

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
имени И.Т. ТРУБИЛИНА»

Факультет прикладной информатики
Системного анализа и обработки информации



УТВЕРЖДЕНО
Декан
Замотайлова Д.А.
протокол от 25.04.2025 № 7

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)
«ПРОГРАММНАЯ ИНЖЕНЕРИЯ»**

Уровень высшего образования: бакалавриат

Направление подготовки: 09.03.03 Прикладная информатика

Направленность (профиль) подготовки: Менеджмент ИТ-проектов, управление жизненным циклом информационных систем

Квалификация (степень) выпускника: бакалавр

Формы обучения: очная, заочная

Год набора (приема на обучение): 2025

Срок получения образования: Очная форма обучения – 4 года
Заочная форма обучения – 4 года 8 месяца(-ев)

Объем: в зачетных единицах: 4 з.е.
в академических часах: 144 ак.ч.

Разработчики:

Доцент, кафедра системного анализа и обработки информации Ефанова Н.В.

Рабочая программа дисциплины (модуля) составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика, утвержденного приказом Минобрнауки от 19.09.2017 № 922, с учетом трудовых функций профессиональных стандартов: "Специалист по информационным системам", утвержден приказом Минтруда России от 13.07.2023 № 586н; "Руководитель проектов в области информационных технологий", утвержден приказом Минтруда России от 27.04.2023 № 369н; "Руководитель проектов в области информационных технологий", утвержден приказом Минтруда России от 18.11.2014 № 893н; "Специалист по информационным системам", утвержден приказом Минтруда России от 18.11.2014 № 896н.

Согласование и утверждение

№	Подразделение или коллегиальный орган	Ответственное лицо	ФИО	Виза	Дата, протокол (при наличии)
---	--	-----------------------	-----	------	---------------------------------

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины - формирование у обучающихся теоретических и практических навыков по изучению и использованию современных технологий профессиональной разработки программного обеспечения

Задачи изучения дисциплины:

- научиться использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности;
- участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью ;
- разрабатывать и адаптировать алгоритмы и прикладное программное обеспечение, пригодные для практического применения ;
- проводить тестирование компонентов программного обеспечения ИС;
- получить знания о принципах управления проектами создания информационных систем на стадиях жизненного цикла .

2. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Компетенции, индикаторы и результаты обучения

ОПК-2 Способен использовать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности

ОПК-2.1 Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

Знать:

ОПК-2.1/Зн1 Современные информационные технологии для решения задач профессиональной деятельности

ОПК-2.1/Зн2 Современные программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

ОПК-2.2 Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности

Уметь:

ОПК-2.2/Ум1 Выбирать современные информационные технологии для решения задач профессиональной деятельности

ОПК-2.2/Ум2 Выбирать современные программные средства, в том числе отечественного производства для решения задач профессиональной деятельности

ОПК-2.3 Владеет навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности

Владеть:

ОПК-2.3/Нв1 Навыками применения современных информационных технологий для решения задач профессиональной деятельности

ОПК-2.3/Нв2 Навыками применения современных программных средств, в том числе отечественного производства для решения задач профессиональной деятельности

ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью

ОПК-4.1 Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы

Знать:

ОПК-4.1/Зн1 Стандарты оформления технической документации

ОПК-4.2 Умеет применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы

Уметь:

ОПК-4.2/Ум1 Применять стандарты оформления технической документации

ОПК-4.3 Владеет навыками составления технической документации на различных этапах жизненного цикла информационной системы

Владеть:

ОПК-4.3/Нв1 Навыками составления технической документации

ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения

ОПК-7.1 Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий

Знать:

ОПК-7.1/Зн1 Основные языки программирования

ОПК-7.1/Зн3 Современные программные среды разработки информационных систем и технологий

ОПК-7.2 Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ

Уметь:

ОПК-7.2/Ум1 Применять языки программирования

ОПК-7.2/Ум3 Применять современные программные среды разработки информационных систем и технологий

ОПК-7.3 Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач

Владеть:

ОПК-7.3/Нв1 Навыками программирования прототипов программно-технических комплексов задач

ОПК-7.3/Нв2 Навыками отладки прототипов программно-технических комплексов задач

ОПК-7.3/Нв3 Навыками тестирования прототипов программно-технических комплексов задач

ОПК-8 Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла

ОПК-8.1 Знает основные технологии создания и внедрения информационных систем, стандарты управления жизненным циклом информационной системы

Знать:

ОПК-8.1/Зн1 Основные технологии создания информационных систем

ОПК-8.1/Зн2 Основные технологии внедрения информационных систем

ОПК-8.1/Зн3 Стандарты управления жизненным циклом информационной системы

ОПК-8.2 Умеет осуществлять организационное обеспечение выполнения работ на всех стадиях и в процессах жизненного цикла информационной системы

Уметь:

ОПК-8.2/Ум1 Осуществлять организационное обеспечение выполнения работ на всех стадиях жизненного цикла информационной системы

ОПК-8.3 Владеет навыками составления плановой и отчетной документации по управлению проектами создания информационных систем на стадиях жизненного цикла

Владеть:

ОПК-8.3/Нв1 Навыками составления плановой документации по управлению проектами создания информационных систем

ОПК-8.3/Нв2 Навыками составления отчетной документации по управлению проектами создания информационных систем

ПК-П2 Способен разрабатывать и адаптировать прикладное программное обеспечение

ПК-П2.1 Знает теоретическое и практическое содержание этапов процесса внедрения, адаптации и настройки прикладного программного обеспечения

Знать:

ПК-П2.1/Зн1 Основы управления изменениями

ПК-П2.1/Зн2 Инструменты и методы анализа требований

ПК-П2.1/Зн3 Возможности ис

ПК-П2.1/Зн4 Инструменты и методы коммуникаций

ПК-П2.1/Зн5 Каналы коммуникаций

ПК-П2.1/Зн6 Модели коммуникаций

Уметь:

ПК-П2.1/Ум1 Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий)

ПК-П2.1/Ум2 Работать в системе учета требований проекта в области ит

ПК-П2.1/Ум3 Осуществлять коммуникации в проекте в области ит

Владеть:

ПК-П2.1/Нв1 Контроль фактического внесения изменений в элементы ис

ПК-П2.1/Нв2 Изменение статуса проверенных запросов на изменение в системе учета требований проекта в области ит

ПК-П2.2 Умеет организовывать и управлять процессом внедрения, адаптации и настройки прикладного программного обеспечения, в т.ч., распределять работы, выделять ресурсы, контролировать исполнение

Знать:

ПК-П2.2/Зн1 Языки программирования и работы с базами данных

ПК-П2.2/Зн2 Инструменты и методы модульного тестирования

ПК-П2.2/Зн4 Инструменты и методы прототипирования пользовательского интерфейса
ПК-П2.2/Зн5 Возможности типовой ис
ПК-П2.2/Зн6 Предметная область автоматизации
ПК-П2.2/Зн7 Технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии
ПК-П2.2/Зн14 Основы программирования
ПК-П2.2/Зн15 Современные объектно-ориентированные языки программирования
ПК-П2.2/Зн16 Современные структурные языки программирования
ПК-П2.2/Зн17 Языки современных бизнес-приложений
ПК-П2.2/Зн23 Отраслевая нормативная техническая документация
ПК-П2.2/Зн24 Источники информации, необходимой для профессиональной деятельности
ПК-П2.2/Зн25 Лучшие практики создания (модификации) и сопровождения ис в экономике

Уметь:

ПК-П2.2/Ум1 Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ис
ПК-П2.2/Ум2 Тестировать результаты прототипирования ис в рамках выполнения работ по созданию (модификации) и сопровождению ис
ПК-П2.2/Ум3 Проводить презентации в рамках выполнения работ по созданию (модификации) и сопровождению ис
ПК-П2.2/Ум4 Проводить переговоры в рамках выполнения работ по созданию (модификации) и сопровождению ис
ПК-П2.2/Ум5 Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий) в рамках выполнения работ по созданию (модификации) и сопровождению ис

Владеть:

ПК-П2.2/Нв1 Разработка прототипа ис на базе типовой ис в соответствии с требованиями заказчика к ис
ПК-П2.2/Нв2 Тестирование прототипа ис на корректность архитектурных решений
ПК-П2.2/Нв3 Обработка результатов тестирования прототипа ис на корректность архитектурных решений
ПК-П2.2/Нв5 Согласование пользовательского интерфейса ис с заказчиком ис

