

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Котова Лариса Анатольевна
Должность: Директор филиала
Дата подписания: 01.06.2026 19:25:47
Уникальный программный ключ:
10730ffe6b1ed036b744b6e9d97700b86e5c04a7

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский технологический университет «МИСИС»
Новотроицкий филиал

Приложение 4

к ОПОП ВО 09.03.03 Прикладная информатика
Прикладная информатика в технических системах

Рабочая программа дисциплины

Программная инженерия

Закреплена за подразделением	Кафедра математики и естествознания (Новотроицкий филиал)	
Направление подготовки	09.03.03 Прикладная информатика	
Образовательная программа	09.03.03 Прикладная информатика / Прикладная информатика в технических системах	
Квалификация	Бакалавр	
Форма обучения	очная	
Общая трудоемкость	7 ЗЕТ	Виды контроля в семестрах: зачет 6 экзамен 7 курсовая работа 7 контрольная работа 6
Часов по учебному плану	252	

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	6 (3.2)		7 (4.1)		Итого	
	уп	рп	уп	рп		
Неделя	10		19			
Вид занятий	уп	рп	уп	рп	уп	рп
Лекции	18	18	17	17	35	35
Лабораторные	27	27	34	34	61	61
Практические	27	27	34	34	61	61
Итого ауд.	72	72	85	85	157	157
Контактная работа	72	72	85	85	157	157
Сам. работа	36	36	32	32	68	68
В том числе сам. работа в рамках ФОС		8		4		
Часы на контроль			27	27	27	27
Итого	108	108	144	144	252	252

Программу составил(и):

к.п.н, Доцент, Абдулвелеева Р.Р.

Рабочая программа дисциплины

Программная инженерия

Составлен на основании учебного плана:

09.03.03_24_Прикладная информатика_ПрПИвТС.plx, утвержденного Ученым советом НИТУ МИСИС в составе соответствующей ОПОП ВО 09.03.03 Прикладная информатика Прикладная информатика в технических системах протокол от 27.11.2025 №68.

Рабочая программа одобрена на заседании

Кафедра математики и естествознания (Новотроицкий филиал)

Протокол от 11.03.2026 г., №3.

Руководитель подразделения Швалёва Анна Викторовна.

1. ЦЕЛИ ОСВОЕНИЯ

1.1	Цели освоения дисциплины: формирование у обучающихся представления о современных процессах проектирования, разработки, тестирования и эксплуатации программного продукта и о взаимосвязи всех аспектов программной инженерии.
1.2	Задачи:
1.3	- изучить понятийный аппарат дисциплины, основные теоретические положения и методы;
1.4	- сформировать умения и навыки применения теоретических знаний для решения профессиональных задач.

2. МЕСТО В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Блок ОП:		Б1.В
2.1	Требования к предварительной подготовке обучающегося:	
2.1.1	Языки и среды разработки интернет-приложений	
2.1.2	Алгоритмизация и программирование	
2.1.3	Информационные системы и технологии	
2.1.4	Технологии программирования	
2.1.5	Экономика	
2.1.6	Языки программирования	
2.1.7	Компьютерная графика	
2.1.8	Вычислительные системы, сети и телекоммуникации	
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:	
2.2.1	Теоретическая механика	
2.2.2	Подготовка к процедуре защиты и защита выпускной квалификационной работы	
2.2.3	Моделирование металлургических процессов с использованием современных программных продуктов	
2.2.4	Электротехника, электроника и схемотехника	

3. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ФОРМИРУЕМЫМИ КОМПЕТЕНЦИЯМИ

УК-2: Способен собирать и интерпретировать данные и определять круг задач в рамках поставленной цели, выбирать оптимальные способы решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений, умение обосновывать принятые решения
Знать:
УК-2-31 способы интерпретации данных, определения круга задач в рамках поставленной цели, оптимальные способы решения и обоснования, исходя из действующих правовых норм, имеющихся ресурсов и ограничений
ОПК-7: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, проектировать и разрабатывать алгоритмы и программы, пригодные для практического применения
Знать:
ОПК-7-31 основные технологии создания и внедрения информационных систем, стандарты управления жизненным циклом информационной системы
УК-2: Способен собирать и интерпретировать данные и определять круг задач в рамках поставленной цели, выбирать оптимальные способы решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений, умение обосновывать принятые решения
Уметь:
УК-2-У1 осуществлять организационное обеспечение выполнения работ на всех стадиях и в процессах жизненного цикла информационной системы
ОПК-7: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, проектировать и разрабатывать алгоритмы и программы, пригодные для практического применения
Уметь:
ОПК-7-У1 формулировать требования к ИС
УК-2: Способен собирать и интерпретировать данные и определять круг задач в рамках поставленной цели, выбирать оптимальные способы решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений, умение обосновывать принятые решения
Владеть:
УК-2-В1 навыками составления плановой и отчетной документации по управлению проектами создания информационных систем на стадиях жизненного цикла

ОПК-7: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, проектировать и разрабатывать алгоритмы и программы, пригодные для практического применения

Владеть:

ОПК-7-В1 навыками составления плановой и отчетной документации по управлению проектами создания информационных систем на стадиях жизненного цикла

4. СТРУКТУРА И СОДЕРЖАНИЕ

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Формируемые индикаторы компетенций	Литература и эл. ресурсы	Примечание	КМ	Выполняемые работы
	Раздел 1. Основные понятия и место программной инженерии							
1.1	Рынок программного обеспечения. Обзор технологий программирования (структурное, модульное, объектно-ориентированное, компонентное программирование). Программный продукт и его основные характеристики. Составляющие стоимости ПО. Предмет программной инженерии. Основные понятия. /Лек/	6	4	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
1.2	Самостоятельное изучение учебного материала в LMS Moodle: Программная инженерия и ее отличия от информатики и других инженерий. Программная инженерия и ее отличия от информатики и других инженерий. Важность предмета. Сложность управления процессом разработки программного обеспечения. Технологии программирования как способ борьбы со сложностью. /Ср/	6	2	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
1.3	Выбор и обоснование темы проекта /Лаб/	6	9	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2			
1.4	Представление выбранной темы проекта на предмет актуальности, постановки цели и задач проекта. /Пр/	6	5	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1			Р2
	Раздел 2. Жизненный цикл ПО: процессы, модели, стандарты							

