

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

**УТВЕРЖДЕНО**  
решением Ученого совета факультета математики,  
информационных и авиационных технологий  
от «01» мая 2024 г., протокол №\_5/24

Председатель \_\_\_\_\_ / М.А. Волков  
«21» мая 2024 г.



## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Дисциплина	<b>Программная инженерия</b>
Факультет	Факультет математики, информационных и авиационных технологий
Кафедра	Кафедра информационных технологий
Курс	3 - очная форма обучения

Направление (специальность): 09.03.03 Прикладная информатика

Направленность (профиль/специализация): Информационная сфера

Форма обучения: очная

Дата введения в учебный процесс УлГУ: 01.09.2024 г.

Программа актуализирована на заседании кафедры: протокол № \_\_\_\_\_ от \_\_\_\_\_ 20\_\_ г.

Программа актуализирована на заседании кафедры: протокол № \_\_\_\_\_ от \_\_\_\_\_ 20\_\_ г.

Программа актуализирована на заседании кафедры: протокол № \_\_\_\_\_ от \_\_\_\_\_ 20\_\_ г.

Сведения о разработчиках:

ФИО	КАФЕДРА	Должность, ученая степень, звание
Санкин Николай Юрьевич	Кафедра информационных технологий	Доцент, Кандидат технических наук

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

## **1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

### **Цели освоения дисциплины:**

сформировать у студентов представление о процессах проектирования, разработки, тестирования и эксплуатации программного продукта и о взаимосвязи всех аспектов программной инженерии.

### **Задачи освоения дисциплины:**

изучить основы жизненного цикла разработки программного продукта, понимать этапы и процессы разработки программного обеспечения, включая планирование, проектирование, кодирование, тестирование, внедрение и сопровождение.

## **2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП**

Дисциплина «Программная инженерия» относится к числу дисциплин блока Б1.О, предназначенного для студентов, обучающихся по направлению: 09.03.03 Прикладная информатика.

В процессе изучения дисциплины формируются компетенции: ОПК-4, ОПК-8, ОПК-9, ПК-1, ПК-2, ПК-4.

Основные положения дисциплины используются в дальнейшем при изучении таких дисциплин как: Преддипломная практика, Эксплуатационная практика, Подготовка к сдаче и сдача государственного экзамена, Выполнение и защита выпускной квалификационной работы, Информационные сети, 1С: Предприятие для программистов и системных администраторов, Информационный менеджмент, Информатизация общества, Базы данных, Мониторинг информационных продуктов и услуг, Управление инновациями, Проектная деятельность, Открытые технологии разработки программного обеспечения, Разработка мобильных приложений, Высокопроизводительные вычисления, Программирование на языке Java, Методы разработки программного обеспечения, Системы реального времени, Прикладная статистика, Имитационное моделирование, Распределенные вычисления, Объектно-ориентированное программирование, Интеллектуальные системы и технологии, Интеллектуальный анализ данных, Программирование для Интернет, Администрирование информационных систем, Технология программирования, Сетевые технологии, Технологическая (проектно-технологическая) практика, Параллельное программирование, Функциональное программирование, Методы машинного обучения, Представление знаний, Методы и системы обработки больших данных, Web-технологии.

## **3. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОСНОВНОЙ ПРОФЕССИОНАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ**

Код и наименование реализуемой компетенции	Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций
<p>ОПК-9 Способен принимать участие в реализации профессиональных коммуникаций с заинтересованными участниками проектной деятельности и в рамках проектных групп</p>	<p><b>знать:</b>            типы заинтересованных лиц в проектах: заказчики, спонсоры, исполнители, пользователи, эксперты, регуляторы; основные принципы работы в команде: типы команд, роли в команде, техники эффективного взаимодействия, методы решения проблем в команде.</p> <p><b>уметь:</b>            общаться с разными целевыми аудиториями: учитывать специфику каждой группы заинтересованных лиц, использовать соответствующий язык и стиль коммуникации.</p> <p><b>владеть:</b>            навыками ведения переговоров: управлять процессом переговоров, определять точки соприкосновения, находить компромиссные решения.</p>
<p>ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью</p>	<p><b>знать:</b>            принципы стандартизации: основы стандартизации, типы стандартов (международные, национальные, отраслевые, корпоративные), методы разработки стандартов, процесс сертификации.</p> <p><b>уметь:</b>            анализировать существующие стандарты и нормативные документы: идентифицировать ключевые требования, выявлять противоречия и неточности, предлагать пути улучшения.</p> <p><b>владеть:</b>            навыками создания технической документации: писать инструкции, спецификации, отчеты.</p>
<p>ОПК-8 Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла</p>	<p><b>знать:</b>            жизненный цикл разработки информационных систем: этапы жизненного цикла, методы и инструменты управления каждым этапом (инициация, планирование, выполнение, мониторинг, закрытие), роль менеджера проекта на каждом этапе.</p> <p><b>уметь:</b>            планировать проект: определять цели проекта, разрабатывать план проекта, определять задачи, сроки, ресурсы, ответственных исполнителей.</p> <p><b>владеть:</b>            методами работы с программным обеспечением для управления проектами MS Project, с системой управления версиями Git.</p>
<p>ПК-1 Способен проводить обследование организаций, выявлять информационные потребности пользователей, формировать требования к информационной системе</p>	<p><b>знать:</b>            основы управления проектами: методологии управления проектами, жизненный цикл проекта, этапы планирования, разработки, внедрения и сопровождения информационных систем; программные средства и платформы инфраструктуры информационных технологий организаций.</p> <p><b>уметь:</b>            анализировать документы: идентифицировать ключевую</p>

Код и наименование реализуемой компетенции	Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций
	<p>информацию, составлять структурированные отчеты, выявлять противоречия и неточности; формулировать требования: описывать функциональные и нефункциональные требования к информационной системе, классифицировать требования, использовать языки моделирования (UML).</p> <p><b>владеть:</b> методами работы с программным обеспечением для моделирования и документирования: MS Visio, CASE-средства, UML-редакторы, системы управления версиями.</p>
ПК-2 Способен разрабатывать и адаптировать прикладное программное обеспечение	<p><b>знать:</b> принципы программирования: парадигмы программирования (объектно-ориентированное, процедурное, функциональное), алгоритмы и структуры данных, основы компиляции и интерпретации кода; языки программирования: основные синтаксические и семантические особенности одного или нескольких языков программирования.</p> <p><b>уметь:</b> проектировать архитектуру программного обеспечения: выбирать подходящий архитектурный стиль, использовать паттерны проектирования, создавать диаграммы архитектуры.</p> <p><b>владеть:</b> интегрированными средами разработки (IDE): IntelliJ IDEA, Eclipse, Visual Studio Code, Android Studio; инструментами тестирования: JUnit, PyTest, Mockito.</p>
ПК-4 Способен составлять технико-экономическое обоснование проектных решений и техническое задание на разработку информационной системы	<p><b>знать:</b> стандарты и методики документирования: ГОСТы и международные стандарты, правила написания технического задания, методы описания бизнес-процессов, создание моделей данных; типы информационных систем: функциональные и технические характеристики различных типов информационных систем.</p> <p><b>уметь:</b> разрабатывать технико-экономическое обоснование: определять цели проекта, анализировать риски, оценивать затраты и эффективность, предлагать оптимальные решения.</p> <p><b>владеть:</b> методами работы с программным обеспечением для моделирования и документирования проекта.</p>