ПК-П2.3 Владеет навыками внедрения, адаптации и настройки прикладного программного обеспечения, обеспечивая соответствие и контроль разработанного кода и процесса кодирования принятым в организации регламентам и стандартам

Знать:

ПК-П2.3/Зн1 Инструменты и методы верификации структуры программного кода
ПК-П2.3/Зн2 Регламенты кодирования на языках программирования
ПК-П2.3/Зн3 Возможности ис
ПК-П2.3/Зн4 Предметная область автоматизации
ПК-П2.3/Зн5 Источники информации, необходимой для профессиональной деятельности при выполнении работ и управлении работами по созданию (модификации) и сопровождению ис
ПК-П2.3/Зн6 Лучшие практики создания (модификации) и сопровождения ис в экономике
ПК-П2.3/Зн9 Методы управления содержанием проекта: документирование требований, анализ продукта, модерлируемые совещания

Уметь:

ПК-П2.3/Ум1 Распределять работы и выделять ресурсы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис
ПК-П2.3/Ум2 Контролировать исполнение поручений в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

Владеть:

ПК-П2.3/Нв1 Обеспечение соответствия разработанного кода ис и процесса создания программного кода ис принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П2.3/Нв2 Назначение и распределение ресурсов в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П2.3/Нв3 Контроль соответствия разработанного кода ис и процесса создания программного кода ис принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8 Способен проводить тестирование компонентов программного обеспечения ис

ПК-П8.1 Знает основные методы тестирования информационных систем и их компонентов

Знать:

ПК-П8.1/Зн1 Инструменты и методы модульного тестирования

ПК-П8.1/Зн2 Предметная область автоматизации

ПК-П8.1/Зн5 Устройство и функционирование современных ис

ПК-П8.1/Зн10 Современные методики тестирования разрабатываемых ис: инструменты и методы модульного тестирования

ПК-П8.1/Зн11 Источники информации, необходимой для профессиональной деятельности при выполнении работ по созданию (модификации) и сопровождению ис

ПК-П8.1/Зн12 Лучшие практики создания (модификации) и сопровождения ис в экономике

Уметь:

ПК-П8.1/Ум1 Проверять на корректность отдельные модули кода ис в рамках выполнения работ по созданию (модификации) и сопровождению ис

ПК-П8.1/Ум2 Кодировать на языках программирования в рамках выполнения работ по созданию (модификации) и сопровождению ис

ПК-П8.1/Ум3 Тестировать результаты кодирования ис в рамках выполнения работ по созданию (модификации) и сопровождению ис

Владеть:

ПК-П8.1/Нв1 Тестирование разрабатываемого модуля ис

ПК-П8.1/Нв2 Устранение обнаруженных несоответствий в модуле ис

ПК-П8.2 Умеет тестировать информационные системы и их компоненты различными способами

Знать:

ПК-П8.2/Зн1 Инструменты и методы модульного тестирования

ПК-П8.2/Зн2 Регламенты модульного тестирования

ПК-П8.2/Зн3 Возможности ис

ПК-П8.2/Зн4 Предметная область автоматизации

ПК-П8.2/Зн5 Источники информации, необходимой для профессиональной деятельности при выполнении работ и управлении работами по созданию (модификации) и сопровождению ис

ПК-П8.2/Зн6 Лучшие практики создания (модификации) и сопровождения ис в экономике

Уметь:

ПК-П8.2/Ум1 Распределять работы и выделять ресурсы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8.2/Ум2 Выдавать поручения и контролировать их исполнение в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

Владеть:

ПК-П8.2/Нв1 Обеспечение соответствия процессов модульного тестирования ис принятым в организации или проекте стандартам и технологиям в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8.2/Нв2 Назначение и распределение ресурсов в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8.2/Нв3 Контроль исполнения в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8.3 Владеет навыками тестирования информационных систем и компонентов программного обеспечения информационных систем

Знать:

ПК-П8.3/Зн1 Инструменты и методы проектирования архитектуры и дизайна ис

ПК-П8.3/Зн2 Инструменты и методы верификации архитектуры и дизайна ис

ПК-П8.3/Зн3 Возможности ис

ПК-П8.3/Зн4 Предметная область автоматизации

ПК-П8.3/Зн8 Основы программирования

ПК-П8.3/Зн9 Современные объектно-ориентированные языки программирования

ПК-П8.3/Зн10 Современные структурные языки программирования

ПК-П8.3/Зн11 Языки современных бизнес-приложений

ПК-П8.3/Зн12 Современные методики тестирования разрабатываемых ис: инструменты и методы модульного тестирования, инструменты и методы тестирования нефункциональных и функциональных характеристик ис

ПК-П8.3/Зн13 Источники информации, необходимой для профессиональной деятельности при выполнении работ и управлении работами по созданию (модификации) и сопровождению ис

ПК-П8.3/Зн14 Лучшие практики создания (модификации) и сопровождения ис в экономике

Уметь:

ПК-П8.3/Ум1 Проектировать архитектуру и дизайн ис в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8.3/Ум2 Проверять (верифицировать) архитектуру и дизайн ис в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8.3/Ум3 Работать с записями по качеству (в том числе с корректирующими действиями, предупреждающими действиями, запросами на исправление несоответствий) в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

Владеть:

ПК-П8.3/Нв1 Воспроизведение зафиксированных в системе учета дефектов и несоответствий в архитектуре и дизайне ис в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

ПК-П8.3/Нв2 Установление причин возникновения дефектов и несоответствий в архитектуре и дизайне ис в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ис

3. Место дисциплины в структуре ОП

Дисциплина (модуль) «Программная инженерия» относится к обязательной части образовательной программы и изучается в семестре(ах): Очная форма обучения - 5, Заочная форма обучения - 5.

В процессе изучения дисциплины студент готовится к решению типов задач профессиональной деятельности, предусмотренных ФГОС ВО и образовательной программой.

4. Объем дисциплины (модуля) и виды учебной работы

Очная форма обучения

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Контактная работа (часы, всего)	Внеаудиторная контактная работа (часы)	Лабораторные занятия (часы)	Лекционные занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
Пятый семестр	144	4	65	3	30	32	52	Экзамен (27)
Всего	144	4	65	3	30	32	52	27

Заочная форма обучения

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Контактная работа (часы, всего)	Внеаудиторная контактная работа (часы)	Лабораторные занятия (часы)	Лекционные занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
Пятый семестр	144	4	17	3	10	4	127	Экзамен
Всего	144	4	17	3	10	4	127	

5. Содержание дисциплины (модуля)

5.1. Разделы, темы дисциплины и виды занятий (часы промежуточной аттестации не указываются)

Очная форма обучения

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Самостоятельная работа	Планируемые результаты обучения, соотношенные с результатами освоения программы
Раздел 1. Введение в дисциплину программной инженерии	8			4	4	ОПК-2.1 ОПК-2.2 ОПК-2.3
Тема 1.1. Введение в дисциплину профессиональной разработки программного обеспечения	4			2	2	
Тема 1.2. Жизненный цикл ИТ-проекта и продукта	4			2	2	
Раздел 2. Стандартизация деятельности в области программной инженерии	12		4	2	6	ОПК-4.1 ОПК-4.2 ОПК-4.3
Тема 2.1. Стандарты и нормативная документация	12		4	2	6	
Раздел 3. Методологические основы разработки программных продуктов	4			2	2	ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 3.1. Методологии и технологии разработки программных продуктов	4			2	2	
Раздел 4. Основы управления ИТ-проектами	6		2	2	2	ОПК-8.1 ОПК-8.2 ОПК-8.3
Тема 4.1. Основные принципы и стандарты управления ИТ-проектами	6		2	2	2	
Раздел 5. Процессы профессиональной разработки программных продуктов	70		20	18	32	ПК-П2.1 ПК-П2.2 ПК-П2.3
Тема 5.1. Управление требованиями	18		4	4	10	
Тема 5.2. Проектирование и разработка программного обеспечения	44		14	10	20	
Тема 5.3. Профессиональные тренды в разработке программного обеспечения	8		2	4	2	
Раздел 6. Процессы тестирования программных продуктов	14		4	4	6	ПК-П8.1 ПК-П8.2 ПК-П8.3
Тема 6.1. Тестирование программных продуктов	14		4	4	6	