2.1	Программный процесс и модель программного процесса. Методы программной инженерии. Роль стандартов в программной инженерии. Основные стандарты программной инженерии. Жизненный цикл программного продукта. Процесс, действие, задача жизненного цикла. Фазы (этапы) жизненного цикла и их связь с процессами. Основные процессы жизненного цикла ПО (ISO12207 и ISO 15504). Вспомогательные процессы жизненного цикла ПО (ISO12207 и ISO 15504). Организационные процессы жизненного цикла ПО (ISO12207 и ISO 15504). /Лек/	6	6	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
2.2	Самостоятельное изучение учебного материала в LMS: Каскадная модель ЖЦ ПО. Преимущества, недостатки, применимость. Спиральная модель ЖЦ ПО. Преимущества, недостатки, применимость. Обзор других типов моделей ЖЦ ПО. Особенности моделей жизненного цикла MSF, RUP, XP. Основные трудности и проблемы программной инженерии. Профессиональные и этические требования ИТ-специалиста. Кодекс этики IEEE-CS/ACM. /Ср/	6	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
2.3	Модели жизненного цикла программной системы. Спецификация требований. /Пр/	6	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			Р3
2.4	Методология и стандарты создания программного обеспечения. /Лаб/	6	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
	Раздел 3. Анализ предметной области и требований к ПО							
3.1	Знакомство с требованиями к разрабатываемому ПО. Фиксация требований к ПО. Составление спецификации требований. /Лаб/	6	10	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			

3.2	Анализ предметной области и требования к ПО. Разработка компонентов модели данных приложения. Выдача заданий для контрольной работы. /Пр/	6	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			Р4
3.3	Моделирование потребности заказчика. Методы выявления требований. Процесс анализа предметной области. Разработка модели системы в шаблоне «ввод-обработка-вывод». Принципы анализа: информационная область, моделирование, разделение на части, ракурсы видения основной информации и деталей реализации. Элементы модели анализа. Моделирование данных: объекты, свойства и связи данных, словарь данных, диаграммы связей между объектами. /Лек/	6	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
3.4	Самостоятельное изучение учебного материала в LMS: Моделирование поведения. Диаграммы перехода состояний, таблицы решений, схемы диалога с пользователем. Выполнение структурного анализа: создание диаграммы связей между объектами, модели потока данных, модели поведения. Объектно-ориентированный (ОО) анализ: сравнение подходов. Базовые компоненты модели ОО анализа. Процесс ОО анализа. /Ср/	6	6	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
3.5	Подготовка к экзамену /Ср/	6	6	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4			
Раздел 4. Архитектура ПО								
4.1	Самостоятельное изучение учебного материала в LMS: Создание простых UML-диаграмм классов, пакетов и компонентов. Анализ, архитектура и проектирование простых систем «клиент-сервер» с использованием UML и акцентом на диаграммах классов и состояний. Выполнение контрольной работы. Подготовка к экзамену. /Ср/	6	6	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			

4.2	Понятие архитектуры ПО. Цели и принципы системного проектирования сложных программных средств. Процессы системного проектирования программных средств. Структурное проектирование сложных программных средств. Проектирование программных модулей и компонентов. Основные понятия, модели, задачи, особенности объектно-ориентированного проектирования программных средств. Средства объектно-ориентированного проектирования программных средств. /Лек/	7	4	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4			
4.3	Разработка внутренней структуры приложений при помощи диаграмм классов. Методологии разработки ПО. Проектирование и создание базы данных приложения. /Лаб/	7	10	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
4.4	Принципы создания удобного пользовательского интерфейса. Определение архитектуры пользовательского интерфейса приложения. /Пр/	7	10	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			Р5
Раздел 5. Качество ПО								

5.1	Стандартизация качества. Методы обеспечения качества ПО. Понятие тестирования. Инструменты тестирования. Критерии тестирования. Виды тестирования. Работа с ошибками. Средства контроля ошибок (bug tracking systems). Принципы верификации и тестирования программ. Процессы и средства тестирования программных компонентов. Процессы оценивания характеристик и испытания программных средств. Организация и методы оценивания характеристик сложных комплексов программ. Средства для испытаний и определения характеристик сложных комплексов программ. Оценивание надежности и безопасности функционирования сложных программных средств. /Лек/	7	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
5.2	Самостоятельное изучение учебного материала в LMS: Процессы и средства тестирования программных компонентов. Процессы оценивания характеристик и испытания программных средств. Организация и методы оценивания характеристик сложных комплексов программ. Средства для испытаний и определения характеристик сложных комплексов программ. Оценивание надежности и безопасности функционирования сложных программных средств. /Ср/	7	20	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
5.3	Методы верификации и тестирования программ и систем. Разработка системных тестов для приложения. Выдача задания для курсовой работы. /Пр/	7	10	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			P1
5.4	Использование отладчиков в процессе тестирования программных средств. Тестирование операций ввода/вывода. /Лаб/	7	10	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
Раздел 6. Сопровождение ПО								

6.1	Организация и методы сопровождения программных средств. Этапы и процедуры при сопровождении программных средств. Задачи и процессы переноса программ и данных на иные платформы. Ресурсы, для обеспечения сопровождения и мониторинга программных средств. /Лек/	7	3	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
6.2	Выявление факторов, определяющих потребность в сопровождении программного обеспечения. Выявление категорий программного обеспечения, нуждающегося в сопровождении. /Пр/	7	10	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			Р1
6.3	Составление заявок предложений о модификации и поиски возможности их удовлетворения. Организация работ по сопровождению информационных систем. /Лаб/	7	10	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
6.4	Выполнение курсовой работы. /Ср/	7	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1 Л2.2Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
Раздел 7. Конструирование ПО								
7.1	Управление проектами. Категории управления проектами. Ролевая модель команды. Модель управления командой. Особенности, преимущества и недостатки. Управление проектом разработки программного обеспечения. Планирование. Диаграмма Ганта. Средства управления проектом. Функции систем управления проектом. /Лек/	7	2	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			
7.2	Области управленческих знаний. Компетенции менеджера IT проекта. Роль и способы общения в команде. Преимущества и недостатки различных способов общения. Основные понятия модели технологической. /Пр/	7	4	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4			Р1

7.3	Основы принятий управленческих решений. /Пр/	6	6	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4				P1
7.4	Оценка трудоемкости и стоимости проекта. /Лаб/	7	4	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3.1 Л3.2 Л3.3 Л3.4 Э1 Э2 Э3 Э4				
Раздел 8. Подготовка к контрольным мероприятиям и выполняемым работам									
8.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	7	0	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л3.1 Л3.2 Л3.3 Л3.4				
8.2	Объем часов самостоятельной работы на подготовку к ВР /Ср/	7	4	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л3.1 Л3.2 Л3.3 Л3.4				
Раздел 9. Подготовка к контрольным мероприятиям и выполняемым работам									
9.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	6	0	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л3.1 Л3.2 Л3.3 Л3.4				
9.2	Объем часов самостоятельной работы на подготовку к ВР /Ср/	6	8	УК-2-31 УК-2-У1 УК-2-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л3.1 Л3.2 Л3.3 Л3.4				

5. ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ

5.1. Контрольные мероприятия (контрольная работа, тест, коллоквиум, экзамен и т.п), вопросы для самостоятельной подготовки

