

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Котова Лариса Анатольевна
Должность: Директор филиала
Дата подписания: 26.09.2023 12:35:24
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

Рабочая программа дисциплины (модуля)

Программная инженерия

Закреплена за подразделением Кафедра математики и естествознания (Новотроицкий филиал)

Направление подготовки 09.03.03 Прикладная информатика

Профиль Прикладная информатика в технических системах

Квалификация	Бакалавр	
Форма обучения	заочная	
Общая трудоемкость	12 ЗЕТ	
Часов по учебному плану	432	Формы контроля на курсах: экзамен 3 курсовая работа 3
в том числе:		
аудиторные занятия	72	
самостоятельная работа	342	
часов на контроль	18	

Распределение часов дисциплины по курсам

Курс	3		Итого	
	уп	рп		
Лекции	20	20	20	20
Лабораторные	20	20	20	20
Практические	32	32	32	32
Итого ауд.	72	72	72	72
Контактная работа	72	72	72	72
Сам. работа	342	342	342	342
Часы на контроль	18	18	18	18
Итого	432	432	432	432

Программу составил(и):

к.п.н, Доцент, Абдулвелеева Р.Р.

Рабочая программа

Программная инженерия

Разработана в соответствии с ОС ВО:

Самостоятельно устанавливаемый образовательный стандарт высшего образования - бакалавриат Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский технологический университет «МИСИС» по направлению подготовки 09.03.03 Прикладная информатика (приказ от 05.03.2020 г. № 95 о.в.)

Составлена на основании учебного плана:

09.03.03 Прикладная информатика, 09.03.03_22_Прикладная информатика_ПрПИВТС_заоч.rlx Прикладная информатика в технических системах, утвержденного Ученым советом ФГАОУ ВО НИТУ "МИСиС" в составе соответствующей ОПОП ВО 30.11.2021, протокол № 35

Утверждена в составе ОПОП ВО:

09.03.03 Прикладная информатика, Прикладная информатика в технических системах, утвержденной Ученым советом ФГАОУ ВО НИТУ "МИСиС" 30.11.2021, протокол № 35

Рабочая программа одобрена на заседании

Кафедра математики и естествознания (Новотроицкий филиал)

Протокол от 24.06.2021 г., №11

Руководитель подразделения Швалёва А.В.

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	Информационные системы и технологии	
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:	
2.2.1	Проектирование информационных систем	
2.2.2	Проектирование систем SCADA	
2.2.3	Экономика	
2.2.4	Дизайн web-приложений	
2.2.5	Подготовка к процедуре защиты и защита выпускной квалификационной работы	
2.2.6	Преддипломная практика	
2.2.7	Разработка UI/UX	
2.2.8	Средства информатизации в металлургии	
2.2.9	Средства информатизации в энергетике	

3. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ФОРМИРУЕМЫМИ КОМПЕТЕНЦИЯМИ

ПК-2: Способен выполнять проектные работы по созданию, модификации (интегрированию программных модулей) и сопровождению ИС, формулировать требования к ИС
Знать:
ПК-2-31 основ и технологии разработки программ
ОПК-8: Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла, демонстрировать практические навыки для решения задач и реализации проектов, в области, соответствующей профилю подготовки, применять знание экономических, организационных и управленческих вопросов, таких как: управление проектами, рисками и изменениями
Знать:
ОПК-8-31 основные технологии создания и внедрения информационных систем, стандарты управления жизненным циклом информационной системы
ПК-2: Способен выполнять проектные работы по созданию, модификации (интегрированию программных модулей) и сопровождению ИС, формулировать требования к ИС
Уметь:
ПК-2-У1 формулировать требования к ИС
ОПК-8: Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла, демонстрировать практические навыки для решения задач и реализации проектов, в области, соответствующей профилю подготовки, применять знание экономических, организационных и управленческих вопросов, таких как: управление проектами, рисками и изменениями
Уметь:
ОПК-8-У1 осуществлять организационное обеспечение выполнения работ на всех стадиях и в процессах жизненного цикла информационной системы
Владеть:
ОПК-8-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 программной инженерии. Жизненный цикл ПО.							
1.1	Управление процессом разработки программного обеспечения. Технологии программирования. Обзор технологий программирования (структурное, модульное, объектно-ориентированное, компонентное программирование). Составляющие стоимости ПО. Программный процесс и модель программного процесса. Методы программной инженерии. Технология, стандарт и сертификация. Роль стандартов в 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/	3	8	ОПК-8-31 ПК-2-У1	Л1.1 Л1.2Л2.1 Э1 Э2 Э3 Э4			
1.2	Самостоятельное изучение учебного материала в LMS Canvas: Программная инженерия и ее отличия от информатики и других инженерий. /Ср/	3	46	ОПК-8-31	Л1.1 Л1.2Л2.1 Э1 Э2 Э3 Э4			
1.3	Жизненный цикл программного продукта /Пр/	3	6	ОПК-8-31				
	Раздел 2. Анализ предметной области и требований к ПО							
2.1	Анализ предметной области и требования к ПО. Фиксация требований к ПО. Составление спецификации требований. Разработка компонентов модели данных приложения. Выдача заданий для контрольной работы. /Пр/	3	8	ПК-2-31	Л1.1 Л1.2Л2.1 Л2.2 Л2.3 Л2.4 Э1 Э2 Э3 Э4			