Раздел 7. Промежуточная аттестация	3	3				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-4.1 ОПК-4.2 ОПК-4.3 ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 7.1. Экзамен	3	3				ОПК-8.1 ОПК-8.2 ОПК-8.3 ПК-П2.1 ПК-П2.2 ПК-П2.3 ПК-П8.1 ПК-П8.2 ПК-П8.3
Итого	117	3	30	32	52	

Заочная форма обучения

Наименование раздела, темы	Всего	Внеаудиторная контактная работа	Лабораторные занятия	Лекционные занятия	Самостоятельная работа	Планируемые результаты обучения, соответствующие результатам освоения программы
Раздел 1. Введение в дисциплину программной инженерии	22			2	20	ОПК-2.1 ОПК-2.2 ОПК-2.3
Тема 1.1. Введение в дисциплину профессиональной разработки программного обеспечения	11			1	10	
Тема 1.2. Жизненный цикл ИТ-проекта и продукта	11			1	10	
Раздел 2. Стандартизация деятельности в области программной инженерии	13		2	1	10	ОПК-4.1 ОПК-4.2 ОПК-4.3
Тема 2.1. Стандарты и нормативная документация	13		2	1	10	
Раздел 3. Методологические основы разработки программных продуктов	11			1	10	ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 3.1. Методологии и технологии разработки программных продуктов	11			1	10	
Раздел 4. Основы управления ИТ-проектами	10				10	ОПК-8.1 ОПК-8.2

Тема 4.1. Основные принципы и стандарты управления ИТ-проектами	10				10	ОПК-8.3
Раздел 5. Процессы профессиональной разработки программных продуктов	63		6		57	ПК-П2.1 ПК-П2.2 ПК-П2.3
Тема 5.1. Управление требованиями	12		2		10	
Тема 5.2. Проектирование и разработка программного обеспечения	41		4		37	
Тема 5.3. Профессиональные тренды в разработке программного обеспечения	10				10	
Раздел 6. Процессы тестирования программных продуктов	22		2		20	ПК-П8.1 ПК-П8.2 ПК-П8.3
Тема 6.1. Тестирование программных продуктов	22		2		20	
Раздел 7. Промежуточная аттестация	3	3				ОПК-2.1 ОПК-2.2 ОПК-2.3 ОПК-4.1 ОПК-4.2 ОПК-4.3 ОПК-7.1 ОПК-7.2 ОПК-7.3
Тема 7.1. Экзамен	3	3				ОПК-8.1 ОПК-8.2 ОПК-8.3 ПК-П2.1 ПК-П2.2 ПК-П2.3 ПК-П8.1 ПК-П8.2 ПК-П8.3
Итого	144	3	10	4	127	

5.2. Содержание разделов, тем дисциплин

Раздел 1. Введение в дисциплину программной инженерии

(Заочная: Лекционные занятия - 2ч.; Самостоятельная работа - 20ч.; Очная: Лекционные занятия - 4ч.; Самостоятельная работа - 4ч.)

Тема 1.1. Введение в дисциплину профессиональной разработки программного обеспечения

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

1. Инженерное дело и программная инженерия
2. Типы программных продуктов
3. Защита программного обеспечения
4. Понятие ИТ-проекта
5. Инструментальные средства разработки

Тема 1.2. Жизненный цикл ИТ-проекта и продукта

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

1. Понятие ЖЦ.
2. Модели ЖЦ.

Раздел 2. Стандартизация деятельности в области программной инженерии

(Заочная: Лабораторные занятия - 2ч.; Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 6ч.)

Тема 2.1. Стандарты и нормативная документация

(Заочная: Лабораторные занятия - 2ч.; Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 6ч.)

1. Основные стандарты программной инженерии
2. Техническое задание
3. Спецификация требований

Раздел 3. Методологические основы разработки программных продуктов

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

Тема 3.1. Методологии и технологии разработки программных продуктов

(Заочная: Лекционные занятия - 1ч.; Самостоятельная работа - 10ч.; Очная: Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.)

1. Предиктивный и гибкий подход
2. Сравнительный анализ методологий
3. Технологии разработки ПО

Раздел 4. Основы управления ИТ-проектами

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 10ч.)

Тема 4.1. Основные принципы и стандарты управления ИТ-проектами

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 2ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 10ч.)

1. Проекты разработки ПО
2. Лучшие практики разработки ПО.
3. Гибридный подход

Раздел 5. Процессы профессиональной разработки программных продуктов

(Заочная: Лабораторные занятия - 6ч.; Самостоятельная работа - 57ч.; Очная: Лабораторные занятия - 20ч.; Лекционные занятия - 18ч.; Самостоятельная работа - 32ч.)

Тема 5.1. Управление требованиями

(Заочная: Лабораторные занятия - 2ч.; Самостоятельная работа - 10ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 10ч.)

Принятая терминология. Классификация требований. Процесс сбора и анализа требований. Методы представления требований

Тема 5.2. Проектирование и разработка программного обеспечения

(Заочная: Лабораторные занятия - 4ч.; Самостоятельная работа - 37ч.; Очная: Лабораторные занятия - 14ч.; Лекционные занятия - 10ч.; Самостоятельная работа - 20ч.)

1. Объектно-ориентированный анализ и проектирование (методология RUP UML) (6 ч)
2. Основы дизайна интерфейса пользователя. (4 ч)

Тема 5.3. Профессиональные тренды в разработке программного обеспечения

(Очная: Лабораторные занятия - 2ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 2ч.; Заочная: Самостоятельная работа - 10ч.)

1. Настольные и сетевые приложения
2. Программирование робототехники
3. Специфика «-техов» (edtech, med-, fin-, food-, agro-)
4. Машинное обучение и анализ данных

Раздел 6. Процессы тестирования программных продуктов

(Заочная: Лабораторные занятия - 2ч.; Самостоятельная работа - 20ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 6ч.)

Тема 6.1. Тестирование программных продуктов

(Заочная: Лабораторные занятия - 2ч.; Самостоятельная работа - 20ч.; Очная: Лабораторные занятия - 4ч.; Лекционные занятия - 4ч.; Самостоятельная работа - 6ч.)

1. Терминология и основные положения.
2. Виды и техники тестирования (классификация).
3. План тестирования, тестовое покрытие, тестовые процедуры, протоколы и отчеты тестирования ИС.

Раздел 7. Промежуточная аттестация

(Заочная: Внеаудиторная контактная работа - 3ч.; Очная: Внеаудиторная контактная работа - 3ч.)

Тема 7.1. Экзамен

(Заочная: Внеаудиторная контактная работа - 3ч.; Очная: Внеаудиторная контактная работа - 3ч.)

Проведение промежуточной аттестации в форме экзамена

6. Оценочные материалы текущего контроля

Раздел 1. Введение в дисциплину программной инженерии

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Установите правильный порядок выбора и разработки программных средств для решения профессиональных задач

Анализ требований к решаемой задаче

Исследование доступных отечественных и зарубежных аналогов

Сравнение характеристик и возможностей ПО

Принятие решения о внедрении или разработке

2. Установите правильный порядок этапов жизненного цикла ПО согласно ГОСТ Р

Формирование требований

Проектирование архитектуры

Реализация (кодирование)

Тестирование

Ввод в эксплуатацию

Сопровождение

3. Сопоставьте методологии разработки ПО с их характеристиками

Методологии:

- A. Waterfall
- Б. Agile
- В. Гибридная модель

Характеристики:

- 1. Жесткое разделение на этапы
- 2. Итеративная разработка
- 3. Комбинация предсказуемости и гибкости

4. Какой инструмент следует выбрать для обеспечения непрерывной интеграции в соответствии с современными практиками?

GitLab

Microsoft Excel

Adobe Photoshop

Windows Notepad

5. Какие методологии относятся к современным гибким подходам разработки ПО? (Выберите 2 варианта)

Waterfall

Scrum

RUP

Kanban

6. Какие этапы жизненного цикла ПО являются обязательными согласно ГОСТ Р? (Выберите 3 варианта)

Формирование требований

Кодирование

Маркетинговые исследования

Тестирование

Рекламная кампания

7. Важность этапа сопровождения

Почему при внедрении ИТ-решений важно учитывать этап сопровождения в жизненном цикле продукта?