Код КМ	Контрольное мероприятие	Проверяемые индикаторы компетенций	Вопросы для подготовки
--------	-------------------------	------------------------------------	------------------------

КМ1	Контрольные вопросы	УК-2-31;ОПК-7-31	<p>1. Программная инженерия – это</p> <p>а) Совокупность инструментальных средств и методов, предназначенных для создания качественного программного обеспечения.</p> <p>б) Совокупность инструментальных средств, предназначенных для создания качественного программного обеспечения.</p> <p>в) Совокупность навыков, инструментальных средств и методов, предназначенных для создания качественного программного обеспечения.</p> <p>г) Наука, изучающая построение программных систем</p> <p>д) Правила проектирования систем со сложной архитектурой</p> <p>2. Программная инженерия занимается</p> <p>а) Вопросами оптимизации кода</p> <p>б) Вопросами разработки новых алгоритмов обработки данных</p> <p>в) Вопросами эффективной разработки программного обеспечения</p> <p>г) Применением средств быстрой разработки программного обеспечения</p> <p>д) Применением средств автоматизированного тестирования программного обеспечения</p> <p>3. Стадии разработки программных систем, общие формы алгоритмов и схем, описывающих эти системы, регламентируются</p> <p>а) Стандартами ЕСПД</p> <p>б) Пунктами ТЗ</p> <p>в) Никак не регламентируются</p> <p>г) Эксплуатационными документами</p> <p>д) Спецификацией ПС</p> <p>4. Псевдокод представляет собой</p> <p>а) Частично формализованный язык для представления описаний метода пошаговой детализации</p> <p>б) Язык, использующий конструкции структурного программирования</p> <p>в) Язык программирования высокого уровня</p> <p>г) Язык с неформальными фрагментами на естественном языке для представления обобщенных операторов и условий</p> <p>д) Формальная запись конструкций языка программирования Фортран</p>
-----	---------------------	------------------	---

КМ2	Контрольные вопросы	УК-2-31;ОПК-7-31	<p>2. Укажите основные процессы жизненного цикла по ГОСТ Р ИСО/МЭК 12207-99. «Информационная технология. Процессы жизненного цикла программных средств»</p> <p>а) Процесс заказа б) Процесс документирования в) Процесс разработки г) Процесс управления д) Процесс сопровождения</p> <p>3. Проблемы, решаемые конфигурационным управлением</p> <p>а) Работа в команде б) Одновременная модификация в) Ограниченное уведомление г) Управление пользователями д) Множество версий</p> <p>4. Этапы последовательной разработки ("водопад")</p> <p>а) Снятие с эксплуатации б) Тестирование в) Анализ требований г) Проектирование д) Системный анализ е) Использование и сопровождение</p> <p>8. Этапы итеративного цикла разработки</p> <p>а) Тестирование б) бизнес-моделирование в) Реализация г) Анализ и проектирование д) Требования</p> <p>5. Порядок разработки программного модуля</p> <p>а) Программирование (кодирование) модуля б) Шлифовка текста модуля в) Изучение и проверка спецификации модуля, выбор языка программирования г) Выбор алгоритма и структуры данных д) Компиляция модуля е) Проверка модуля</p>
-----	---------------------	------------------	--

КМЗ	Лабораторная работа	УК-2-31;ОПК-7-31	<p>РАЗРАБОТКА СТРУКТУРЫ СИСТЕМЫ</p> <p>Построение структурной схемы программной системы. На данном этапе система по функциональному признаку разделяется на основные подсистемы, между ними указываются информационные связи и/или связи по управлению, описывается основное назначение подсистем. При разработке структурной схемы используется методология структурного проектирования, в основе которой лежит алгоритмическая декомпозиция и иерархия вида «часть-целое», учитывающая, что внутренние связи элементов внутри подсистем сильнее, чем связь между подсистемами. Декомпозиция системы может повторяться многократно, вплоть до уровня конкретных процедур, при этом должна быть обеспечена целостность системы, а все составляющие компоненты взаимоувязаны. Для этого используются такие принципы разработки, как «сверху-вниз», «разделяй и властвуй», «иерархическое упорядочивание» и другие.</p> <p>В первом приближении можно придерживаться нормативного понятия системы. Система (греч. - «составленное из частей», «соединение» от «соединяю») - множество элементов, находящихся в отношениях и связях друг с другом, которое образует определённую целостность, единство.</p> <p>Главным свойством системы является ее целостность: комплекс объектов, рассматриваемых в качестве системы, должен обладать общими свойствами и поведением. Необходимо рассматривать связи системы с внешней средой.</p> <p>В самом общем случае понятие «система» характеризуется:</p> <ul style="list-style-type: none"> наличием множества элементов; наличием связей между ними; целостным характером данного устройства или процесса. <p>Система должна представлять собой совокупность элементов (объектов, субъектов), находящихся между собой в определенной зависимости и составляющих некоторое единство (целостность), направленное на достижение определенной цели. Система может являться элементом другой системы более высокого порядка (надсистема) и включать в себя системы более низкого порядка (подсистемы). То есть систему можно рассматривать как набор подсистем, организованных для достижения определенной цели и описанных с помощью набора моделей, а подсистему – как группу элементов, часть которых составляет спецификацию поведения, представленного другими ее составляющими.</p> <p>К типовым можно отнести следующие подсистемы:</p> <ul style="list-style-type: none"> подсистему управления; подсистемы ввода-вывода; подсистему настройки параметров; файловую подсистему; подсистему визуализации; подсистему документирования; подсистему взаимодействия с базой данных; справочную подсистему. <p>Полученная в результате декомпозиции структура системы должна сопровождаться кратким описанием включенных в нее подсистем.</p> <p>В состав системы входят следующие подсистемы:</p> <ul style="list-style-type: none"> Подсистема управления, которая отвечает за взаимодействие подсистем между собой и представлена в виде иерархического меню; Справочная подсистема, которая содержит сведения о системе (руководство пользователю) и ее ее разработчиках.
-----	---------------------	------------------	---