#### 4. ОБЩАЯ ТРУДОЕМКОСТЬ ДИСЦИПЛИНЫ

4.1. Объем дисциплины в зачетных единицах (всего): 5 ЗЕТ

4.2. Объем дисциплины по видам учебной работы (в часах): 180 часов

Форма обучения: очная

Вид учебной работы	Количество часов (форма обучения <u>очная</u> )	
	Всего по плану	В т.ч. по семестрам
		5
1	2	3
Контактная работа обучающихся с преподавателем в соответствии с УП	72	72
Аудиторные занятия:	72	72
Лекции	36	36
Семинары и практические занятия	-	-
Лабораторные работы, практикумы	36	36
Самостоятельная работа	72	72
Форма текущего контроля знаний и контроля самостоятельной работы: тестирование, контр. работа, коллоквиум, реферат и др. (не менее 2 видов)	Тестирование	Тестирование
Курсовая работа	-	-
Виды промежуточной аттестации (экзамен, зачет)	Экзамен (36)	Экзамен
Всего часов по дисциплине	180	180

#### 4.3. Содержание дисциплины. Распределение часов по темам и видам учебной работы

Форма обучения: очная

Название разделов и тем	Всего	Виды учебных занятий					Форма текущего контроля знаний
		Аудиторные занятия			Занятия в интерактивной форме	Самостоятельная работа	
		Лекции	Практические занятия, семинары	Лабораторные работы, практикумы			
1	2	3	4	5	6	7	8
<b>Раздел 1. Общая характеристика областей знаний программной инженерии и их взаимосвязи</b>							
Тема 1.1. Программная инженерия, ее место в инженерной деятельности	8	2	0	2	1	4	Тестирование

Название разделов и тем	Всего	Виды учебных занятий					Форма текущего контроля знаний
		Аудиторные занятия			Занятия в интерактивной форме	Самостоятельная работа	
		Лекции	Практические занятия, семинары	Лабораторные работы, практикумы			
1	2	3	4	5	6	7	8
ти при создании компьютерных систем							
Тема 1.2. Методы объектного анализа и построения моделей предметных областей.	8	2	0	2	1	4	Тестирование
Тема 1.3. Формальные спецификации, доказательство и верификация программ	8	2	0	2	1	4	Тестирование
Тема 1.4. Интерфейсы, взаимодействие и изменение программ и данных: основы интеграции и преобразования разных языковых программ и данных	8	2	0	2	1	4	Тестирование
Тема 1.5. Инженерия приложений и предметной области	8	2	0	2	1	4	Тестирование
Тема 1.6. Методы	8	2	0	2	1	4	Тестирование

Название разделов и тем	Всего	Виды учебных занятий					Форма текущего контроля знаний
		Аудиторные занятия			Занятия в интерактивной форме	Самостоятельная работа	
		Лекции	Практические занятия, семинары	Лабораторные работы, практикумы			
1	2	3	4	5	6	7	8
управления проектом, риском и конфигурацией							
<b>Раздел 2. Управление требованиями и качеством</b>							
Тема 2.1. Методы определения требований в программной инженерии	8	2	0	2	1	4	Тестирование
Тема 2.2. Методы анализа требований . Структурный анализ	8	2	0	2	2	4	Тестирование
Тема 2.3. Внутренние и внешние характеристики качества ПО	8	2	0	2	1	4	Тестирование
Тема 2.4. Метрики качества	8	2	0	2	1	4	Тестирование
<b>Раздел 3. Методологии проектирования программных систем</b>							
Тема 3.1. Методологии проектирования программных систем	8	2	0	2	1	4	Тестирование
Тема 3.2. Проектирова	8	2	0	2	1	4	Тестирование

Название разделов и тем	Всего	Виды учебных занятий					Форма текущего контроля знаний
		Аудиторные занятия			Занятия в интерактивной форме	Самостоятельная работа	
		Лекции	Практические занятия, семинары	Лабораторные работы, практикумы			
1	2	3	4	5	6	7	8
ние программных систем с использованием UML							
Тема 3.3. Интеграция и программ и данных	8	2	0	2	2	4	Тестирование
<b>Раздел 4. Тестирование программного продукта</b>							
Тема 4.1. Виды и методы тестирования на различных стадиях разработки ПО	10	2	0	3	1	5	Тестирование
Тема 4.2. Категории тестов системного тестирования	9	2	0	2	1	5	Тестирование
Тема 4.3. Документирование тестирования	11	3	0	3	1	5	Тестирование
<b>Раздел 5. Сопровождение программных систем</b>							
Тема 5.1. Сопровождение программных систем	10	3	0	2	0	5	Тестирование
<b>Итого подлежит изучению</b>	144	36	0	36	18	72	

## 5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

## **Раздел 1. Общая характеристика областей знаний программной инженерии и их взаимосвязи**

### **Тема 1.1. Программная инженерия, ее место в инженерной деятельности при создании компьютерных систем**

Программная инженерия, ее место в инженерной деятельности при создании компьютерных систем и общее описание областей знаний профессионального ядра знаний SWEBOOK. ЖЦ стандарта ISO/IEC 12207 и связь его процессов с областями знаний SWEBOOK.

### **Тема 1.2. Методы объектного анализа и построения моделей предметных областей.**

Методы объектного анализа и построения моделей предметных областей. Объектно-ориентированные и стандартизованные методы проектирования архитектуры системы

### **Тема 1.3. Формальные спецификации, доказательство и верификация программ**

Формальные спецификации, доказательство и верификация программ: формальные методы спецификаций, методы доказательства правильности программ с помощью утверждений, пред- и постусловий и верификации.

### **Тема 1.4. Интерфейсы, взаимодействие и изменение программ и данных: основы интеграции и преобразования разных языковых программ и данных**

Методы изменения (реинжиниринг, рефакторинг) компонентов и систем; принципы взаимодействия неоднородных компонентов в современных промежуточных средах.

### **Тема 1.5. Инженерия приложений и предметной области**

Инженерия приложений и предметной области: основы инженерии приложений и предметных областей (доменов), тенденции и направления их развития в плане повторного использования компонентов.

### **Тема 1.6. Методы управления проектом, риском и конфигурацией**

Методы управления проектом, риском и конфигурацией: анализ и описание инженерии программирования, принципов и методов планирования и управления программным проектом, рисками и формированием версий программных систем.