Раздел 2. Стандартизация деятельности в области программной инженерии

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Сопоставьте виды технической документации с их назначением:

Виды технической документации:

- A. Техническое задание
- Б. Руководство пользователя
- В. Программа и методика испытаний
- Г. Отчет о тестировании

Назначение документов:

- 1. Описание процедуры проверки системы
- 2. Исходные требования к разработке
- 3. Результаты проверки соответствия
- 4. Инструкция по эксплуатации

2. Сопоставьте разделы технического задания с их содержанием:

Разделы:

- A. Общие положения
- Б. Требования к системе
- В. Этапы разработки
- Г. Порядок приемки

Содержание разделов:

- 1. Цели и назначение

2. Функциональные характеристики
3. Сроки и стадии проекта
4. Критерии проверки

3. Установите последовательность разработки руководства пользователя:

Анализ функциональности системы
Определение целевой аудитории
Разработка структуры документа
Написание инструкций
Тестирование на пользователях
Публикация финальной версии

4. Сопоставьте виды требований с соответствующими разделами спецификации:

Виды требований:

- А. Функциональные требования
- Б. Требования к интерфейсам
- В. Требования к безопасности
- Г. Требования к производительности

Разделы документа спецификации требований (SRS):

1. Use-case диаграммы
2. Блок-схемы взаимодействия
3. Политики доступа
4. Системные характеристики

5. Установите правильную последовательность разделов в техническом задании согласно ГОСТ 34.602

Основания для разработки
Назначение системы
Требования к системе
Состав и содержание работ
Порядок контроля и приемки

6. Какой документ является основным для формализации требований к ПО?

Презентация для руководства
Спецификация требований к программному обеспечению (SRS)
Маркетинговый план
Штатное расписание

7. Какие участники должны участвовать в согласовании технической документации?

Заказчик
Главный разработчик
Случайные пользователи
Системный аналитик
Уборщица офиса

8. Какой раздел технического задания описывает условия приемки системы?

Требования к составу работ
Порядок контроля и приемки
Назначение системы
Источники разработки

9. Какие элементы включает структура стандарта организации? (Выберите 3)

Область применения
Личные заметки автора
Нормативные ссылки
Кейсы
Термины и определения

10. Чем принципиально отличается инструкция администратора от руководства пользователя?

Содержит технические детали настройки и обслуживания системы

Написана более простым языком

Включает только скриншоты интерфейса

Не требует утверждения

Содержит пошаговые инструкции действий пользователя

11. Цели разработки технических стандартов в программной инженерии

Каковы основные цели разработки технических стандартов в программной инженерии?

12. Участие стейкхолдеров в разработке ТЗ

Почему при разработке технического задания важно участие всех заинтересованных сторон (стейкхолдеров)?

13. Роль спецификации требований (SRS)

Какова роль спецификации требований в жизненном цикле ПО?

Раздел 3. Методологические основы разработки программных продуктов

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Сопоставьте принципы ООП с их определениями:

Принципы ООП:

А. Инкапсуляция

Б. Наследование

В. Полиморфизм

Определение (суть):

1. Скрытие внутренней реализации и объединение данных с методами

2. Создание новых классов на основе существующих

3. Возможность объектов с одинаковым интерфейсом иметь разную реализацию

2. Установите последовательность работы с git:

Клонирование репозитория

Создание ветки

Внесение изменений

Фиксация изменений

Отправка изменений

Слияние веток

3. Сопоставьте языки программирования с их типами:

Языки программирования:

А. Java

Б. Python

В. C++

Типы:

1. Компилируемый в байт-код

2. Интерпретируемый

3. Компилируемый

4. Установите последовательность работы с переменной:

Объявление

Инициализация

Использование

Изменение значения

5. Установите порядок создания класса:

Объявление класса

Добавление полей

Добавление методов

Создание объекта

6. Установите последовательность вызова методов:

Создание объекта

Вызов метода

Возврат результата

Использование результата

7. Установите порядок обработки ошибок:

Обнаружение ошибки

Выброс исключения

Перехват исключения

Обработка ошибки

8. Какой принцип ООП позволяет скрыть внутреннюю реализацию класса?

Наследование

Инкапсуляция

Полиморфизм

Абстракция

9. Какие из перечисленных языков являются строго типизированными? (Выберите 2 варианта)

Python

Java

JavaScript

C#

10. Что из перечисленного является преимуществом использования функций? (Выберите 3 варианта)

Повторное использование кода

Упрощение отладки

Увеличение скорости выполнения

Улучшение читаемости кода

Отложенный эффект скорости обработки данных

11. Что из перечисленного является хорошей практикой именования переменных?

Использование транслита

Описательные имена

Однобуквенные имена

Имена из одной буквы и последовательных чисел

12. Что выведет код на Python?

[1,2,3]

[1,2,3,4]

[4]

Ошибку

```
def change_list(lst):  
    lst.append(4)
```

```
my_list = [1,2,3]  
change_list(my_list)  
print(my_list)
```

13. Какой принцип ООП позволяет вызывать один и тот же метод для разных объектов, получая различное поведение?

Инкапсуляция

Наследование

Полиморфизм

Абстракция

14. Что из перечисленного НЕ является преимуществом наследования?

Повторное использование кода

Упрощение тестирования

Увеличение скорости выполнения программы

Создание иерархий классов

15. Преимущество использования шаблонов проектирования

Какое преимущество дает использование шаблонов проектирования?

16. Документирование сложных алгоритмов

Почему важно документировать сложные алгоритмы?

17. Правильная обработка исключений

Как влияет грамотная обработка исключений на надежность программы?

Раздел 4. Основы управления ИТ-проектами

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Сопоставьте этапы проекта с их ключевыми задачами:

Этапы проекта:

- А. Инициация
- Б. Планирование
- В. Реализация
- Г. Внедрение
- Д. Завершение

Ключевые задачи:

- 1. Определение целей, стейкхолдеров и экономического обоснования
- 2. Составление графика, бюджета и распределение ресурсов
- 3. Непосредственное создание функционала согласно спецификациям
- 4. Передача системы заказчику и обучение пользователей
- 5. Подведение итогов, анализ достижения KPI и документирование уроков

2. Сопоставьте роли в проекте с их обязанностями:

Роли:

- А. Владелец продукта (Product owner)
- Б. Скрам-мастер (Scrum master)
- В. Разработчик
- Г. Тестирующий

Обязанности:

- 1. Формирование требований и приоритетов
- 2. Обеспечение выполнения методологии
- 3. Написание кода по ТЗ
- 4. Проверка соответствия требованиям

3. Сопоставьте артефакты Agile с их назначением:

Артефакты:

- А. Бэклог продукта (Product backlog)
- Б. Бэклог спринта (Sprint backlog)
- В. Инкремент продукта
- Г. Диаграмма сгорания (Burndown chart)

Назначение:

- 1. Полный перечень требований
- 2. Задачи на спринт
- 3. Результат спринта
- 4. График выполнения работ

4. Установите правильный порядок этапов планирования проекта:

Определение целей и масштаба проекта

Идентификация заинтересованных сторон

Разработка иерархической структуры работ (WBS)

Оценка длительности задач

Создание сетевого графика

Утверждение плана проекта

5. Установите порядок проведения ретроспективы спринта:

Сбор данных о прошедшем спринте
Генерация идей улучшений
Принятие решений по изменениям
Включение улучшений в следующий спринт
Документирование результатов

6. Какие из перечисленных процессов относятся к фазе инициации проекта? (Выберите 3)

Разработка архитектуры системы
Определение стейкхолдеров
Разработка устава проекта
Оценка экономической целесообразности
Написание кода

7. Какой документ фиксирует требования к ИС на этапе планирования?

Отчет о тестировании
Техническое задание (ТЗ)
Пользовательская инструкция
Релиз-план

8. Какие роли обязательны в Scrum-команде? (Выберите 3 варианта)

Product Owner
Scrum Master
Разработчики
Системный архитектор
Аналитик

9. Какие артефакты создаются в рамках управления качеством? (Выберите 2 варианта)

Чек-листы тестирования
Маркетинг-план
Отчеты о дефектах
Финансовый отчет

10. Какой процесс выполняется на этапе завершения проекта?

Документирование уроков (Lessons Learned)
Написание кода
Рефакторинг
Тестирование

11. Основные цели фазы инициации проекта

Каковы основные цели фазы инициации проекта?

12. Важность ТЗ

Почему разработка технического задания критически важна для проекта?

13. Практический кейс

В проекте по созданию интернет-магазина разработчики начали кодировать функционал корзины, но позже выяснилось, что заказчик ожидал иной логики расчетов скидок. Какой этап был пропущен или выполнен некачественно?