КМ4	Экзамен	УК-2-31;ОПК-7-31	<ol style="list-style-type: none"> 1. Основные понятия и определения программной инженерии. 2. IT-проекты: понятие, особенности реализации, статистика успешности, причины неудач. 3. Инженерия программного обеспечения (программная инженерия): история, определения, инженерная деятельность, область действия программной инженерии. 4. Требования к ПО: функциональные и нефункциональные. 5. Проектирование ПО: виды дизайна, цели архитектуры, техники проектирования. 6. Проектирование ПО: ключевые вопросы проектирования, уровни архитектуры, оценка качества проектирования. 7. Проектирование ПО: стратегии и методы, нотации проектирования. 8. Конструирование ПО: основы конструирования. 9. Конструирование ПО: управление конструированием, языки конструирования. 10. Конструирование ПО: кодирование, тестирование в конструировании, оценка качества, интеграция. 11. Тестирование ПО: основные концепции и определение тестирования, уровни тестирования. 12. Тестирование ПО: техники тестирования, метрики тестирования, управление тестированием. 13. Сопровождение ПО: основные концепции, категории сопровождения, ключевые вопросы сопровождения ПО. 14. Сопровождение ПО: процессы сопровождения, техники сопровождения. 15. Управление конфигурацией ПО: основные понятия, составляющие части. 16. Управление проектами ПО: организационное, задачи, инженерия измерений. 17. Процесс инженерии ПО: определения, составляющие части. 18. Методы инженерии ПО: классификация методов, категории методов. 19. Инструменты инженерии ПО: классификация, области применения. 20. Качество ПО: внешние и внутренние характеристики качества, основные и дополнительные, техники гарантии качества. 21. Стандарт ISO 12207. Основные определения. 22. Стандарт ISO 12207. Процессы жизненного цикла (ЖЦ). 23. Модель жизненного цикла разработки ПО как процесс (обобщенная схема, иерархия элементов). 24. Каскадная (водопадная) модель жизненного цикла ПП. 25. Итеративная и инкрементальная модель жизненного цикла ПП. 26. Спиральная модель жизненного цикла ПП. 27. V-образная модель жизненного цикла ПП. 28. Инкрементная (пошаговая) модель жизненного цикла ПП. 29. Модель быстрого прототипирования. 30. Модель жизненного цикла MSF. 31. Модель жизненного цикла RUP. 32. Модель жизненного цикла XP. 33. Управление командой проекта: ролевая модель команды. 34. Модели организации команд: административная модель,
5.2. Перечень работ, выполняемых по дисциплине (Курсовая работа, Курсовой проект, РГР, Реферат, ЛР, ПР и т.п.)			
Код работы	Название работы	Проверяемые индикаторы компетенций	Содержание работы

P1	Курсовая работа	УК-2-У1;ОПК-7-У1;УК-2-В1;ОПК-7-В1	<p>Примерная тематика курсовых работ</p> <ol style="list-style-type: none"> 1. Программное обеспечение банкомата. 2. Информационная система публичной библиотеки. 3. Информационная система поликлиники. 4. Информационная система деканата. 5. Система мгновенного обмена сообщениями. 6. Информационная система склада. 7. Система учета рабочего времени. 8. Информационная система жилищного агентства. 9. Информационная система технической экспертизы. 10. Система продажи билетов для проезда 11. Пакет программного обеспечения для регистратора в больнице 12. Программная система для call-центра банка 13. Организация и ведение спортивного чемпионата 14. Построение расписания занятий в ВУЗе 15. Автоматизация работы компании по аренде жилых и нежилых помещений 16. Автоматизация работы автосалона 17. Программа ведения личной библиотеки 18. Программа учета транспортных средств предприятия 19. Автоматизация отдела кадров предприятия 20. Автоматизация работы торгового представителя розничных продовольственных товаров
P2	Лабораторная работа 1	ОПК-7-В1;ОПК-7-У1;УК-2-В1;УК-2-У1	<p>Выбор темы проекта</p> <p>Можно в качестве темы использовать курсовую работу по объектно-ориентированному программированию и для нее выполнить этапы: бизнес-анализа, бизнес-моделирования, проектирования архитектуры, документирования указанных этапов.</p> <p>«Исходные данные к проекту» включает в себя следующие подразделы:</p> <p>Характеристики объекта автоматизации (или управления);</p> <p>Требования к информационному обеспечению.</p> <p>Требования к техническому обеспечению.</p> <p>Требования к программному обеспечению.</p> <p>Общие требования к проектируемой системе.</p> <p>Перечень дополнительных работ (если необходимо).</p> <p>Характеристики объекта автоматизации. Здесь указываются общие характеристики объекта автоматизации, характерные для рассматриваемой предметной области:</p> <p>полное название объекта (ов);</p> <p>условия его функционирования;</p> <p>количественные и качественные показатели объекта, которые являются ограничениями процесса функционирования.</p>

P3	Лабораторная работа 2	УК-2-У1;УК-2-В1;ОПК-7-У1;ОПК-7-В1	<p>ОПИСАНИЕ И АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ</p> <p>RAD включает в себя элементы методологии объектно-ориентированного проектирования и анализа предметной области. Для быстрой и эффективной разработки программной системы с минимальным браком требуется определить верное направление работы.</p> <p>Чтобы правильно построить систему, сначала необходимо построить ее модель.</p> <p>Принцип моделирования: «Лучшие модели - те, что ближе к реальности».</p> <p>Сначала нужно подробно изучить предметную область, для которой разрабатывается программа.</p> <p>В соответствии с методологией ООАП выделяются следующие шаги работы над проектом (системой).</p> <p>Описание предметной области: «Под предметной областью (application domain) принято понимать ту часть реального мира, которая имеет существенное значение или непосредственное отношение к процессу функционирования программы. Предметная область включает в себя только те объекты и взаимосвязи между ними, которые необходимы для описания требований и условий решения некоторой задачи. Необходимо выделить основные объекты (компоненты), участвующие в функционировании системы, определить их наиболее существенные характеристики, взаимосвязи в рамках решаемой задачи, а также определить основные информационные потоки в системе. Компоненты выбираются таким образом, чтобы при последующей разработке их было удобно представить в форме классов и объектов. Выбрать язык представления информации о концептуальной схеме предметной области.</p> <p>Сложность предметной области определяет количество объектов и связей между ними, поэтому описание должно включать в себя базовые термины и определения, сопровождаться различными примерами, в нем могут приводиться различного рода классификации, поясняющие различные свойства описываемых объектов. Если в системе используются математические модели, то они также должны быть описаны с учетом специфики применения.</p>
P4	Лабораторная работа 3	УК-2-У1;УК-2-В1;ОПК-7-У1;ОПК-7-В1	<p>ПОСТАНОВКА ЗАДАЧИ</p> <p>Постановка задачи – заключительный этап первой фазы ЖЦ системы. На данном этапе формулируются все требования, которым должна удовлетворять система. Постановка задачи пишется в повествовательной форме в будущем времени на основе ТЗ, в ней должны быть обязательно взаимоувязаны виды автоматизируемой деятельности (с привязкой к объекту(ам) автоматизации) со всеми ограничениями, накладываемыми на них, учтены особенности разрабатываемого информационного обеспечения и перечислены функции, которые должна выполнять система (с привязкой к процессам и информационному обеспечению).</p> <p>В системе также должна быть обеспечена возможность получения справочной информации как о самой системе, так и предоставляемых ею возможностях.</p> <p>Система должна выполнять следующие функции: (здесь перечисляются все функции, которые были определены в разделе 2.5.1 ТЗ)».</p> <p>5 см. раздел 2.2 ТЗ «Требования к информационному обеспечению системы»</p> <p>6 Последняя функция не обязательна, т.к. достаточно сложна при реализации.</p> <p>7 см. раздел 2.5.1 ТЗ «Функции, реализуемые системой».</p> <p>8 Эта часть постановки задачи обязательна.</p>