## **Раздел 2. Управление требованиями и качеством**

### **Тема 2.1. Методы определения требований в программной инженерии**

Методы определения требований в программной инженерии: сбор, накопление, спецификации и классификация требований.

### **Тема 2.2. Методы анализа требований. Структурный анализ**

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

Методы анализа требований. Структурный анализ: диаграммы потоков данных; описание потоков данных и процессов. Методы анализа, ориентированные на структуры данных. Метод анализа Джексона.

### **Тема 2.3. Внутренние и внешние характеристики качества ПО**

Внутренние и внешние характеристики качества ПО. Методики повышения качества ПО и оценка их эффективности. Стандарты IEEE, связанные с качеством ПО. Закон контроля качества ПО. СММ (модель зрелости процесса разработки ПО).

### **Тема 2.4. Метрики качества**

Метрики качества. Метрики объектно-ориентированных программных систем (специфика), набор метрик.

## **Раздел 3. Методологии проектирования программных систем**

### **Тема 3.1. Методологии проектирования программных систем**

Организация коллектива разработчиков в компании Microsoft. Метод MSF и его основные модели. Управление проектом и рисками в MSF. Цели и задачи проектной группы в MSF. Agile-методология как пример гибкой методологии разработки. Рекомендации Agile по управлению продуктом, командой, контрактами, рисками. Инженерные практики. Контроль и обеспечение качества. Анализ требований. Масштабирование Agile. Гибкие методологии проектирования.

### **Тема 3.2. Проектирование программных систем с использованием UML**

Проектирование программных систем с использованием UML: назначение UML, его базовые понятия. UML как средство документирования. Диаграммы классов в UML и моделирование предметной области. Ассоциации. Обобщение. Зависимости. Диаграммы последовательности и взаимодействия компонент в UML. Декомпозиция системы средствами UML. Структурный метод проектирования программных систем.

### **Тема 3.3. Интеграции программ и данных**

Краткая характеристика технологий интеграции программ и данных: COM, CORBA, .NET и др. Проблемы преобразования данных при замене одной базы данных на другую. Методы переноса данных. Цели и задачи внесения изменений в программные системы в процессе их сопровождения. Методы внесения изменений. Основные проблемы, возникающие при внесении изменений.

## **Раздел 4. Тестирование программного продукта**

### **Тема 4.1. Виды и методы тестирования на различных стадиях разработки ПО**

Тестирование, отладка, дефект, отказ, сбой. Объекты тестирования. Роль тестирования в различных процессах разработки ПО. Уровни и виды тестирования: модульное (unit testing),

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

интеграционное (integration testing), системное (system testing). Регрессионное тестирование. Тестирование белого и черного ящика. Виды дефектов, обнаруживаемые на каждом уровне. Нисходящее и восходящее тестирование.

#### **Тема 4.2. Категории тестов системного тестирования**

тестирование целостности; стрессовое тестирование; корректность использования ресурсов; оценка производительности; эффективность защиты от искажения данных и некорректных действий; проверка инсталляции и конфигурации на разных платформах; корректность документации. Проблемы регрессионного тестирования. Приемочное тестирование.

#### **Тема 4.3. Документирование тестирования**

Версии программного продукта, системы контроля версий. Версии программного продукта и их связь с количеством дефектов. Точка конвергенции (bug convergence). Количественные критерии качества тестирования. Системы документирования дефектов (bug-tracking systems). Категории классификации дефектов: серьезность, приоритет.

### **Раздел 5. Сопровождение программных систем**

#### **Тема 5.1. Сопровождение программных систем**

Основные концепции, ключевые вопросы и техники сопровождения программных систем. Организация и методы сопровождения программных систем. Задачи и процессы переноса программ и данных на другие платформы. Управление конфигурацией, идентификацией, контролем, учетом статуса, версиями программного проекта. Этапы и процедуры при управлении конфигурацией программных средств. Технологическое обеспечение при сопровождении и управлении конфигурацией программных средств.

## **6. ТЕМЫ ПРАКТИЧЕСКИХ И СЕМИНАРСКИХ ЗАНЯТИЙ**

### **7. ЛАБОРАТОРНЫЕ РАБОТЫ, ПРАКТИКУМЫ**

Разработка технического задания

Цели: изучение рекомендаций стандартов на составление технического задания и получение практических навыков в разработке и структуризации требований в пределах одной проблемной области

Содержание: В соответствии с выбранной темой разработать техническое задание по одной из приведенных форм с уточнением требований по соответствующему стандарту.

Результаты: Студенты научились разрабатывать техническое задание по приведенной форме с уточнением требований по соответствующему стандарту

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Разработка календарного плана проекта

Цели: Получить практические навыки работы с системой управления проектами.

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

**Содержание:** создать проект в выбранной системе управления проектами. Определить задачи, соответствующие этапам ЖЦПО. Установить сроки выполнения задач. Назначить ответственных за выполнение задач. Установить зависимости между задачами. Создать диаграмму Гантта, отражающая график выполнения проекта.

**Результаты:** Студенты продемонстрировали понимание основ моделирования данных, умение строить ER-диаграммы, правильно описывать связи и создавать документацию.

**Ссылка:** <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

**Построение ER-диаграммы логической модели данных**

**Цели:** Овладеть навыками построения ER-диаграммы для логической модели данных.

**Содержание:** Выбрать тему для модели данных (например, библиотека, магазин, университет). Провести анализ выбранной темы и выделить основные сущности, их атрибуты, типы связей между сущностями. Построить ER-диаграммы: Используя выбранный инструмент (например, Microsoft Visio, Lucidchart, Draw.io) построить ER-диаграмму логической модели данных для выбранной темы.

**Результаты:** Определены сущности, типы связей и ключевые атрибуты для создания модели данных. Получен практический опыт работы с инструментами для построения ER-диаграмм.

**Ссылка:** <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

**Выгрузка модели данных в СУБД и реверс инжиниринг БД**

**Цели:** Получить навыки выгрузки модели данных в СУБД.

**Содержание:** Построить модель базы данных в ERwin согласно варианту предыдущей работы. Выполнить перенос разработанной базы данных, используя существующую модель данных (ER-диаграмму), созданную в предыдущей лабораторной работе. Выбрать СУБД (например, MySQL, PostgreSQL, MS SQL Server). Используя CASE-средство (например, Oracle SQL Developer, MySQL Workbench) или специализированные инструменты (например, SQL Developer Data Modeler) выгрузить модель данных в выбранную СУБД. Сгенерируйте SQL-скрипт для создания таблиц и связей в базе данных. Реверс-инжиниринг БД: создайте пустую базу данных в выбранной СУБД и импортируйте сгенерированный SQL-скрипт для создания таблиц и связей. Используя CASE-средство, выполнить реверс-инжиниринг базы данных (получите модель данных из существующей базы). Сравните полученную модель с исходной моделью данных.

