MHz/ ОЗУ 6Gb/ SSD-512Gb/ LG 22'; 6 шт. персональных компьютеров Digitech (комплект2) Intel Core i5 3000 MHz/ DDR-4 12Gb/ SSD-512Gb/ Philips 21eb; проектор NEC V300X 3D; экран проекционный настенный Lumien Master Picture; маршрутизатор Gigabit Switch TEG-S16S; макет системы мобильного мониторинга; лабораторный стенд для изучения микроконтроллера; роботизированная платформа IE-POP-BOT; аппаратно-программный комплекс «Изучение принципов построения и исследования инфокоммуникационных локальных сетей». Маркерная доска. Доступ к сети Интернет.

#### **9. Методические указания по освоению дисциплины**

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводится в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на лабораторную работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института

Самостоятельная работа оказывает важное влияние на формирование личности будущего бакалавра, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Форма заключительного контроля при промежуточной аттестации – экзамен. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и

своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.03.04 Программная инженерия* и профилю подготовки *Программная инженерия*  
Рабочую программу составил *Холкина Н.Е.*\_\_\_\_\_

Программа рассмотрена и одобрена на заседании кафедры *ПИИ*

протокол № 13 от 14.05.2021 года.

Заведующий кафедрой *ПИИ* \_\_\_\_\_ *Жизняков А.Л.*  
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 24.05.2021 года.

Председатель комиссии ФИТР \_\_\_\_\_ *Рыжкова М.Н.*  
(Подпись) (Ф.И.О.)

**Фонд оценочных материалов (средств) по дисциплине**  
Проектирование программного обеспечения

**1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине**

Темы для подготовке к текущему контролю в форме собеседования:

1. Понятие жизненного цикла ПО. Стандарты и проблемы жизненного цикла ПО.
2. Жизненный цикл в соответствии со стандартом ISO 12207. Структура жизненного цикла. Основные процессы.
3. Жизненный цикл в соответствии со стандартом ISO 15504. Классификация процессов. Категория процессов Потребитель-поставщик.
4. Жизненный цикл в соответствии со стандартом ISO 15504. Категории процессов «Инженерные процессы», «Вспомогательные процессы»
5. Жизненный цикл в соответствии со стандартом ISO 15504. Категории процессов «Управленческие процессы», «Организационные процессы».
6. Понятие Модели жизненного цикла программного продукта. Что является схемой модели ЖЦ ПО?
7. Каскадная модель. Принципы. Основные фазы. Преимущества и недостатки. Применимость.
8. Спиральная модель. Принципы. Схема. Преимущества и недостатки. Применимость.
9. Итерационная модель. Принципы. Схема. Преимущества и недостатки. Применимость.
10. V-образная модель. Принципы. Схема. Преимущества и недостатки. Применимость.
11. Инкрементная модель. Принципы. Схема. Преимущества и недостатки. Применимость.
12. Модель быстрого прототипирования. Принципы. Схема. Преимущества и недостатки. Применимость.
13. Модель жизненного цикла MSF. Базовые концепции и принципы.
14. MSF. Компромиссы. Симбиоз итеративного и фазового подходов. Вехи.
15. MSF. Проектная группа. Фазы жизненного цикла. Основные фазы. Преимущества и недостатки. Применимость.
16. Модель жизненного цикла RUP. Структура RUP. Преимущества и недостатки. Применимость.
17. Модель жизненного цикла RUP. Соответствие рабочих процессов и диаграмм UML.
18. Модель Extreme Programming. Принципы. Схема. Основные фазы.
19. Методология SCRUM. Роли и их назначение. Артефакты. Спринты.
20. Сравнение процессов разработки: каскадная модель, спиральная, инкрементная.
21. Понятие надежности ПО.