P5	Лабораторная работа 4	УК-2-У1;УК-2-В1;ОПК-7-У1;ОПК-7-В1	<p>РАЗРАБОТКА СТРУКТУРЫ СИСТЕМЫ</p> <p>Построение структурной схемы программной системы. На данном этапе система по функциональному признаку разделяется на основные подсистемы, между ними указываются информационные связи и/или связи по управлению, описывается основное назначение подсистем. При разработке структурной схемы используется методология структурного проектирования, в основе которой лежит алгоритмическая декомпозиция и иерархия вида «часть-целое», учитывающая, что внутренние связи элементов внутри подсистем сильнее, чем связь между подсистемами. Декомпозиция системы может повторяться многократно, вплоть до уровня конкретных процедур, при этом должна быть обеспечена целостность системы, а все составляющие компоненты взаимоувязаны. Для этого используются такие принципы разработки, как «сверху-вниз», «разделяй и властвуй», «иерархическое упорядочивание» и другие.</p> <p>Система (греч. - «составленное из частей», «соединение» от «соединяю») - множество элементов, находящихся в отношениях и связях друг с другом, которое образует определённую целостность, единство.</p> <p>Главным свойством системы является ее целостность: комплекс объектов, рассматриваемых в качестве системы, должен обладать общими свойствами и поведением.</p> <p>Необходимо рассматривать и связи системы с внешней средой. В самом общем случае понятие «система» характеризуется: наличием множества элементов; наличием связей между ними; целостным характером данного устройства или процесса.</p> <p>Система должна представлять собой совокупность элементов (объектов, субъектов), находящихся между собой в определенной зависимости и составляющих некоторое единство (целостность), направленное на достижение определенной цели. Система может являться элементом другой системы более высокого порядка (надсистема) и включать в себя системы более низкого порядка (подсистемы). То есть систему можно рассматривать как набор подсистем, организованных для достижения определенной цели и описанных с помощью набора моделей (возможно, с различных точек зрения), а подсистему – как группу элементов, часть которых составляет спецификацию поведения, представленного другими ее составляющими.</p> <p>К типовым можно отнести следующие подсистемы:</p> <ul style="list-style-type: none"> подсистему управления; подсистемы ввода-вывода; подсистему настройки параметров; файловую подсистему; подсистему визуализации; подсистему документирования; подсистему взаимодействия с базой данных; справочную подсистему. <p>Полученная в результате декомпозиции структура системы должна сопровождаться кратким описанием включенных в нее подсистем.</p> <p>В состав системы входят следующие подсистемы:</p> <ul style="list-style-type: none"> Подсистема управления, которая отвечает за взаимодействие подсистем между собой и представлена в виде иерархического меню; ... Файловая подсистема, которая ... Подсистема работы со словарем, которая ... Подсистема визуализации, которая ... Справочная подсистема, которая содержит сведения о системе (руководство пользователю) и ее об ее разработчиках.
----	-----------------------	-----------------------------------	--

5.3. Оценочные материалы, используемые для экзамена (билеты, тесты и т.п.)

«МИСИС»
НОВОТРОИЦКИЙ ФИЛИАЛ

Кафедра электроэнергетики и электротехники

БИЛЕТ К ЭКЗАМЕНУ № 0

Дисциплина: «Программная инженерия»
Направление: 09.03.03 «Прикладная информатика»
Форма обучения: очная
Форма проведения экзамена: устная

1. Жизненный цикл программного продукта. Процесс, действие, задача жизненного цикла. Фазы (этапы) жизненного цикла и их связь с процессами.
2. Тестирование программного обеспечения. Цели и задачи тестирования.

Составил доцент: _____ Р.Р.Абдулвелеева

Зав. кафедрой МиЕ: _____ А.В.Швалёва

«01» сентября 2026 г.

Дистанционно экзамен проводится в LMS Moodle. Экзаменационный тест содержит 30 заданий. На решение отводится 30 минут. Разрешенные попытки - две. Зачитывается наилучший результат.

Образец заданий для экзамена, проводимого дистанционно в LMS Moodle:

1. Программная инженерия – это
 - а) Совокупность инструментальных средств и методов, предназначенных для создания качественного программного обеспечения.
 - б) Совокупность инструментальных средств, предназначенных для создания качественного программного обеспечения.
 - в) Совокупность навыков, инструментальных средств и методов, предназначенных для создания качественного программного обеспечения.
 - г) Наука, изучающая построение программных систем
 - д) Правила проектирования систем со сложной архитектурой
2. Программная инженерия занимается
 - а) Вопросами оптимизации кода
 - б) Вопросами разработки новых алгоритмов обработки данных
 - в) Вопросами эффективной разработки программного обеспечения
 - г) Применением средств быстрой разработки программного обеспечения
 - д) Применением средств автоматизированного тестирования программного обеспечения
3. Стадии разработки программных систем, общие формы алгоритмов и схем, описывающих эти системы, регламентируются
 - а) Стандартами ЕСПД
 - б) Пунктами ТЗ
 - в) Никак не регламентируются
 - г) Эксплуатационными документами
 - д) Спецификацией ПС
4. Псевдокод представляет собой
 - а) Частично формализованный язык для представления описаний метода пошаговой детализации
 - б) Язык, использующий конструкции структурного программирования
 - в) Язык программирования высокого уровня
 - г) Язык с неформальными фрагментами на естественном языке для представления обобщенных операторов и условий
 - д) Формальная запись конструкций языка программирования Фортран
5. Укажите основные процессы жизненного цикла по ГОСТ Р ИСО/МЭК 12207-99. «Информационная технология. Процессы жизненного цикла программных средств»
 - а) Процесс заказа
 - б) Процесс документирования
 - в) Процесс разработки
 - г) Процесс управления