**Результаты:** Модель данных, созданная в предыдущей лабораторной работе, выгружена в выбранную СУБД. Создан SQL-скрипт, который корректно создает таблицы и связи в базе данных. Модель данных, полученная в результате реверс-инжиниринга, соответствует исходной модели данных.

**Ссылка:** <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

**Построение функциональной SADT-модели информационной системы**

**Цели:** Научиться строить функциональные SADT-модели для описания процессов в информационной системе.

**Содержание:** Выбрать тему для функциональной SADT-модели (например, процесс обработки заказов, процесс управления проектами, процесс оказания услуг). Провести анализ выбранной темы и выделить основные функции и подфункции процесса. Используя выбранный инструмент для построения SADT-моделей (например, Microsoft Visio, PowerDesigner) постройте SADT-модель для выбранного процесса. Используйте следующие элементы SADT: функции (прямоугольники), данные (овалы), стрелки (линии, соединяющие элементы), Применить правила построения SADT-моделей: каждая функция должна быть декомпозирована на подфункции, число входных и выходных данных должно быть минимальным, функции должны быть независимы и не пересекаться.

**Результаты:** Построена функциональная SADT-модель для выбранного процесса. Создан документ, описывающий процесс, SADT-модель и ее входные и выходные данные. Студенты должны

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

продемонстрировать понимание принципов построения SADT-моделей и умение применять их для описания бизнес-процессов. Проведён анализ и декомпозиция процесса на основные функции и подфункций

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Разработка модели IDEF3

Цели: Научиться строить функциональные модели IDEF3. Изучить основные элементы и правила построения моделей IDEF3.

Содержание: Выбрать тему для модели IDEF3 (например, процесс оформления заказа, процесс управления проектами, процесс обслуживания клиентов, процесс обработки заявки, процесс проведения научного исследования). Провести анализ выбранной темы и выделить основные функции и подфункции процесса. Построение модели IDEF3: используя выбранный инструмент для построения моделей IDEF3 (Microsoft Visio, iGrafx, PowerDesigner), построить модель IDEF3 для выбранного процесса.

Результаты: Студенты должны продемонстрировать понимание принципов построения моделей IDEF3 и умение применять их для описания бизнес-процессов. Анализ и декомпозиция процесса на основные функции и подфункции должны быть выполнены корректно и полно. Графическое представление модели IDEF3 должно быть выполнено в соответствии с правилами и с использованием соответствующего программного обеспечения.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Построение модели потоков данных по стандарту DFD

Цели: Научиться строить модели потоков данных для описания информационных потоков в системе. Изучить основные элементы и правила построения DFD-моделей

Содержание: Выбрать тему для модели DFD (процесс обработки заказа, процесс управления проектами, процесс обслуживания клиентов, процесс обработки заявки, процесс проведения научного исследования). Провести анализ выбранной темы и выделить основные процессы, данные, хранилища данных, внешние сущности и информационные потоки. Используя выбранный инструмент для построения DFD-моделей (например, Microsoft Visio, iGrafx, PowerDesigner), построить DFD-модель для выбранного процесса.

Результаты: Студент должен продемонстрировать понимание принципов построения DFD-моделей и умение применять их для описания информационных потоков в системе.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Проектирование архитектуры программного комплекса

Цели: изучение базовых принципов и шаблонов построения архитектуры приложений, выбор стратегии и шаблона проектирования, применяемых при проектировании слоев, компонентов и сервисов.

Содержание: Выбрать тему для приложения, которое нужно спроектировать (интернет-магазин, система управления проектами, система обработки заявок). Изучить принципы и шаблоны архитектуры: прочитать документацию по основным принципам архитектуры приложений (SOLID, DRY, KISS, YAGNI); выбрать шаблон проектирования (MVC, MVVM, Microservices, Layered Architecture). Определить ключевые функциональные требования к приложению; выберите стратегию проектирования слоев (например, трехслойная архитектура, микросервисы, Event-Driven Architecture); выберите шаблоны проектирования, которые будут применяться для реализации слоев, компонентов и сервисов. Визуальное проектирование архитектуры: используя Microsoft Visio, создайте диаграмму архитектуры приложения; на диаграмме отобразите слои, компоненты, сервисы и их взаимодействия.

Результаты: Студенты должны продемонстрировать понимание основных принципов и шаблонов

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

проектирования архитектуры приложений, умение выбирать подходящие стратегии и шаблоны проектирования, визуально проектировать архитектуру приложения в Microsoft Visio.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Применение UML для построения диаграммы классов проектируемой системы

Цели: Изучить основы языка UML и его применение для моделирования классов в объектно-ориентированных системах.

Содержание: Выбрать тему для проектируемой системы (система управления заказами, система онлайн-обучения, система управления проектами). Проанализировать выбранную систему и выделить основные классы, их атрибуты и методы. Построение диаграммы классов: используя выбранный инструмент для построения UML-диаграмм (например, Microsoft Visio, StarUML, Lucidchart, Draw.io), построить диаграмму классов для проектируемой системы. Отобразить на диаграмме классы, их атрибуты, методы и взаимосвязи между классами (ассоциации, зависимости, наследование). Использовать нотацию UML для описания классов, их атрибутов, методов и взаимосвязей.

Результаты: Студенты должны продемонстрировать понимание основ UML и умение строить диаграммы классов, отражающие структуру системы.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Применение диаграммы вариантов использования для описания требований к разрабатываемой системе

Цели: Ознакомиться с концепцией диаграмм вариантов использования (Use Case Diagrams) в UML. Научиться строить диаграммы вариантов использования для описания функциональных требований к разрабатываемой системе.

Содержание: Выбрать тему для разрабатываемой системы (интернет-магазин, система управления проектами, система обработки заявок, платформа для онлайн-обучения). Провести анализ выбранной темы и выделить основные действующие лица (актеры) системы и варианты использования (use cases), которые они выполняют. Используя выбранный инструмент для построения UML-диаграмм (например, Microsoft Visio, StarUML, Draw.io), построить диаграмму вариантов использования для разрабатываемой системы. Отобразите на диаграмме действующие лица, варианты использования и их связи (ассоциации). Для каждого варианта использования создайте отдельное описание, которое включает: название варианта использования; краткое описание варианта использования; актёров, которые участвуют в варианте использования; предварительные условия для выполнения варианта использования; последовательность шагов (действий), которые выполняются в варианте использования; возможные альтернативные пути выполнения; результат выполнения варианта использования.

Результаты: Студенты должны продемонстрировать понимание принципов построения диаграмм вариантов использования в UML и умение применять их для описания требований к системе.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Разработка диаграммы последовательности

Цели: Изучить концепцию диаграммы последовательности в UML. Научиться строить диаграммы последовательности для описания взаимодействия объектов в проектируемой системе.