Раздел 5. Процессы профессиональной разработки программных продуктов

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Команде нужно адаптировать opensource-CRM под требования заказчика. Сопоставьте задачи с этапами работы:

Этапы:

А. Анализ
Б. Доработка кода
В. Тестирование
Г. Внедрение

Этапы работы:

1. Исследование исходного кода и возможностей кастомизации
2. Написание кастомных модулей под бизнес-процессы
3. Проверка работы изменений на тестовом стенде
4. Настройка прав доступа под роли сотрудников

2. Установите правильную последовательность этапов разработки нового модуля для ERP-системы:

Анализ требований заказчика
Проектирование API и схемы БД
Написание кода (бэкенд + фронтенд)
Модульное тестирование
Интеграция с основным решением
Регрессионное тестирование
Документирование изменений

3. Какие из перечисленных действий относятся к этапу адаптации готового ПО? (Выберите 3 варианта)

Настройка параметров под требования заказчика
Написание кода ядра системы с нуля
Интеграция с внешними API заказчика
Модификация конфигурационных файлов
Разработка новой архитектуры базы данных

4. Какой инструмент используют для контроля версий при командной разработке?

Jira
Git
Jenkins
Docker

5. Роль документации при передаче ПО на сопровождение

Какова роль документации при передаче ПО на сопровождение?

Раздел 6. Процессы тестирования программных продуктов

Форма контроля/оценочное средство: Задача

Вопросы/Задания:

1. Сопоставьте виды тестирования с их целями:

Виды тестирования:

- А. Модульное
- Б. Интеграционное
- В. Регрессионное

Цели тестирования:

1. Проверка отдельных модулей
2. Проверка взаимодействия компонентов
3. Проверка, что изменения не сломали существующую функциональность

2. Сопоставьте виды тестирования с их целями:

Виды тестирования:

- А. Модульное
- Б. Интеграционное
- В. Нагрузочное
- Г. Регрессионное

Цели тестирования

1. Проверка работы отдельных функций
2. Проверка взаимодействия модулей
3. Проверка производительности
4. Проверка, что изменения не сломали старый функционал

3. Сопоставьте артефакты тестирования с их описанием:

Артефакты:

- А. Тест-план

Б. Тест-кейс

В. Чек-лист

Г. Баг-репорт

Описание:

1. Документ с описанием стратегии тестирования

2. Шаги для проверки конкретного сценария

3. Перечень пунктов для быстрой проверки

4. Описание дефекта с шагами воспроизведения

4. Установите порядок проведения регрессионного тестирования:

Выбор тест-кейсов для регрессии

Подготовка тестовых данных

Запуск тестов

Анализ результатов

Фиксация выявленных дефектов

5. Установите порядок работы с тестовыми данными

Определение требований к данным

Генерация тестовых данных

Маскирование персональных данных (если нужно)

Загрузка данных в тестовое окружение

6. Установите порядок действий при падении теста:

Проверка актуальности теста

Локализация проблемы (код/тест/данные)

Фиксация бага или корректировка теста

Повторный запуск теста

7. Сопоставьте уровни тестирования с примерами:

Уровни тестирования:

А. Модульное

Б. Интеграционное

В. Системное

Примеры:

1. Тест метода расчета скидки

2. Тест взаимодействия с платежным шлюзом

3. Тест оформления заказа от корзины до оплаты

8. Сопоставьте метрики тестирования с их формулами:

Метрики тестирования:

А. Покрытие кода

Б. Эффективность тестов

В. Плотность дефектов

Формулы:

1. $(\text{Протестированные строки кода} / \text{Общее количество строк}) \times 100\%$

2. $(\text{Количество найденных дефектов} / \text{Количество выполненных тест-кейсов})$

3. $(\text{Количество дефектов} / \text{Размер модуля в KLOC})$

9. Установите последовательность тест-дизайна:

Анализ требований

Создание тест-кейсов

Определение тестовых данных

Подготовка тестового окружения

10. Установите порядок работы с баг-репортом

Обнаружение дефекта

Запись шагов воспроизведения

Определение критичности (Blocker, Major, Minor)

Назначение исполнителя

Исправление и верификация

Заккрытие баг-репорта

11. Что проверяют при модульном тестировании? (Выберите 2 варианта)

Корректность работы отдельных функций
Скорость работы системы под нагрузкой
Взаимодействие между модулями
Обработку граничных случаев

12. Какая метрика НЕ используется для оценки качества тестирования?

Покрывтие кода (Code Coverage)
Количество найденных критических дефектов
Скорость написания кода
Эффективность тест-кейсов (Pass/Fail Rate)

13. Что проверяет интеграционное тестирование?

Отдельные функции модуля
Взаимодействие между компонентами системы
Производительность под нагрузкой
Удобство интерфейса

14. Какие из перечисленных практик улучшают процесс тестирования? (Выберите 3 варианта)

Раннее вовлечение тестировщиков в проект
Отказ от документации тест-кейсов
Использование тестовых данных, близких к реальным
Регулярный регрессионный тест

15. Какой тип тестирования выявляет проблемы с безопасностью? (Выберите 2 варианта)

Тестирование на SQL-инъекции
Юнит-тестирование
Проверка аутентификации
Модульное тестирование

16. Что НЕ включает тест-план?

Цели и объем тестирования
Критерии начала/окончания тестов
Исходный код приложения
Распределение ролей в команде

17. Какой этап тестирования выполняется последним?

Модульное тестирование
Интеграционное тестирование
Приемочное тестирование
Регрессионное тестирование

18. Какие из перечисленных дефектов относятся к критическим (Blocker)? (Выберите 2 варианта)

Падение системы при выполнении базового сценария
Опечатка в сообщении об ошибке
Невозможность завершить операцию
Несоответствие цвета кнопки макету

19. Чего НЕ делает тестировщик при обнаружении дефекта?

Записывает шаги воспроизведения
Определяет критичность дефекта
Немедленно исправляет код
Создает баг-репорт

20. Какой критерий является ключевым при оценке юзабилити интерфейса?

Скорость работы системы
Интуитивная понятность навигации

Количество строк кода
Коэффициент покрытия тестами

21. Тестирование в ЖЦ

Почему этап тестирования нельзя исключать из жизненного цикла?

22. Практический кейс: Проблемы с приемкой

После завершения разработки системы электронного документооборота заказчик отказывается подписывать акт сдачи-приемки, так как часть функций работает нестабильно. Какой этап не был выполнен должным образом?

23. Важность модельного тестирования

Почему модульное тестирование критически важно при разработке ПО?

24. Необходимость регрессионного тестирования после внесения изменений в код

Почему регрессионное тестирование необходимо после внесения изменений в код?

25. Преимущества автоматизации тестирования

Какие преимущества дает автоматизация тестирования?

Раздел 7. Промежуточная аттестация

Форма контроля/оценочное средство:

Вопросы/Задания:

.

7. Оценочные материалы промежуточной аттестации

Очная форма обучения, Пятый семестр, Экзамен

Контролируемые ИДК: ОПК-2.1 ОПК-4.1 ОПК-7.1 ОПК-8.1 ОПК-2.2 ОПК-4.2 ОПК-7.2 ОПК-8.2 ОПК-2.3 ОПК-4.3 ОПК-7.3 ОПК-8.3 ПК-П2.1 ПК-П8.1 ПК-П2.2 ПК-П8.2 ПК-П2.3 ПК-П8.3

Вопросы/Задания:

1. Программная инженерия как дисциплина, комплексно охватывающая процесс разработки информационных систем

2. Структурное программирование.

3. Процедурно-ориентированное программирование.

4. Объектно-ориентированное программирование.

5. Модульное программирование.

6. Современные парадигмы программирования.

7. Унифицированный язык моделирования UML

8. Методология проектирования Microsoft.

9. Гибкие методологии разработки ПО: Scrum.

10. Гибкие методологии разработки ПО: Kanban.

11. Спецификация требований: состав и структура документа.