- д) Процесс сопровождения
6. Проблемы, решаемые конфигурационным управлением
- а) Работа в команде
- б) Одновременная модификация
- в) Ограниченное уведомление
- г) Управление пользователями
- д) Множество версий
7. Этапы последовательной разработки ("водопад")
- а) Снятие с эксплуатации
- б) Тестирование
- в) Анализ требований
- г) Проектирование
- д) Системный анализ
- е) Использование и сопровождение
8. Этапы итеративного цикла разработки
- а) Тестирование
- б) бизнес-моделирование
- в) Реализация
- г) Анализ и проектирование
- д) Требования
9. Порядок разработки программного модуля
- а) Программирование (кодирование) модуля
- б) Шлифовка текста модуля
- в) Изучение и проверка спецификации модуля, выбор языка программирования
- г) Выбор алгоритма и структуры данных
- д) Компиляция модуля
- е) Проверка модуля
10. Содержание технического задания на программный продукт в порядке следования
- а) Техничко-экономические показатели
- б) Назначение разработки
- в) Стадии и этапы разработки
- г) Введение
- д) Требования к программной документации
- е) Порядок контроля и приёмки
- ж) Основания для разработки
- з) Требования к программе или программному изделию
11. Чем определяется сложность ПО?
- а) количеством пользователей
- б) объемом обрабатываемых данных
- в) требованиями по быстродействию
12. В чем заключается согласованность ПО?
- а) в том, что ПО основывается на объективных посылках
- б) в том, что ПО должно быть согласовано с большим количеством интерфейсов
- в) в согласованности заказчика и исполнителя
13. К какому типу проектов относятся проекты по разработке ПО:
- а) и к творческим, и к промышленным проектам
- б) к промышленным проектам
- в) к творческим проектам
14. На каком уровне процессы в полной мере существуют лишь в рамках отдельных проектов:
- а) на начальном уровне
- б) на управляемом уровне
- в) на оптимизирующемся уровне
15. Какой из участников создания модели при описании системы не несет ответственности за качество моделирования:
- а) автор
- б) эксперт
- в) читатель

16. Какой вопрос решается в сфере программной инженерии:
- вопросы создания компьютерных программ и/или программного обеспечения
 - бизнес-реинжиниринг
 - вопрос поддержки жизненного цикла разработки ПО
17. Как называется процесс разбиения одной сложной задачи на несколько простых подзадач?
- абстракция
 - декомпозиция
 - реинжиниринг
 - верификация
18. Интерфейс пользователя — это
- набор методов взаимодействия компьютерной программы и пользователя этой программы
 - набор методов для взаимодействия между программами
 - способ взаимодействия между объектами
 - прежде всего, набор правил
19. Что из приведенного является критериями оценки удобства интерфейсов?
- скорость обучения
 - адаптация к стилю работы пользователя
 - абстракция
 - реинжиниринг
20. Легкость применения программного обеспечения это
- характеристики ПО, позволяющие минимизировать усилия пользователя по подготовке исходных данных, применению ПО
 - отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов
 - характеристики ПО, позволяющие минимизировать усилия по внесению изменений для устранения в нем ошибок и по его модификации
 - способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей
21. Мобильность программного обеспечения это
- способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей
 - способность ПО безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени
 - способность ПО быть перенесенным из одной среды (аппаратного / программного) в другое
 - отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов
22. Под отладкой программного средства понимают
- Деятельность, направленная на обнаружение и исправление ошибок в ПС с использованием процессов выполнения его программ
 - Процесс выполнения его программ на некотором наборе данных, для которого заранее известен результат применения или известны правила поведения этих программ
 - Отладка = Тестирование + Поиск ошибок + Редактирование
 - Процесс поиска и исправления ошибок (без тестирования)
 - Настройка ПС на требуемые наборы данных
23. Устойчивость программного обеспечения — это
- свойство, характеризующее способность ПС завершать автоматически корректное функционирование ПК, несмотря на неправильные (ошибочные) входные данные
 - свойство, способное противостоять преднамеренным или непреднамеренным деструктивным действиям пользователя
 - свойство, характеризующее способность ПС продолжать корректное функционирование, несмотря на неправильные (ошибочные) входные данные
 - отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов
24. UML — это:
- язык программирования, имеющий синтаксис схожий с C ++
 - унифицированный язык визуального моделирования, использует нотацию диаграмм
 - набор стандартов и спецификаций качества программного обеспечения
 - адрес в сети интернет
25. При конструировании программного обеспечения на этапе разработки или выбора алгоритма решения реализуется следующее

- а) архитектурные обработки программы
б) выбор языка программирования
в) совершенствование программы
г) синтаксические отладки
26. Проектирование ПО в основном рассматривается как
а) архитектурное проектирование
б) коммуникационные методы
в) детальные методы
г) совершенствование программы
27. На этапе тестирования пользователь выполняет следующее
а) синтаксические отладки
б) выбор тестов и метода тестирования
в) определение формы выдачи результатов
г) архитектурные обработки программы
28. Что из приведенного не является одним из методов проектирования программного обеспечения?
а) структурное программирование
б) объектно-ориентированное программирование
в) алгебраическое программирование
г) синтаксические отладки
29. В обсуждении требований на систему принимают участие:
а) аналитики и разработчики будущей системы
б) представители заказчика из нескольких профессиональных групп
в) специалисты, производящие установку системы
30. Спецификация требований к ПО — это:
а) процесс проверки правильности спецификации требований на их соответствие, непротиворечивость, полноту и выполнимость, а также на соответствие стандартам
б) формализованное описание функциональных, нефункциональных и системных требований, требований к характеристикам качества, а также к структуре ПО, принципам взаимодействия с другими компонентами, алгоритмам и структуре данных системы
в) проверка требований, для того чтобы убедиться, что они определяют именно данную систему
31. Объект предметной области — это:
а) образ с поведением, которое обусловлено его характеристиками и взаимоотношениями с другими объектами предметной области
б) конкретный образ с поведением, которое обусловлено его характеристиками и взаимоотношениями с другими объектами предметной области
в) значение некоторой абстрактной сущности предметной области
32. Реинженерия (reengineering) — это:
а) внесение изменений в компоненты или интерфейсы (добавление, расширение и т. д.), добавление экземпляров компонентов, новых функций или системных сервисов
б) эволюция программы путем ее изменения в целях повышения удобства ее эксплуатации, сопровождения или изменения ее функций
в) полная переделка компонентов, а иногда и перепрограммирование всей системы
33. Функциональный аудит конфигурации проводится:
а) для подтверждения информации о текущем статусе идентифицированных объектов конфигурационного управления, предложенных изменениях, а также о выявленных дефектах и отклонениях
б) для подтверждения взаимного соответствия документации и фактической конфигурации продукта
в) (Правильный ответ) для подтверждения соответствия фактических характеристик конфигурации продукта требованиям заказчика
34. После получения новой версии системы заказчику передаются:
а) версия
б) инструменты управления версиями для самостоятельного внесения изменений при сопровождении системы
в) документация
г) отчеты о выявленных ошибках
д) конфигурация
35. Какую роль выполняет менеджер в процессе работы над ошибками:
а) нахождение ошибок
б) контроль хода проекта
в) исправление ошибок

36. Какой из участников создания модели при описании системы не несет ответственности за качество моделирования:

- а) автор
- б) эксперт
- в) читатель

37. С какой ролью можно совмещать разработку:

- а) архитектура
- б) управление продуктом
- в) тестирование

38. На каком уровне зрелости осуществляется анализ причин возникновения проблем и предотвращение их появления в будущем:

- а) на уровне зрелости 3
- б) на уровне зрелости 4
- в) на уровне зрелости 5

5.4. Методика оценки освоения дисциплины (модуля, практики. НИР)

Критерии оценки ответов на экзамене, проводимом в устной форме

Оценка «Отлично» ставится, если

- на теоретические вопросы даны развернутые ответы, при необходимости изложен математический аппарат (формулы, графики и т.д.) приведены соответствующие схемы, таблицы, рисунки и т.д., правильно решена задача
- обучающийся хорошо ориентируется в материале, владеет терминологией, приводит примеры, обосновывает, анализирует, высказывает свою точку зрения по анализируемым явлениям и процессам, правильно применяет полученные знания при решении практических задач. Ответы излагаются свободно, уверенно без использования листа устного опроса

Оценка «Хорошо» ставится, если

- на теоретические вопросы даны полные ответы, но имела место неточность в определении каких-либо понятий, явлений и т.д. Задача решена.

- обучающийся ориентируется в материале хорошо, но допускает ошибки при формулировке, описании отдельных категорий

Оценка «Удовлетворительно» ставится, если

- на теоретические вопросы даны общие неполные ответы
- обучающийся слабо ориентируется в материале, не может решать задачи, не может привести пример, не может анализировать и обосновывать

Оценка «Неудовлетворительно» ставится, если

- не решена задача и правильный ответ дан на один вопрос (либо ни на один)
- обучающийся в материале дисциплины практически не ориентируется, т.е. не может дать даже общих сведений по вопросу.

Критерии оценки ответов на экзамене, проводимом в дистанционной форме в LMS Moodle

$90 \leq$ Процент верных ответов ≤ 100 - отлично

$75 \leq$ Процент верных ответов < 90 - хорошо

$60 \leq$ Процент верных ответов < 75 – удовлетворительно

Критерии оценки выполнения курсовой работы:

1. Теоретические сведения изложены в достаточном объеме, четко и последовательно
2. Используются выводы (позиции, мнения и др.) известных ученых, профессионалов
3. Исследуются и сравниваются разные подходы, методики, приводятся собственные суждения и выводы
4. Описана актуальность работы и предметная область.
5. Описывается процесс анализа и моделирования предметной области
6. Описываются алгоритмы работы и интерфейс программы
7. Приведен процесс тестирования ПО
8. Текст написан грамотно, стилистически выдержан
9. Текст оформлен в соответствии с требованиями

Работа оценивается на отлично, если:

теоретические сведения изложены в достаточном объеме, четко и последовательно, использованы выводы (позиции, мнения и др.) известных ученых, профессионалов, исследуются и сравниваются разные подходы, методики, приводятся собственные суждения и выводы, имеются примеры, даются ссылки на источники, текст написан грамотно, стилистически выдержан и оформлен в соответствии с требованиями.

Процесс анализа и моделирования предметной области описан полностью, обязательно должен включать в себя требования к ПО; архитектуру ПО, включая три модели: информационную, состояний и процессов; спецификацию ПО; схему БД.

В полном объеме описываются алгоритмы работы и интерфейс программы, приведены все диалоговые окна с подробным описанием функционала каждого элемента интерфейса.

Приведен процесс тестирования ПО, который включает в себя: стратегию тестирования; тест-план; отчет по проведенному тестированию; результаты выполнения автоматизированного теста.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение работы оценивается как хорошее, если она соответствует всем критериям, перечисленным выше, но в работе отсутствуют некоторые элементы описания процесса анализа и моделирования предметной области, отсутствует полное описание функционала каждого элемента интерфейса, описание тестирования приведено не полностью.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение работы оценивается как удовлетворительное, если она соответствует всем критериям, перечисленным выше, но в работе отсутствуют некоторые элементы описания процесса анализа и моделирования предметной области, отсутствует описание функционала каждого элемента интерфейса, описание тестирования не приведено. Отсутствует описание актуальности работы и предметной области.

Если работа допущена до защиты с оценкой «отлично», в процессе защиты студент хорошо владеет материалом, не использует при этом опорных конспектов и т.д., с легкостью отвечает на любой вопрос по курсовой работе, то в этом случае студенту за выполнение курсовой работы ставится оценка «отлично», которая и проставляется в зачетную книжку и в ведомость.

В процессе защиты оценка повышаться не может, т.е. если студент допущен до защиты с оценкой «хорошо», «отлично» он уже в любом случае не сможет получить, а вот «удовлетворительно» может – если при защите возникают определенные трудности с ориентацией в материале, ответами на вопросы по курсовой работе.

Если студент совершенно не владеет материалом курсовой работы, то получает «неудовлетворительно».

Если работа не соответствует критериям выполнения курсовой работы, то оценивается неудовлетворительно и до защиты не допускается.

Критерии оценки выполнения расчетно-графической работы:

1. Теоретические сведения изложены в достаточном объеме, четко и последовательно
2. Исследуются и сравниваются разные подходы, методики, приводятся собственные суждения и выводы
3. Приведены основные цели разработки
4. Приведены требования к программному продукту
5. Определены сроки и этапы разработки
6. Регламентирован процесс приемосдаточных мероприятий.
7. Расставлены ссылки на источники
8. Текст написан грамотно, стилистически выдержан
9. Текст оформлен в соответствии с требованиями

Работа оценивается на отлично, если:

- теоретические сведения изложены в достаточном объеме, четко и последовательно, приводятся собственные суждения и выводы, даются ссылки на источники, текст написан грамотно, стилистически выдержан и оформлен в соответствии с требованиями.

- требования к программному продукту приведены в полном объеме, верно определены сроки и этапы разработки, полностью регламентирован процесс приемосдаточных мероприятий. Верно определены основные цели разработки.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение работы оценивается как хорошее, если она соответствует всем критериям, перечисленным выше, но требования к программному продукту, процесс приемосдаточных мероприятий приведены не полностью.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение работы оценивается как удовлетворительное, если она соответствует всем критериям, перечисленным выше, но требования к программному продукту приведены не полностью, отсутствует описание процесса приемосдаточных мероприятий, отсутствуют основные цели разработки.