**Общее распределение баллов текущего контроля по видам учебных работ для студентов**

Рейтинг-контроль 1	Устный опрос, отчеты по лабораторной работе	15
Рейтинг-контроль 2	Устный опрос, отчеты по лабораторной работе	15
Рейтинг-контроль 3	Устный опрос, отчеты по лабораторной работе	18
Посещение занятий студентом	Отметка в журнале посещений	5

Дополнительные баллы (бонусы)	за своевременную защиту всех лабораторных работ	7
Выполнение семестрового плана самостоятельной работы	нет	0

## 2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.

Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

Для проведения экзаменационного тестирования используются задания в тестовой форме. Примеры заданий приведены далее.

### Методические материалы, характеризующие процедуры оценивания

На основе типовых заданий программным комплексом информационно-образовательного портала МИ ВлГУ формируются в автоматическом режиме тестовые задания для студентов. Программный комплекс формирует индивидуальные задания для каждого зарегистрированного в системе студента и устанавливает время прохождения тестирования. Результатом тестирования является процент правильных ответов, с учетом индивидуального семестрового рейтинга студента формируется итоговая оценка.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	Уровень сформированности компетенций
Более 80	«Отлично»	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	<b>Высокий уровень</b>
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, 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	<b>Продвинутый уровень</b>

50-65	«Удовлетворительно»	Содержание курса 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, возможно, содержат ошибки	<i><b>Пороговый уровень</b></i>
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	<i><b>Компетенции не сформированы</b></i>

### **3. Задания в тестовой форме по дисциплине**

Примеры заданий:

Методология проектирования, соединяющая в себе процесс объектной декомпозиции и приёмы представления логической и физической, а также статической и динамической моделей проектируемой системы

- Объектно-ориентированное проектирование
- Объектно-ориентированный анализ
- Процедурно-ориентированное программирование
- Алгоритмическое программирование

Какой подход НЕ относится к основным подходам к организации процесса создания и использования программного средства

- Комплексное программирование
- Водопадный подход
- Прототипирование
- Исследовательское программирование

Абстракция, которая позволяет представить фактически неограниченный набор различных вычислений одной программой, которая есть абстракция всех этих наборов, называется абстракция через ... (параметризацию)

Аббревиатура методологии структурного анализа и проектирования, с представлением систем в виде иерархии функций с использованием специального графического языка описания моделей систем, интегрирующая процесс моделирования, управления конфигурацией проекта и руководство проектом: ... (SADT)

Под жизненным циклом программного средства (ПС) понимают

- период разработки и эксплуатации, начиная от момента возникновения замысла ПС и заканчивая прекращением всех видов его использования
- момент возникновения ПС и момент непосредственной его эксплуатации
- использование ПС для решения практических задач на компьютере путем выполнения ее программ
- хранение, внедрение и сопровождение ПС

На диаграмме потоков данных первичные информационные потоки порождаются

- внешними сущностями
- работами

- хранилищами данных
- подсистемами

Аббревиатура языка графического описания для объектного моделирования в области разработки программного обеспечения, для моделирования бизнес-процессов, системного проектирования и отображения организационных структур: ... (UML)

В функциональную спецификацию программной системы входит:

- описания внешней информационной среды, к которой должны применяться программное обеспечение
- описание нежелательных (исключительных) ситуаций, которые могут возникнуть при выполнении программ, и реакций на эти ситуации, которые должны обеспечить соответствующие программы.
- критерии качества программного средства

Упрощенная версия программного средства (ПС) для выявления действительных потребностей пользователей называется ... ПС (ответ в именительном падеже): (прототип)

В технологии проектирования SADT диаграммы DFD определяют для предметной области потоки ... (чего?) (данных)

В результате объектно-ориентированной разработки проекта с использованием UML для моделирования поведения объектов системы при переходе из одного состояния в другое формируются диаграмма ... (чего?): (состояния)

Критерий качества, определяющий способность программного средства безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени с достаточно большой вероятностью, это – ... (ответ в именительном падеже) (надежность)

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/question/edit.php?courseid=3011>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.