Содержание: Получить у преподавателя индивидуальное задание с описанием системы и выбранного варианта использования. Проанализировать полученный вариант использования, выделить участвующие объекты (классы) и действия, которые они выполняют. Используя выбранный инструмент для построения UML-диаграмм (например, Microsoft Visio, StarUML, Draw.io), построить диаграмму последовательности для выбранного варианта использования. Отобразить на диаграмме: участвующие объекты (классы); сообщения (взаимодействия) между

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

объектами (указать направление, тип и аргументы); последовательность выполнения действий; жизненный цикл каждого объекта. Использовать нотацию UML для описания объектов, сообщений и жизненных циклов.

Результаты: Студенты должны продемонстрировать понимание принципов построения диаграмм последовательности в UML и умение применять их для описания динамики взаимодействия объектов в системе.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Разработка диаграммы сотрудничества

Цели: Ознакомиться с инструментами, позволяющими строить диаграммы сотрудничества.

Содержание: Проанализировать выбранную тему и выделить основные объекты (классы) системы и их взаимосвязи, которые участвуют в определенном варианте использования. Используя выбранный инструмент для построения UML-диаграмм (например, Microsoft Visio, StarUML, Draw.io), построить диаграмму сотрудничества для выбранного варианта использования. Отобразите на диаграмме: участвующие объекты (классы) и их взаимосвязи (ассоциации, зависимости, наследование); сообщения (взаимодействия) между объектами (указать направление, тип и аргументы); номер последовательности сообщений. Использовать нотацию UML для описания объектов, сообщений и их последовательности.

Результаты: Студенты должны продемонстрировать понимание принципов построения диаграмм сотрудничества в UML и умение применять их для описания структуры взаимодействия объектов в системе.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Применение диаграммы состояний

Цели: Научиться применять диаграмму состояний для описания поведения разрабатываемой системы

Содержание: Выбрать реальный процесс, который надо смоделировать с помощью диаграммы состояний. Определить объекты, участвующие в выбранном процессе. Для одного из выбранных объектов разработать диаграмму состояний с использованием UML-нотации. Описать каждое состояние, переход, событие и действие на диаграмме. Проанализировать полученную диаграмму состояний.

Результаты: Студент должен продемонстрировать понимание принципов построения диаграмм состояний в UML и умение применять их для описания структуры взаимодействия объектов в системе.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Диаграмма деятельности

Цели: Применить диаграмму деятельности для описания поведения разрабатываемой системы.

Содержание: Студенты должны выбрать реальный процесс, который можно смоделировать с помощью диаграммы деятельности. Определить действия, участвующие в выбранном процессе. Разработать диаграмму деятельности для выбранного процесса с использованием UML-нотации. Описать каждое действие, поток управления, ветвление, слияние, узел принятия решения, узел синхронизации на диаграмме.

Результаты: Студенты должны продемонстрировать понимание принципов построения диаграмм деятельности в UML и умение применять их для описания структуры взаимодействия объектов в системе.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Компонентная диаграмма

Цели: Получить навыки построения диаграммы компонентов для описания поведения

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

разрабатываемой системы.

Содержание: Студенты должны выбрать реальное программное приложение, которое можно смоделировать с помощью диаграммы компонентов. Например: систему управления базой данных, веб-приложение, десктопное приложение, мобильные приложение. Определить компоненты приложения и их функции. Определить интерфейсы, через которые компоненты взаимодействуют друг с другом. Разработать диаграмму компонентов для выбранного приложения с использованием UML-нотации.

Результаты: Получена диаграмма компонентов UML, моделирующая архитектуру выбранного приложения

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

Применение диаграммы размещения

Цели: Применить диаграмму размещения для описания поведения разрабатываемой системы. Ознакомиться с инструментами, позволяющими строить диаграммы размещения.

Содержание: Разработать диаграмму размещений в соответствии с индивидуальным вариантом. Индивидуальный вариант предоставляется преподавателем. Выбрать реальное программное приложение или систему для моделирования. Определить узлы системы, компоненты, размещенные на узлах, и их связи. Разработать диаграмму размещения для выбранного приложения. Проанализировать полученную диаграмму и сделать выводы о физической архитектуре приложения. Результаты: Студент должен продемонстрировать понимание принципов построения диаграмм размещений.

Ссылка: <http://lib.ulsu.ru/MegaPro/Download/MObject/10234>

## 8. ТЕМАТИКА КУРСОВЫХ, КОНТРОЛЬНЫХ РАБОТ, РЕФЕРАТОВ

Данный вид работы не предусмотрен УП.

## 9. ПЕРЕЧЕНЬ ВОПРОСОВ К ЭКЗАМЕНУ

1. Раскройте роль программной инженерии в обеспечении качества и надежности компьютерных систем. Объясните, как методы верификации и валидации ПО способствуют минимизации рисков и повышению эффективности разработки сложных систем. Приведите примеры инструментов и техник, используемых в программной инженерии для обеспечения качества.
2. Опишите жизненный цикл разработки программного обеспечения (ПО) по каскадной модели и укажите ее преимущества и недостатки по сравнению с итеративной моделью разработки. Приведите примеры ситуаций, когда каскадная модель может быть предпочтительнее, и когда – итеративная.
3. Сравните и сопоставьте два популярных метода объектного анализа: UML и ORM. Опишите основные элементы, используемые в каждом методе, для моделирования предметной области. Приведите пример, демонстрирующий, как можно представить одну и ту же предметную область с помощью обоих методов, и обсудите преимущества и недостатки каждого подхода.
4. Представьте, что вам необходимо разработать программное обеспечение для управления библиотекой. Опишите процесс объектного анализа этой предметной области, используя UML диаграммы. Какие классы вы бы выделили? Какие атрибуты и методы они будут иметь? Покажите, как бы вы представили отношения между классами с помощью диаграммы классов и диаграммы последовательности для одного конкретного сценария (например, выдача книги читателю).