Если расчетно-графическая работа не соответствует критериям, перечисленным выше, то оценивается неудовлетворительно.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л1.1	Лаврищева Е.М.	Программная инженерия. Парадигмы, технологии и CASE-средства: учебник		М.: Юрайт, 2019
Л1.2	Антамошкин О.А.	Программная инженерия. Теория и практика: учебник		Красноярск : Сибирский федеральный университет, 2012

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л2.1	Абдулаев В.И.	Программная инженерия: учебное пособие		Йошкар-Ола : ПГТУ, 2016

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л2.2	Флоренсов А.Н.	Системное программное обеспечение: учебное пособие		Омск : Издательство ОмГТУ, 2017

6.1.3. Методические разработки

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л3.1	А.С.Измайлова	Методические указания по выполнению курсовых работ: Методические указания		НФ НИТУ «МИСиС», 2020
Л3.2	В.Т. Николаев, С.В. Купцов, В.Н. Тикменов	Практика программирования в инженерных расчётах: учебное пособие		Москва : Физматлит, 2018
Л3.3	Э.Р. Ипатов, Ю.В. Ипатов	Методологии и технологии системного проектирования информационных систем: учебник		Москва : Издательство «Флинта», 2016
Л3.4	Долженко, А.И.	Технологии командной разработки программного обеспечения информационных систем		Москва : Национальный Открытый Университет «ИНТУИТ», 2016

6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Э1	Научная электронная библиотека eLIBRARY	https://www.elibrary.ru/
Э2	LMS Moodle	https://lms.misis.ru
Э3	НФ НИТУ МИСиС	http://nf.misis.ru/
Э4	Университетская библиотека ONLINE	https://biblioclub.ru/

6.3 Перечень программного обеспечения

П.1	WinPro 10 RUSUpgrdOLVNLEachAcdmcAP
П.2	Microsoft Office 2010 Russian Academic OPEN 1 License No Level
П.3	Microsoft Office 2007 Russian Academic OPEN No Level
П.4	Браузер Google Chrome
П.5	Microsoft Teams
П.6	Zoom
П.7	Браузер Yandex

6.4. Перечень информационных справочных систем и профессиональных баз данных

И.1	https://www.computer.org/education/bodies-of-knowledge/software-engineering - Свод знаний по программной инженерии (SWEBOOK)
И.2	http://www.firststeps.ru/java/java1.html - Разработка приложений на Java
И.3	http://www.gpntb.ru - Государственная публичная научно-техническая библиотека

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Ауд.	Назначение	Вид	Оснащение
123	Учебная лаборатория (компьютерный класс) Кабинет курсового и дипломного проектирования, самостоятельной работы обучающихся	Ср	14 шт. - Системный блок; 14 шт. - Монитор LCD LG21,5; 1 шт. - Экран настенный 200x200; 1 шт. - Проектор ACER X118DLP 3600; 1 шт. - Подвес для проектора; 1 шт. - Коммутатор D-Link; 1 шт. - Доска ученическая; 27 шт. - Столы ученические; 52 шт. - Стулья; 4 шт. - Жалюзи.

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

Освоение дисциплины предполагает как проведение традиционных аудиторных занятий, так и работу в электронной информационно-образовательной среде НИТУ «МИСиС» (ЭИОС), частью которой непосредственно предназначенной для осуществления образовательного процесса является Электронный образовательный ресурс LMS Moodle. Он доступен по URL адресу <https://lms.misis.ru> и позволяет использовать специальный контент и элементы электронного обучения и дистанционных образовательных технологий. LMS Moodle используется преимущественно для асинхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет».

Чтобы эффективно использовать возможности LMS Moodle а соответственно и успешно освоить дисциплину, нужно:

- 1) зарегистрироваться на курс. Для этого нужно перейти по ссылке ... Логин и пароль совпадает с логином и паролем от личного кабинета НИТУ МИСиС;
- 2) в рубрике «В начало» ознакомиться с содержанием курса, вопросами для самостоятельной подготовки, условиями допуска к аттестации, формой промежуточной аттестации (зачет/экзамен), критериями оценивания и др.;
- 3) в рубрике «Модули», заходя в соответствующие разделы изучать учебные материалы, размещенные преподавателем. В т.ч. пользоваться литературой, рекомендованной преподавателем, переходя по ссылкам;
- 4) в рубрике «Библиотека» возможно подбирать для выполнения письменных работ (контрольные, домашние работы, курсовые работы/проекты) литературу, размещенную в ЭБС НИТУ «МИСиС»;
- 5) в рубрике «Задания» нужно ознакомиться с содержанием задания к письменной работе, сроками сдачи, критериями оценки. В установленные сроки выполнить работу(ы), подгрузить здесь же для проверки. Удобно называть файл работы следующим образом (название предмета (сокращенно), группа, ФИО, дата актуализации (при повторном размещении)). Например, Экономика_Иванов_И.И._БМТ-19_20.04.2020. Если работа содержит рисунки, формулы, то с целью сохранения форматирования ее нужно подгружать в pdf формате.

Работа, подгружаемая для проверки, должна:

- содержать все структурные элементы: титульный лист, введение, основную часть, заключение, список источников, приложения (при необходимости);

- быть оформлена в соответствии с требованиями.

Преподаватель в течение установленного срока (не более десяти дней) проверяет работу и размещает в комментариях к заданию рецензию. В ней он указывает как положительные стороны работы, так замечания. При наличии в рецензии замечаний и рекомендаций, нужно внести поправки в работу, подгрузить ее заново для повторной проверки. При этом важно следить за сроками, в течение которых должно быть выполнено задание. При нарушении сроков, указанных преподавателем возможность подгрузить работу остается, но система выводит сообщение о нарушении сроков. По окончании семестра подгрузить работу не получится;

6) в рубрике «Тесты» пройти тестовые задания, освоив соответствующий материал, размещенный в рубрике «Модули»;

7) в рубрике «Оценки» отслеживать свою успеваемость;

8) в рубрике «Объявления» читать объявления, размещаемые преподавателем, давать обратную связь;

9) в рубрике «Обсуждения» создавать обсуждения и участвовать в них (обсуждаются общие моменты, вызывающие вопросы у большинства группы). Данная рубрика также может быть использована для взаимной проверки;

10) проявлять регулярную активность на курсе.

Преимущественно для синхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет» используется Microsoft Teams (MS Teams). Чтобы полноценно использовать его возможности нужно установить приложение MS Teams на персональный компьютер и телефон. Старостам нужно создать группу в MS Teams.

Участие в группе позволяет:

- слушать лекции;

- работать на практических занятиях;

- быть на связи с преподавателем, задавая ему вопросы или отвечая на его вопросы в общем чате группы в рабочее время с 9.00 до 17.00;

- осуществлять совместную работу над документами (вкладка «Файлы»).

При проведении занятий в дистанционном синхронном формате нужно всегда работать с включенной камерой.

Исключение – если преподаватель попросит отключить камеры и микрофоны в связи с большими помехами. На аватарках должны быть исключительно деловые фото.

При проведении лекционно-практических занятий ведется запись. Это дает возможность просмотра занятия в случае невозможности присутствия на нем или при необходимости вновь обратиться к материалу и заново его просмотреть.