12. Техническое задание: назначение, основные разделы документа.
13. Технический проект: назначение, основные разделы документа.
14. Структура ТЭО.
15. Рабочий проект: назначение, основные разделы документа.
16. Программа тестовых испытаний.
17. Документирование требований – основы языка Gherkin.
18. Понятие и структура тестового скрипта.
19. Документация пользователя.
20. Принципы и инструменты документирования кода.
21. Тестирование как процесс обеспечения качества ПС: понятие, цели, виды тестирования.
22. Автоматизация тестирования.
23. Разработка через тестирование (TDD).
24. Понятие и структура тестового скрипта.
25. Тестовое покрытие.
26. Системы контроля версий. Git.
27. Опытно-промышленная эксплуатация.
28. Стратегии развертывания системы.
29. Интеграционное тестирование системы.
30. Миграция данных.
31. Основные понятия ООП: инкапсуляция, наследование, полиморфизм.
32. Основы ООП: класс как абстрактный тип, конструкторы и деструкторы.
33. Основы ООП: объявление классов, объекты классов, указатель this.
34. Основы ООП: спецификаторы доступа класса.
35. Основы ООП: «друзья» классов (дружественные функции и классы).

36. Основы ООП: перегрузка и шаблоны функций.
37. Основы ООП: наследование классов, шаблоны классов.
38. Основы ООП: перегрузка стандартных операций.
39. Основы ООП: обработка исключительных ситуаций.
40. Унифицированный язык моделирования UML: цели создания и назначение.
41. Методология проектирования UML: диаграммы вариантов использования.
42. Методология проектирования UML: диаграммы состояний.
43. Методология проектирования UML: диаграммы деятельности.
44. Методология проектирования UML: диаграммы классов.
45. Методология проектирования UML: диаграммы последовательности.
46. Методология проектирования UML: диаграммы кооперации.
47. Методология проектирования UML: диаграммы компонентов.
48. Методология проектирования UML: диаграммы развертывания.
49. Методология проектирования Microsoft: схема слоев – назначение, пример.
50. Понятие и модели жизненного цикла программного средства.
51. Основные процессы жизненного цикла программного средства.
52. Гибкие методологии разработки ПО: сравнительный анализ, Scrum, преимущества и недостатки.
53. Авторская и коллективная разработка.
54. Общинная модель разработки ПО. Основные отличия от коллективной разработки.
55. Коллективная разработка. Модель группы и иерархическая модель.
56. Коллективная разработка: роли и обязанности членов проектной группы. Деструктивное сочетание ролей.
57. Коллективная разработка: роли, деструктивное и созидательное сочетание ролей.
58. Управление требованиями: определение требований, управления требованиями, место и роль управления требованиями в процессе жизненного цикла ПО.

59. Управление требованиями: классификация требований.

60. Управление требованиями: методы сбора требований.

61. Проектирование и разработка интерфейса ПС: понятие, эргономика, типы диалогов.

62. Проектирование и разработка интерфейса ПС: раскадровка, макетирование.

63. Проектирование и разработка интерфейса ПС: принципы построения интерфейсов.

64. Практика работы с требованиями

описать не менее 5 историй пользователя при выявлении требований к программному продукту. Примеры программных продуктов: будильник, секундомер, блокнот, графический редактор, музыкальный проигрыватель, почтовый клиент, электронный учебник, банкомат, кинокасса, заказ такси, оплата картой, пульсометр

65. Практика проектирования UML

построить требуемую диаграмму UML заданной предметной области (согласно программному продукту). Варианты диаграмм UML: ВИ, последовательности, активности, состояний

66. Практика ООП

описать класс с заданными компонентными данными и функциями. Не менее 2х компонентных данных, минимум 1 конструктор, не менее 3 компонентных функций. Класс по программному продукту

67. Практика Agile

спланировать итерации проекта разработки программного продукта с учетом стандартных этапов жизненного цикла программного продукта. Длительность итерации принять равной 1 дню

68. Практика использования языка Gherkin

описать на языке Gherkin пользовательский сценарий авторизации аккаунта для использования программного продукта. Сценарий должен иметь 1 основной и 1 альтернативный поток

Заочная форма обучения, Пятый семестр, Экзамен

Контролируемые ИДК: ОПК-2.1 ОПК-4.1 ОПК-7.1 ОПК-8.1 ОПК-2.2 ОПК-4.2 ОПК-7.2 ОПК-8.2 ОПК-2.3 ОПК-4.3 ОПК-7.3 ОПК-8.3 ПК-П2.1 ПК-П8.1 ПК-П2.2 ПК-П8.2 ПК-П2.3 ПК-П8.3

Вопросы/Задания:

1. Программная инженерия как дисциплина, комплексно охватывающая процесс разработки информационных систем

2. Структурное программирование.

3. Процедурно-ориентированное программирование.

4. Объектно-ориентированное программирование.

5. Модульное программирование.

6. Современные парадигмы программирования.

7. Унифицированный язык моделирования UML
8. Методология проектирования Microsoft.
9. Гибкие методологии разработки ПО: Scrum.
10. Гибкие методологии разработки ПО: Kanban.
11. Спецификация требований: состав и структура документа.
12. Техническое задание: назначение, основные разделы документа.
13. Технический проект: назначение, основные разделы документа.
14. Структура ТЭО.
15. Рабочий проект: назначение, основные разделы документа.
16. Программа тестовых испытаний.
17. Документирование требований – основы языка Gherkin.
18. Понятие и структура тестового скрипта.
19. Документация пользователя.
20. Принципы и инструменты документирования кода.
21. Тестирование как процесс обеспечения качества ПС: понятие, цели, виды тестирования.
22. Автоматизация тестирования.
23. Разработка через тестирование (TDD).
24. Понятие и структура тестового скрипта.
25. Тестовое покрытие.
26. Системы контроля версий. Git.
27. Опытнo-промышленная эксплуатация.
28. Стратегии развертывания системы.
29. Интеграционное тестирование системы.
30. Миграция данных.

31. Основные понятия ООП: инкапсуляция, наследование, полиморфизм.
32. Основы ООП: класс как абстрактный тип, конструкторы и деструкторы.
33. Основы ООП: объявление классов, объекты классов, указатель this.
34. Основы ООП: спецификаторы доступа класса.
35. Основы ООП: «друзья» классов (дружественные функции и классы).
36. Основы ООП: перегрузка и шаблоны функций.
37. Основы ООП: наследование классов, шаблоны классов.
38. Основы ООП: перегрузка стандартных операций.
39. Основы ООП: обработка исключительных ситуаций.
40. Унифицированный язык моделирования UML: цели создания и назначение.
41. Методология проектирования UML: диаграммы вариантов использования.
42. Методология проектирования UML: диаграммы состояний.
43. Методология проектирования UML: диаграммы деятельности.
44. Методология проектирования UML: диаграммы классов.
45. Методология проектирования UML: диаграммы последовательности.
46. Методология проектирования UML: диаграммы кооперации.
47. Методология проектирования UML: диаграммы компонентов.
48. Методология проектирования UML: диаграммы развертывания.
49. Методология проектирования Microsoft: схема слоев – назначение, пример.
50. Понятие и модели жизненного цикла программного средства.
51. Основные процессы жизненного цикла программного средства.
52. Гибкие методологии разработки ПО: сравнительный анализ, Scrumban, преимущества и недостатки.
53. Авторская и коллективная разработка.
54. Общинная модель разработки ПО. Основные отличия от коллективной разработки.

55. Коллективная разработка. Модель группы и иерархическая модель.
56. Коллективная разработка: роли и обязанности членов проектной группы. Деструктивное сочетание ролей.
57. Коллективная разработка: роли, деструктивное и созидательное сочетание ролей.
58. Управление требованиями: определение требований, управления требованиями, место и роль управления требованиями в процессе жизненного цикла ПО.
59. Управление требованиями: классификация требований.
60. Управление требованиями: методы сбора требований.
61. Проектирование и разработка интерфейса ПС: понятие, эргономика, типы диалогов.
62. Проектирование и разработка интерфейса ПС: раскадровка, макетирование.
63. Проектирование и разработка интерфейса ПС: принципы построения интерфейсов.
64. Практика работы с требованиями
описать не менее 5 историй пользователя при выявлении требований к программному продукту. Примеры программных продуктов: будильник, секундомер, блокнот, графический редактор, музыкальный проигрыватель, почтовый клиент, электронный учебник, банкомат, кинокасса, заказ такси, оплата картой, пульсометр
65. Практика проектирования UML
построить требуемую диаграмму UML заданной предметной области (согласно программному продукту). Варианты диаграмм UML: ВИ, последовательности, активности, состояний
66. Практика ООП
описать класс с заданными компонентными данными и функциями. Не менее 2х компонентных данных, минимум 1 конструктор, не менее 3 компонентных функций. Класс по программному продукту
67. Практика Agile
спланировать итерации проекта разработки программного продукта с учетом стандартных этапов жизненного цикла программного продукта. Длительность итерации принять равной 1 дню
68. Практика использования языка Gherkin
описать на языке Gherkin пользовательский сценарий авторизации аккаунта для использования программного продукта. Сценарий должен иметь 1 основной и 1 альтернативный поток