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

5. Формальные спецификации программного обеспечения. Почему их использование важно для повышения надежности и качества программных систем. Сравните преимущества и недостатки использования формальных методов спецификации по сравнению с неформальными методами (например, естественный язык). Приведите пример формальной спецификации простого алгоритма, используя какой-либо формальный язык (например, логику предикатов или Z-нотацию – по выбору студента).
6. Методы верификации и валидации программного обеспечения, используемые при работе с формальными спецификациями. В чем состоит разница между верификацией и валидацией? Как формальные методы позволяют проводить доказательство корректности программ? Рассмотрите, какие сложности могут возникнуть при использовании формальных методов верификации для больших и сложных программных систем.
7. Привести различные подходы к интеграции программных систем, разработанных на разных языках программирования. Сравните преимущества и недостатки использования таких методов, как обмен сообщениями (message passing), использование общих библиотек, преобразование данных (data transformation) и использование промежуточного программного обеспечения (middleware). Приведите примеры технологий, используемых для решения задач интеграции в гетерогенных средах.
8. Рассмотрите задачу интеграции данных из двух различных источников: базы данных, разработанной на основе реляционной модели (например, MySQL), и файла в формате XML. Опишите, какие этапы необходимо выполнить для преобразования и интеграции данных из этих источников. Какие инструменты и технологии вы бы использовали для решения этой задачи? Какие проблемы могут возникнуть при преобразовании данных, и как их можно решить?
9. Основные этапы процесса разработки программного обеспечения, ориентированного на предметную область (Domain-Driven Design - DDD). Объясните, как выявление и моделирование предметной области влияет на архитектуру и дизайн программного обеспечения. Какие инструменты и техники используются для работы с предметной областью в рамках DDD? Приведите пример, иллюстрирующий применение принципов DDD при разработке программного обеспечения для конкретной предметной области (например, банковская система, система управления складом).
10. Сравните и сопоставьте два подхода к разработке программного обеспечения: подход, ориентированный на данные (data-centric approach), и подход, ориентированный на предметную область (domain-centric approach). В каких ситуациях каждый из этих подходов предпочтительнее? Какие преимущества и недостатки имеет каждый подход? Приведите примеры программных систем, разработанных с использованием каждого из подходов.
11. Сравните и сопоставьте две методологии управления проектами: Agile и Waterfall. Опишите основные принципы, процессы и артефакты, используемые в каждой методологии. В каких ситуациях каждая из этих методологий наиболее эффективна? Какие риски связаны с применением каждой методологии, и как их можно минимизировать?
12. Опишите процесс управления рисками в проекте разработки программного обеспечения. Какие этапы включает в себя этот процесс? Какие методы используются для идентификации, анализа и оценки рисков? Как разрабатываются планы реагирования на риски? Приведите примеры конкретных рисков, характерных для проектов разработки ПО, и предложите способы их минимизации или управления.
13. Сравните и сопоставьте два метода сбора требований к программному обеспечению: интервьюирование и анализ документов. Опишите преимущества и недостатки каждого метода. В каких ситуациях каждый из этих методов наиболее эффективен? Какие техники можно использовать для повышения эффективности сбора требований с помощью этих методов?

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

Приведите примеры типов документов, которые могут быть полезны при анализе требований.

14. Опишите процесс создания спецификации требований к программному обеспечению. Какие этапы включает в себя этот процесс? Какие техники и инструменты можно использовать для документирования требований? Как обеспечить, чтобы спецификация требований была полной, непротиворечивой и понятной для всех заинтересованных сторон? Приведите пример, демонстрирующий использование разных техник документирования требований (например, user stories, use cases, диаграммы UML).

15. Опишите основные этапы структурного анализа требований к программному обеспечению. Какие диаграммы используются в структурном анализе (например, диаграммы потоков данных (DFD), диаграммы сущность-связь (ERD))? Объясните назначение каждой диаграммы и покажите, как они взаимосвязаны. Приведите пример, демонстрирующий использование диаграмм DFD и ERD для моделирования простой системы (например, система учета книг в библиотеке).

16. Сравните структурный анализ с объектно-ориентированным анализом. Какие преимущества и недостатки имеет каждый подход? В каких ситуациях предпочтительнее использовать структурный анализ, а в каких – объектно-ориентированный анализ? Обсудите, как можно интегрировать элементы структурного и объектно-ориентированного анализа при разработке сложных программных систем.

17. Опишите основные внутренние и внешние характеристики качества программного обеспечения (ПО) согласно модели ISO/IEC 9126. Приведите примеры конкретных метрик для измерения каждой из этих характеристик. Объясните, как внутренние характеристики качества влияют на внешние характеристики и как это учитывается при разработке и тестировании ПО. Проиллюстрируйте свой ответ конкретными примерами.

18. Рассмотрите разработку мобильного приложения для онлайн-покупок. Определите, какие внутренние и внешние характеристики качества ПО являются наиболее важными для этого приложения. Обоснуйте свой выбор, указав, как эти характеристики влияют на удовлетворенность пользователей и успех приложения на рынке. Предложите конкретные методы и техники, которые можно использовать для обеспечения высокого качества ПО в соответствии с определенными характеристиками.

19. Опишите различные категории метрик качества программного обеспечения (например, функциональные, нефункциональные, внутренние, внешние). Приведите примеры конкретных метрик для каждой категории, объяснив, что они измеряют и как их можно использовать для оценки качества ПО. Объясните, какие сложности могут возникнуть при измерении качества ПО с помощью метрик, и как можно минимизировать эти сложности.

20. Рассмотрим разработку веб-приложения для онлайн-бронирования билетов. Определите, какие метрики качества наиболее важны для этого приложения. Обоснуйте свой выбор, объяснив, как эти метрики будут использоваться для оценки качества приложения на разных этапах разработки и эксплуатации (например, тестирование, мониторинг производительности). Какие инструменты или методы можно использовать для сбора данных для этих метрик?

21. Сравните и сопоставьте две методологии разработки программного обеспечения: экстремальное программирование (XP) и Scrum. Опишите основные принципы, практики и артефакты, используемые в каждой методологии. В каких ситуациях каждая из этих методологий наиболее эффективна? Какие преимущества и недостатки имеет каждая методология? Как эти методологии относятся к Agile-манифесту?

22. Представьте, что вы являетесь руководителем проекта по разработке мобильного приложения для заказа еды на дом. Выберите подходящую методологию разработки (обоснуйте свой выбор) и опишите, как вы будете организовывать процесс разработки, используя выбранную методологию.

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

Какие этапы разработки вы бы выделили? Какие инструменты и техники вы бы использовали для управления проектом и командой? Как вы будете контролировать качество и сроки выполнения проекта?

23. Разработайте диаграмму классов UML для системы онлайн-магазина, включающую следующие сущности: клиент, товар, заказ, корзина, способ оплаты. Укажите атрибуты и методы для каждого класса, а также отношения между классами (ассоциации, агрегация, композиция, обобщение). Объясните свой выбор атрибутов и методов, а также типов отношений между классами.

24. Опишите процесс моделирования взаимодействия между объектами в системе онлайн-бронирования отелей с использованием диаграммы последовательности UML. Рассмотрите сценарий бронирования номера пользователем. Какие объекты участвуют во взаимодействии? В какой последовательности они обмениваются сообщениями? Какие методы вызываются у каждого объекта? Нарисуйте диаграмму последовательности, иллюстрирующую данный сценарий.

25. Опишите три различных подхода к интеграции программных систем, указав их преимущества и недостатки. Приведите конкретный пример для каждого подхода, объясняя, в каких ситуациях он является оптимальным выбором.

26. Рассмотрите проблему интеграции двух существующих систем с различными форматами данных. Опишите этапы процесса интеграции, уделив особое внимание способам преобразования данных. Какие инструменты и технологии можно использовать для решения этой задачи? Какие риски существуют при интеграции данных и как их можно снизить?

27. Опишите жизненный цикл разработки ПО, используя каскадную (водопадную) модель. Для каждого этапа жизненного цикла укажите соответствующие виды тестирования и обоснуйте их необходимость. Какие виды тестирования являются наиболее важными и почему?

28. Сравните и противопоставьте модульное и интеграционное тестирование. Какие цели преследуются при проведении каждого из этих видов тестирования? Какие методы и техники используются для их проведения? Какие метрики можно использовать для оценки эффективности модульного и интеграционного тестирования?

29. Опишите не менее пяти категорий системного тестирования программного обеспечения. Для каждой категории приведите конкретный пример тестового случая и объясните, какие аспекты системы он проверяет. Укажите, какие методы и инструменты можно использовать для проведения тестов в рамках каждой категории.

30. Рассмотрите взаимосвязь между различными категориями системного тестирования. Объясните, как результаты одного вида тестирования могут влиять на планирование и проведение других видов тестирования. Приведите примеры ситуаций, когда необходимо провести дополнительные виды тестирования в результате выявленных проблем на предыдущих этапах.

31. Опишите необходимую документацию для полного цикла тестирования программного обеспечения. Для каждого типа документа укажите его цель, содержание и аудиторию. Как правильно структурировать и хранить тестовую документацию для обеспечения её доступности и понятности для различных участников процесса разработки?

32. Рассмотрите ситуацию, когда в ходе тестирования были обнаружены серьезные дефекты. Опишите, какую документацию необходимо подготовить и как её следует представить заказчику и команде разработчиков. Какие критерии используются для классификации обнаруженных дефектов по серьезности? Как документировать процесс исправления дефектов и верификации исправлений?

33. Опишите различные типы сопровождения программного обеспечения, приведя примеры для каждого типа. Укажите факторы, влияющие на выбор подхода к сопровождению (например, тип программного обеспечения, бюджет, сроки). Какие метрики можно использовать для оценки эффективности процесса сопровождения?

34. Рассмотрите проблему сопровождения устаревшей системы. Какие риски связаны с поддержкой такой системы? Какие стратегии можно применить для улучшения ситуации (например, рефакторинг, миграция на новую платформу, замена системы)? Опишите факторы, которые необходимо учитывать при выборе стратегии сопровождения устаревшего ПО.

## 10. САМОСТОЯТЕЛЬНАЯ РАБОТА ОБУЧАЮЩИХСЯ

*Содержание, требования, условия и порядок организации самостоятельной работы обучающихся с учетом формы обучения определяются в соответствии с «Положением об организации самостоятельной работы обучающихся», утвержденным Ученым советом УлГУ (протокол №8/268 от 26.03.2019г.).*

*По каждой форме обучения: очная/заочная/очно-заочная заполняется отдельная таблица*

Форма обучения: очная

Название разделов и тем	Вид самостоятельной работы (проработка учебного материала, решение задач, реферат, доклад, контрольная работа, подготовка к сдаче зачета, экзамена и др).	Объем в часах	Форма контроля (проверка решения задач, реферата и др.)
<b>Раздел 1. Общая характеристика областей знаний программной инженерии и их взаимосвязи</b>			
Тема 1.1. Программная инженерия, ее место в инженерной деятельности при создании компьютерных систем	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 1.2. Методы объектного анализа и построения моделей предметных областей.	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 1.3. Формальные спецификации, доказательство и верификация программ	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 1.4. Интерфейсы, взаимодействие и изменение программ и данных: основы интеграции и преобразования разных языковых программ и данных	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

<b>Название разделов и тем</b>	<b>Вид самостоятельной работы (проработка учебного материала, решение задач, реферат, доклад, контрольная работа, подготовка к сдаче зачета, экзамена и др).</b>	<b>Объем в часах</b>	<b>Форма контроля (проверка решения задач, реферата и др.)</b>
Тема 1.5. Инженерия приложений и предметной области	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 1.6. Методы управления проектом, риском и конфигурацией	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
<b>Раздел 2. Управление требованиями и качеством</b>			
Тема 2.1. Методы определения требований в программной инженерии	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 2.2. Методы анализа требований. Структурный анализ	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 2.3. Внутренние и внешние характеристики качества ПО	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 2.4. Метрики качества	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
<b>Раздел 3. Методологии проектирования программных систем</b>			
Тема 3.1. Методологии проектирования программных систем	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 3.2. Проектирование программных систем с использованием UML	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	4	Вопросы к экзамену, Тестирование
Тема 3.3. Интеграции программ и данных	Проработка учебного материала с использованием ресурсов учебно-	4	Вопросы к экзамену, Тестирование

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

Название разделов и тем	Вид самостоятельной работы (проработка учебного материала, решение задач, реферат, доклад, контрольная работа, подготовка к сдаче зачета, экзамена и др).	Объем в часах	Форма контроля (проверка решения задач, реферата и др.)
	методического и информационного обеспечения дисциплины.		
<b>Раздел 4. Тестирование программного продукта</b>			
Тема 4.1. Виды и методы тестирования на различных стадиях разработки ПО	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	5	Вопросы к экзамену, Тестирование
Тема 4.2. Категории тестов системного тестирования	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	5	Вопросы к экзамену, Тестирование
Тема 4.3. Документирование тестирования	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	5	Вопросы к экзамену, Тестирование
<b>Раздел 5. Сопровождение программных систем</b>			
Тема 5.1. Сопровождение программных систем	Проработка учебного материала с использованием ресурсов учебно-методического и информационного обеспечения дисциплины.	5	Вопросы к экзамену, Тестирование

## 11. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### а) Список рекомендуемой литературы основная

1. Мейер, Б. Объектно-ориентированное программирование и программная инженерия / Б. Мейер ; Б. Мейер. - Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. - 285 с. - Книга находится в премиум-версии ЭБС IPR BOOKS. - Текст. - Лицензия до 30.11.2021. - электронный. - Электрон. дан. (1 файл). - URL: <http://www.iprbookshop.ru/79706.html>. - Режим доступа: ЭБС IPR BOOKS; для авторизир. пользователей. - ISBN 978-5-4486-0513-0. / .— ISBN 0\_145875
2. Черткова Елена Александровна. Программная инженерия. Визуальное моделирование программных систем : Учебник для вузов / Е.А. Черткова. - 2-е изд. ; испр. и доп. - Москва : Юрайт, 2021. - 147 с. - (Высшее образование). - <https://urait.ru/bcode/471564>. -

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

<https://urait.ru/book/cover/58E3F4E5-6D84-4208-B4DB-5F28ABD6758E>. - Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользователей. - Электрон. дан. - ISBN 978-5-534-09172-4 : 349.00. / .— ISBN 0\_282066

3. Лаврищева Екатерина Михайловна. Программная инженерия. Парадигмы, технологии и CASE-средства : Учебник для вузов / Е.М. Лаврищева. - 2-е изд. - Москва : Юрайт, 2021. - 280 с. - (Высшее образование). - <https://urait.ru/bcode/470942>. - <https://urait.ru/book/cover/5BFAEF55-6F98-4F81-853D-CBF4D10724EE>. - Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользователей. - Электрон. дан. - ISBN 978-5-534-01056-5 : 839.00. / .— ISBN 0\_282478