8. Материально-техническое и учебно-методическое обеспечение дисциплины

8.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. ЕФАНОВА Н. В. Программная инженерия: учебник / ЕФАНОВА Н. В., Иванова Е. А.. - Краснодар: КубГАУ, 2022. - 174 с. - 978-5-907667-60-0. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=12688> (дата обращения: 07.07.2025). - Режим доступа: по подписке

2. Маран М. М. Программная инженерия / Маран М. М.. - 3-е изд., стер. - Санкт-Петербург: Лань, 2022. - 196 с. - 978-5-8114-9323-4. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/189470.jpg> (дата обращения: 19.06.2025). - Режим доступа: по подписке

3. Системная и программная инженерия / Миронов А. Н., Воронцов Ю. А., Михайлова Е. К., Трушин С. М.. - Москва: РТУ МИРЭА, 2022. - 129 с. - Текст: электронный. // RuSpLAN: [сайт]. - URL: <https://e.lanbook.com/img/cover/book/310997.jpg> (дата обращения: 19.06.2025). - Режим доступа: по подписке

Дополнительная литература

1. Леоненков,, А. В. Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose: учебное пособие / А. В. Леоненков,. - Объектно-ориентированный анализ и проектирование с использованием UML и IBM Rational Rose - Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. - 317 с. - 978-5-4497-0667-6. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/97554.html> (дата обращения: 19.06.2025). - Режим доступа: по подписке

2. Программная инженерия. Часть II: учебное пособие / . - Программная инженерия. Часть II - Ставрополь: Северо-Кавказский федеральный университет, 2017. - 100 с. - 2227-8397. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/83193.html> (дата обращения: 19.06.2025). - Режим доступа: по подписке

3. Киселева,, Т. В. Программная инженерия. Часть 1: учебное пособие / Т. В. Киселева,. - Программная инженерия. Часть 1 - Ставрополь: Северо-Кавказский федеральный университет, 2017. - 137 с. - 2227-8397. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/69425.html> (дата обращения: 19.06.2025). - Режим доступа: по подписке

4. ЕФАНОВА Н. В. Программная инженерия: метод. рекомендации / ЕФАНОВА Н. В., Иванова Е. А.. - Краснодар: КубГАУ, 2020. - 35 с. - Текст: электронный. // : [сайт]. - URL: <https://edu.kubsau.ru/mod/resource/view.php?id=9081> (дата обращения: 07.07.2025). - Режим доступа: по подписке

5. Липаев,, В. В. Документирование сложных программных комплексов: электронное дополнение к учебному пособию «программная инженерия сложных заказных программных продуктов» (для бакалавров) / В. В. Липаев,. - Документирование сложных программных комплексов - Саратов: Вузовское образование, 2015. - 115 с. - 2227-8397. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/27294.html> (дата обращения: 19.06.2025). - Режим доступа: по подписке

6. Липаев,, В. В. Экономика программной инженерии заказных программных продуктов: дополнение к учебному пособию «программная инженерия сложных заказных программных продуктов» (для бакалавров) / В. В. Липаев,. - Экономика программной инженерии заказных программных продуктов - Саратов: Вузовское образование, 2015. - 139 с. - 2227-8397. - Текст: электронный // IPR SMART: [сайт]. - URL: <https://www.iprbookshop.ru/27303.html> (дата обращения: 19.06.2025). - Режим доступа: по подписке

8.2. Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся

Профессиональные базы данных

Не используются.

Ресурсы «Интернет»

1. <https://edu.kubsau.ru/> - Образовательный портал КубГАУ
2. <http://www.iprbookshop.ru/> - IPRbook
3. <https://znanium.com/> - Znanium.com

8.3. Программное обеспечение и информационно-справочные системы, используемые при осуществлении образовательного процесса по дисциплине

Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине позволяют:

- обеспечить взаимодействие между участниками образовательного процесса, в том числе синхронное и (или) асинхронное взаимодействие посредством сети «Интернет»;
- фиксировать ход образовательного процесса, результатов промежуточной аттестации по дисциплине и результатов освоения образовательной программы;
- организовать процесс образования путем визуализации изучаемой информации посредством использования презентаций, учебных фильмов;
- контролировать результаты обучения на основе компьютерного тестирования.

Перечень лицензионного программного обеспечения:

1 Microsoft Windows - операционная система.

2 Microsoft Office (включает Word, Excel, Power Point) - пакет офисных приложений.

Перечень профессиональных баз данных и информационных справочных систем:

1 Гарант - правовая, <https://www.garant.ru/>

2 Консультант - правовая, <https://www.consultant.ru/>

3 Научная электронная библиотека eLibrary - универсальная, <https://elibrary.ru/>

Доступ к сети Интернет, доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения

(обновление производится по мере появления новых версий программы)

1. Индиго;

Перечень информационно-справочных систем

(обновление выполняется еженедельно)

Не используется.

8.4. Специальные помещения, лаборатории и лабораторное оборудование

Университет располагает на праве собственности или ином законном основании материально-техническим обеспечением образовательной деятельности (помещениями и оборудованием) для реализации программы бакалавриата, специалитета, магистратуры по Блоку 1 "Дисциплины (модули)" и Блоку 3 "Государственная итоговая аттестация" в соответствии с учебным планом.

Каждый обучающийся в течение всего периода обучения обеспечен индивидуальным неограниченным доступом к электронной информационно-образовательной среде университета из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети "Интернет", как на территории университета, так и вне его. Условия для функционирования электронной информационно-образовательной среды могут быть созданы с использованием ресурсов иных организаций.

Лекционный зал

310эк

- 0 шт.

Компьютерный класс

401эк

Персональный компьютер IRU i5/16Gb/512GbSSD/23.8 - 1 шт.

402эк

Персональный компьютер IRU i5/16Gb/512GbSSD/23.8 - 1 шт.

404эк

Персональный компьютер UNIVERSALD1 i5/16Gb/512GbSSD/23.8 - 1 шт.
408эк

Персональный компьютер IRU i5/16Gb/512GbSSD/23.8 - 1 шт.

9. Методические указания по освоению дисциплины (модуля)

Учебная работа по направлению подготовки осуществляется в форме контактной работы с преподавателем, самостоятельной работы обучающегося, текущей и промежуточной аттестаций, иных формах, предлагаемых университетом. Учебный материал дисциплины структурирован и его изучение производится в тематической последовательности. Содержание методических указаний должно соответствовать требованиям Федерального государственного образовательного стандарта и учебных программ по дисциплине. Самостоятельная работа студентов может быть выполнена с помощью материалов, размещенных на портале поддержки Moodle.

Методические указания по формам работы

Лекционные занятия

Передача значительного объема систематизированной информации в устной форме достаточно большой аудитории. Дает возможность экономно и систематично излагать учебный материал. Обучающиеся изучают лекционный материал, размещенный на портале поддержки обучения Moodle.

Лабораторные занятия

Практическое освоение студентами научно-теоретических положений изучаемого предмета, овладение ими техникой экспериментирования в соответствующей отрасли науки. Лабораторные занятия проводятся с использованием методических указаний, размещенных на образовательном портале университета.

Описание возможностей изучения дисциплины лицами с ОВЗ и инвалидами

Для инвалидов и лиц с ОВЗ может изменяться объём дисциплины (модуля) в часах, выделенных на контактную работу обучающегося с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающегося (при этом не увеличивается количество зачётных единиц, выделенных на освоение дисциплины).

Фонды оценочных средств адаптируются к ограничениям здоровья и восприятия информации обучающимися.

Основные формы представления оценочных средств – в печатной форме или в форме электронного документа.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением зрения:

- устная проверка: дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.;
- с использованием компьютера и специального ПО: работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, дистанционные формы, если позволяет острота зрения - графические работы и др.;
- при возможности письменная проверка с использованием рельефно-точечной системы Брайля, увеличенного шрифта, использование специальных технических средств (тифлотехнических средств): контрольные, графические работы, тестирование, домашние задания, эссе, отчеты и др.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением слуха:

- письменная проверка: контрольные, графические работы, тестирование, домашние задания, эссе, письменные коллоквиумы, отчеты и др.;

- с использованием компьютера: работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, графические работы, дистанционные формы и др.;

- при возможности устная проверка с использованием специальных технических средств (аудиосредств, средств коммуникации, звукоусиливающей аппаратуры и др.): дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.