#### **дополнительная**

1. Программная инженерия: Часть II : учебное пособие. Направление подготовки 09.03.03 Прикладная информатика. Профиль подготовки «Прикладная информатика в экономике». Бакалавриат. Ч. 2 : Программная инженерия: Часть II. - Ставрополь : СКФУ, 2017. - 100 с. - Библиогр.: доступна в карточке книги, на сайте ЭБС Лань. - Книга из коллекции СКФУ - Информатика. - <https://e.lanbook.com/book/155149>. - <https://e.lanbook.com/img/cover/book/155149.jpg>. - Режим доступа: ЭБС "Лань"; для авторизир. пользователей. / .— ISBN 0\_382645

2. Программная инженерия. Часть 1 : учебное пособие. Ч. 1 : Программная инженерия. Часть 1. - Ставрополь : СКФУ, 2017. - 137 с. - Библиогр.: доступна в карточке книги, на сайте ЭБС Лань. - Книга из коллекции СКФУ - Информатика. - <https://e.lanbook.com/book/155150>. - <https://e.lanbook.com/img/cover/book/155150.jpg>. - Режим доступа: ЭБС "Лань"; для авторизир. пользователей. / .— ISBN 0\_382646

3. Минакова О. В. Программная инженерия. Основные принципы, методы и инструменты : учебник для вузов / О. В. Минакова. - Санкт-Петербург : Лань, 2024. - 212 с. - Книга из коллекции Лань - Информатика. - URL: <https://e.lanbook.com/book/414989>. - <https://e.lanbook.com/img/cover/book/414989.jpg>. - Режим доступа: ЭБС "Лань"; для авторизир. пользователей. - ISBN 978-5-507-49278-7. / .— ISBN 0\_547215

#### **учебно-методическая**

1. Санкин Н. Ю. Программная инженерия : методические указания для самостоятельной работы студентов по направлениям подготовки 09.03.03 «Прикладная информатика» / Н. Ю. Санкин ; УлГУ, ФМИиАТ. - 2024. - Неопубликованный ресурс. - URL: <http://lib.ulsu.ru/MegaPro/Download/MObject/16380>. - Режим доступа: ЭБС УлГУ. - Текст : электронный. / .— ISBN 0\_557627.

#### **б) Программное обеспечение**

- Операционная система "Альт образование"
- Офисный пакет "Мой офис"

#### **в) Профессиональные базы данных, информационно-справочные системы**

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

## 1. Электронно-библиотечные системы:

1.1. Цифровой образовательный ресурс IPRsmart : электронно-библиотечная система : сайт / ООО Компания «Ай Пи Ар Медиа». - Саратов, [2024]. – URL: <http://www.iprbookshop.ru>. – Режим доступа: для зарегистрир. пользователей. - Текст : электронный.

1.2. Образовательная платформа ЮРАЙТ : образовательный ресурс, электронная библиотека : сайт / ООО Электронное издательство ЮРАЙТ. – Москва, [2024]. - URL: <https://urait.ru>. – Режим доступа: для зарегистрир. пользователей. - Текст : электронный.

1.3. База данных «Электронная библиотека технического ВУЗа (ЭБС «Консультант студента») : электронно-библиотечная система : сайт / ООО Политехресурс. – Москва, [2024]. – URL: <https://www.studentlibrary.ru/cgi-bin/mb4x>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

1.4. Консультант врача. Электронная медицинская библиотека : база данных : сайт / ООО Высшая школа организации и управления здравоохранением-Комплексный медицинский консалтинг. – Москва, [2024]. – URL: <https://www.rosmedlib.ru>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

1.5. Большая медицинская библиотека : электронно-библиотечная система : сайт / ООО Букап. – Томск, [2024]. – URL: <https://www.books-up.ru/ru/library/> . – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

1.6. ЭБС Лань : электронно-библиотечная система : сайт / ООО ЭБС Лань. – Санкт-Петербург, [2024]. – URL: <https://e.lanbook.com>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

1.7. ЭБС **Znanium.com** : электронно-библиотечная система : сайт / ООО Знаниум. - Москва, [2024]. - URL: <http://znanium.com> . – Режим доступа : для зарегистрир. пользователей. - Текст : электронный.

**2. КонсультантПлюс** [Электронный ресурс]: справочная правовая система. /ООО «Консультант Плюс» - Электрон. дан. - Москва : КонсультантПлюс, [2024].

**3. eLIBRARY.RU:** научная электронная библиотека : сайт / ООО «Научная Электронная Библиотека». – Москва, [2024]. – URL: <http://elibrary.ru>. – Режим доступа : для авториз. пользователей. – Текст : электронный

**4. Федеральная государственная информационная система «Национальная электронная библиотека» :** электронная библиотека : сайт / ФГБУ РГБ. – Москва, [2024]. – URL: <https://нэб.рф>. – Режим доступа : для пользователей научной библиотеки. – Текст : электронный.

**5. Российское образование :** федеральный портал / учредитель ФГАУ «ФИЦТО». – URL: <http://www.edu.ru>. – Текст : электронный.

Министерство науки и высшего образования РФ Ульяновский государственный университет Ф – Рабочая программа дисциплины	Форма	
--	-------	--

**6. Электронная библиотечная система УлГУ :** модуль «Электронная библиотека» АБИС Мега-ПРО / ООО «Дата Экспресс». – URL: <http://lib.ulsu.ru/MegaPro/Web>. – Режим доступа : для пользователей научной библиотеки. – Текст : электронный.

## 12. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ:

Аудитории для проведения лекций, семинарских занятий, для выполнения лабораторных работ и практикумов, для проведения текущего контроля и промежуточной аттестации, курсового проектирования, групповых и индивидуальных консультаций (*выбрать необходимое*)

Аудитории укомплектованы специализированной мебелью, учебной доской. Аудитории для проведения лекций оборудованы мультимедийным оборудованием для представления информации большой аудитории. Помещения для самостоятельной работы оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к электронной информационно-образовательной среде, электронно-библиотечной системе. Перечень оборудования, используемого в учебном процессе:

- Мультимедийное оборудование: компьютер/ноутбук, экран, проектор/телевизор
- Компьютерная техника

## 13. СПЕЦИАЛЬНЫЕ УСЛОВИЯ ДЛЯ ОБУЧАЮЩИХСЯ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

В случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) могут предлагаться одни из следующих вариантов восприятия информации с учетом их индивидуальных психофизических особенностей:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации;

- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации;

- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.

В случае необходимости использования в учебном процессе частично/исключительно дистанционных образовательных технологий, организация работы ППС с обучающимися с ОВЗ и инвалидами предусматривается в электронной информационно-образовательной среде с учетом их индивидуальных психофизических особенностей.

Разработчик	Доцент Кандидат технических наук	Санкин Николай Юрьевич
	Должность, ученая степень, звание	ФИО