Формы контроля и оценки результатов обучения инвалидов и лиц с ОВЗ с нарушением опорно-двигательного аппарата:

- письменная проверка с использованием специальных технических средств (альтернативных средств ввода, управления компьютером и др.): контрольные, графические работы, тестирование, домашние задания, эссе, письменные коллоквиумы, отчеты и др.;

- устная проверка, с использованием специальных технических средств (средств коммуникаций): дискуссии, тренинги, круглые столы, собеседования, устные коллоквиумы и др.;

- с использованием компьютера и специального ПО (альтернативных средств ввода и управления компьютером и др.): работа с электронными образовательными ресурсами, тестирование, рефераты, курсовые проекты, графические работы, дистанционные формы предпочтительнее обучающимся, ограниченным в передвижении и др.

Адаптация процедуры проведения промежуточной аттестации для инвалидов и лиц с ОВЗ.

В ходе проведения промежуточной аттестации предусмотрено:

- предъявление обучающимся печатных и (или) электронных материалов в формах, адаптированных к ограничениям их здоровья;

- возможность пользоваться индивидуальными устройствами и средствами, позволяющими адаптировать материалы, осуществлять приём и передачу информации с учетом их индивидуальных особенностей;

- увеличение продолжительности проведения аттестации;

- возможность присутствия ассистента и оказания им необходимой помощи (занять рабочее место, передвигаться, прочитать и оформить задание, общаться с преподавателем).

Формы промежуточной аттестации для инвалидов и лиц с ОВЗ должны учитывать индивидуальные и психофизические особенности обучающегося/обучающихся по АОПОП ВО (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями зрения:

- предоставление образовательного контента в текстовом электронном формате, позволяющем переводить плоскостную информацию в аудиальную или тактильную форму;

- возможность использовать индивидуальные устройства и средства, позволяющие адаптировать материалы, осуществлять приём и передачу информации с учетом индивидуальных особенностей и состояния здоровья студента;

- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;

- использование чёткого и увеличенного по размеру шрифта и графических объектов в мультимедийных презентациях;

- использование инструментов «лупа», «проектор» при работе с интерактивной доской;

- озвучивание визуальной информации, представленной обучающимся в ходе занятий;

- обеспечение раздаточным материалом, дублирующим информацию, выводимую на экран;

- наличие подписей и описания у всех используемых в процессе обучения рисунков и иных графических объектов, что даёт возможность перевести письменный текст в аудиальный;

- обеспечение особого речевого режима преподавания: лекции читаются громко, разборчиво, отчётливо, с паузами между смысловыми блоками информации, обеспечивается интонирование, повторение, акцентирование, профилактика рассеивания внимания;

- минимизация внешнего шума и обеспечение спокойной аудиальной обстановки;

- возможность вести запись учебной информации студентами в удобной для них форме (аудиально, аудиовизуально, на ноутбуке, в виде пометок в заранее подготовленном тексте);

- увеличение доли методов социальной стимуляции (обращение внимания, апелляция к

ограничениям по времени, контактные виды работ, групповые задания и др.) на практических и лабораторных занятиях;

- минимизирование заданий, требующих активного использования зрительной памяти и зрительного внимания;
- применение поэтапной системы контроля, более частый контроль выполнения заданий для самостоятельной работы.

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями опорно-двигательного аппарата (маломобильные студенты, студенты, имеющие трудности передвижения и патологию верхних конечностей):

- возможность использовать специальное программное обеспечение и специальное оборудование и позволяющее компенсировать двигательное нарушение (коляски, ходунки, трости и др.);
- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;
- применение дополнительных средств активизации процессов запоминания и повторения;
- опора на определенные и точные понятия;
- использование для иллюстрации конкретных примеров;
- применение вопросов для мониторинга понимания;
- разделение изучаемого материала на небольшие логические блоки;
- увеличение доли конкретного материала и соблюдение принципа от простого к сложному при объяснении материала;
- наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
- увеличение доли методов социальной стимуляции (обращение внимания, апелляция к ограничениям по времени, контактные виды работ, групповые задания др.);
- обеспечение беспрепятственного доступа в помещения, а также пребывания в них;
- наличие возможности использовать индивидуальные устройства и средства, позволяющие обеспечить реализацию эргономических принципов и комфортное пребывание на месте в течение всего периода учёбы (подставки, специальные подушки и др.).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с нарушениями слуха (глухие, слабослышащие, позднооглохшие):

- предоставление образовательного контента в текстовом электронном формате, позволяющем переводить аудиальную форму лекции в плоскпечатную информацию;
- наличие возможности использовать индивидуальные звукоусиливающие устройства и сурдотехнические средства, позволяющие осуществлять приём и передачу информации; осуществлять взаимобратный перевод текстовых и аудиофайлов (блокнот для речевого ввода), а также запись и воспроизведение зрительной информации;
- наличие системы заданий, обеспечивающих систематизацию вербального материала, его схематизацию, перевод в таблицы, схемы, опорные тексты, глоссарий;
- наличие наглядного сопровождения изучаемого материала (структурно-логические схемы, таблицы, графики, концентрирующие и обобщающие информацию, опорные конспекты, раздаточный материал);
- наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
- обеспечение практики опережающего чтения, когда студенты заранее знакомятся с материалом и выделяют незнакомые и непонятные слова и фрагменты;
- особый речевой режим работы (отказ от длинных фраз и сложных предложений, хорошая артикуляция; четкость изложения, отсутствие лишних слов; повторение фраз без изменения слов и порядка их следования; обеспечение зрительного контакта во время говорения и чуть более медленного темпа речи, использование естественных жестов и мимики);
- чёткое соблюдение алгоритма занятия и заданий для самостоятельной работы (называние темы, постановка цели, сообщение и запись плана, выделение основных понятий и методов их изучения, указание видов деятельности студентов и способов проверки усвоения материала, словарная работа);

- соблюдение требований к предъявляемым учебным текстам (разбивка текста на части; выделение опорных смысловых пунктов; использование наглядных средств);
- минимизация внешних шумов;
- предоставление возможности соотносить вербальный и графический материал; комплексное использование письменных и устных средств коммуникации при работе в группе;
- сочетание на занятиях всех видов речевой деятельности (говорения, слушания, чтения, письма, зрительного восприятия с лица говорящего).

Специальные условия, обеспечиваемые в процессе преподавания дисциплины студентам с прочими видами нарушений (ДЦП с нарушениями речи, заболевания эндокринной, центральной нервной и сердечно-сосудистой систем, онкологические заболевания):

- наличие возможности использовать индивидуальные устройства и средства, позволяющие осуществлять приём и передачу информации;
- наличие системы заданий, обеспечивающих систематизацию вербального материала, его схематизацию, перевод в таблицы, схемы, опорные тексты, глоссарий;
- наличие наглядного сопровождения изучаемого материала;
- наличие чёткой системы и алгоритма организации самостоятельных работ и проверки заданий с обязательной корректировкой и комментариями;
- обеспечение практики опережающего чтения, когда студенты заранее знакомятся с материалом и выделяют незнакомые и непонятные слова и фрагменты;
- предоставление возможности соотносить вербальный и графический материал; комплексное использование письменных и устных средств коммуникации при работе в группе;
- сочетание на занятиях всех видов речевой деятельности (говорения, слушания, чтения, письма, зрительного восприятия с лица говорящего);
- предоставление образовательного контента в текстовом электронном формате;
- предоставление возможности предкурсового ознакомления с содержанием учебной дисциплины и материалом по курсу за счёт размещения информации на корпоративном образовательном портале;
- возможность вести запись учебной информации студентами в удобной для них форме (аудиально, аудиовизуально, в виде пометок в заранее подготовленном тексте);
- применение поэтапной системы контроля, более частый контроль выполнения заданий для самостоятельной работы;
- стимулирование выработки у студентов навыков самоорганизации и самоконтроля;
- наличие пауз для отдыха и смены видов деятельности по ходу занятия.

10. Методические рекомендации по освоению дисциплины (модуля)

Дисциплина "Программная инженерия" ведётся в соответствии с календарным учебным планом и расписанием занятий по неделям. Темы проведения занятий определяются тематическим планом рабочей программы дисциплины.