2.2	<p>Моделирование потребности заказчика. Методы выявления требований. Процесс анализа предметной области. Разработка модели системы в шаблоне «ввод-обработка-вывод».</p> <p>Принципы анализа: информационная область, моделирование, разделение на части, ракурсы видения основной информации и деталей реализации.</p> <p>Элементы модели анализа. Моделирование данных: объекты, свойства и связи данных, словарь данных, диаграммы связей между объектами. /Лек/</p>	3	2	ОПК-8-31	Л1.1 Л1.2Л2.1 Л2.3 Л2.4 Э1 Э2 Э3 Э4			
2.3	<p>Самостоятельное изучение учебного материала в LMS Canvas: 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/</p>	3	56	ОПК-8-У1	Л1.1 Л1.2Л2.1 Л2.3 Л2.4 Э1 Э2 Э3 Э4			
2.4	Проведение экзамена /Экзамен/	3	8	ОПК-8-В1	Э1 Э2 Э3 Э4			
2.5	Подготовка к экзамену /Ср/	3	36	ОПК-8-31				
Раздел 3. Архитектура ПО								
3.1	<p>Самостоятельное изучение учебного материала в LMS Canvas: 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/</p>	3	56	ОПК-8-У1	Л1.1 Л1.2Л2.1 Л2.3 Л2.4 Э1 Э2 Э3 Э4			

3.2	<p>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. Задачи и особенности объектно-ориентированного проектирования программных средств. Основные понятия и модели объектно-ориентированного проектирования программных средств. Варианты представления моделей и средства объектно-ориентированного проектирования программных средств. /Лек/</p>	3	4	ОПК-8-31	Л1.1 Л1.2Л2.1 Л2.3 Л2.4 Э1 Э2 Э3 Э4			
3.3	<p>Разработка внутренней структуры приложений при помощи диаграмм. Методологии разработки ПО. Проектирование и создание базы данных приложения. /Лаб/</p>	3	16	ОПК-8-У1	Л1.1 Л1.2Л2.1 Л2.2 Л2.3 Л2.4Л3.2 Э1 Э2 Э3 Э4			
3.4	<p>Принципы создания удобного пользовательского интерфейса. Определение архитектуры пользовательского интерфейса приложения. /Пр/</p>	3	12	ОПК-8-31	Л1.1 Л1.2Л2.1 Л2.2 Л2.3 Л2.4 Э1 Э2 Э3 Э4			
	<p>Раздел 4. Качество ПО. Сопровождение ПО.</p>							
4.1	<p>Стандартизация качества. Методы обеспечения качества ПО. Понятие тестирования. Инструменты тестирования. Критерии тестирования. Виды тестирования. Принципы верификации и тестирования программ. Процессы оценивания характеристик и испытания программных средств. Организация и методы сопровождения программных средств. Этапы и процедуры при сопровождении программных средств. Задачи и процессы переноса программ и данных на иные платформы. /Лек/</p>	3	6	ОПК-8-31	Л1.1 Л1.2Л2.1 Л2.4 Э1 Э2 Э3 Э4			

4.2	Методы верификации и тестирования программ и систем. Разработка системных тестов для приложения. Выдача задания для курсовой работы. Выявление факторов, определяющих потребность в сопровождении программного обеспечения. Выявление категорий программного обеспечения, нуждающегося в сопровождении. /Пр/	3	6	ОПК-8-31	Л1.1 Л1.2Л2.1 Л2.2 Л2.4 Э1 Э2 Э3 Э4			
4.3	Использование отладчиков в процессе тестирования программных средств. Тестирование операций ввода/вывода. Составление заявок предложений о модификации и поиски возможности их удовлетворения. Организация работ по сопровождению информационных систем. Оценка трудоемкости и стоимости проекта. /Лаб/	3	4	ОПК-8-У1	Л1.1 Л1.2Л2.1 Л2.2 Л2.4Л3.2 Э1 Э2 Э3 Э4			
4.4	Самостоятельная работа с источниками литературы и ЛМС /Ср/	3	56	ОПК-8-В1				
Раздел 5. Проектная деятельность в разработке ПО								
5.1	Самостоятельное изучение учебного материала в LMS Canvas: Области управленческих знаний. Компетенции менеджера IT проекта. Роль и способы общения в команде. Преимущества и недостатки различных способов общения. Корпоративная политика. Типы внешних стратегий команд. СММ. Основные понятия модели технологической зрелости. СММ. Пять уровней зрелости модели СММ и их характеристика. Подготовка к защите курсовой работы. Подготовка к экзамену. /Ср/	3	38	ОПК-8-31 ПК-2-31	Л1.1 Л1.2Л2.1Л3.1 Э1 Э2 Э3 Э4			
5.2	Проведение экзамена /Экзамен/	3	10	ОПК-8-31 ПК-2-31	Э1 Э2 Э3 Э4			
5.3	Выполнение курсовой работы /Ср/	3	54	ОПК-8-У1 ОПК-8-В1	Э1			

5. ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ

5.1. Вопросы для самостоятельной подготовки к экзамену (зачёту с оценкой)

Вопросы к экзамену (ПК-1-31, ПК-1-32, ПК-1-У1, ПК-1-У2, ПК-3-31, ПК-3-32, ПК-3-У1, ОПК-7-31, ОПК-7-У1, ОПК-8-31, ОПК-8-У1, УК-6-31, УК-6-32, УК-6-У1):

1. Рынок программного обеспечения. Сложность управления процессом разработки программного обеспечения.
2. Технологии программирования (структурное, модульное, объектно-ориентированное, компонентное программирование).
3. Программный продукт и его основные характеристики. Составляющие стоимости ПО.
4. Программная инженерия и ее отличия от информатики и других инженерий.
5. Программный процесс и модель программного процесса. Методы программной инженерии. CASE системы.
6. Профессиональные и этические требования ИТ-специалиста. Кодекс этики IEEE-CS/ACM.
7. Технология, стандарт и сертификация. Роль стандартов в программной инженерии.
8. Основные стандарты программной инженерии и кто их разрабатывает?
9. Жизненный цикл программного продукта. Процесс, действие, задача жизненного цикла. Фазы (этапы) жизненного цикла и их связь с процессами.
10. Основные процессы жизненного цикла ПО (ISO12207 и ISO 15504).
11. Вспомогательные процессы жизненного цикла ПО (ISO12207 и ISO 15504).
12. Организационные процессы жизненного цикла ПО (ISO12207 и ISO 15504).
13. Каскадная модель ЖЦ ПО. Преимущества, недостатки, применимость.
14. Спиральная модель ЖЦ ПО. Преимущества, недостатки, применимость.
15. Обзор других типов моделей ЖЦ ПО.
16. Особенности моделей жизненного цикла MSF, RUP, XP.
17. Процесс анализа предметной области. Область анализа: повторное использование.
18. Методы идентификации потребностей. Моделирование потребности заказчика.
19. Модели архитектуры программной системы: стили, шаблоны.
20. Модели анализа требований. Словарь данных и другие модели информационной области.
21. Роль модели «карта диалогов» и её отличие от Диаграммы перехода состояний.
22. Специфицирование: принципы специфицирования, способы и форма представления спецификаций.
23. Спецификация требований к программному обеспечению. Назначение и содержание документа.
24. Выполнение структурного анализа: создание модели потока данных, спецификация процессов.
25. Виды архитектуры ПО.
26. Понятие и определение архитектуры ПО.
27. Основные этапы архитектурного проектирования ПО.
28. Модели архитектуры на этапе структурирования ПО.
29. Модели архитектуры на этапе определения принципов управления работой ПО.
30. Модели архитектуры на этапе модульной композиции ПО.
31. Назначение языка UML.
32. Классификация диаграмм UML.
33. Тестирование программного обеспечения. Цели и задачи тестирования.
34. Тестирование методом «черного ящика» и методом прозрачного ящика. Особенности, достоинства, недостатки методов.
35. Проектирование тестов для "межклассовых" испытаний.
36. Отличие функционального тестирования от сборочного. Функциональное тестирование программного обеспечения, тестирование программной (вычислительной) системы.
37. Методы верификации программ и систем.
38. Что такое проект и его основные характеристики. Непроекты и их связь с проектами.
39. Управление и управление проектами. Категории управления проектами.
40. Особенности управления ИТ-проектами. Треугольник ограничений проекта.
41. Области управленческих знаний.
42. Компетенции менеджера IT проекта.
43. Ролевая модель команды. Роли и их ответственности.
44. Модель управления командой. Критерии выбора модели.
45. Административная модель, модель хаоса, модель открытой архитектуры. Особенности, преимущества и недостатки.
46. Роль и способы общения в команде. Преимущества и недостатки различных способов общения.
47. Корпоративная политика. Типы внешних стратегий команд.
48. Управление проектом разработки программного обеспечения. Концепция. Формула Барии Боэма. Риски.
49. Управление проектом разработки программного обеспечения. Планирование. Диаграмма Ганта. Критический путь.
50. Средства управления проектом. Функции систем управления проектом. Обзор систем управления проектами.
51. Оценка трудоемкости программного проекта. Методы.

5.2. Перечень работ, выполняемых по дисциплине (модулю, практике, НИР) - эссе, рефераты, практические и расчетно-графические работы, курсовые работы, проекты и др.

Курсовая работа «Разработка программного обеспечения» (ПК-1-В1, ПК-3-В1, ОПК-7-В1, ОПК-8-В1, УК-6-В1, УК-6-В2). Целью выполнения курсовой работы является закрепление у обучающихся и проверка со стороны преподавателя навыков использования современных подходов к программированию, основанных на современных требованиях информационных технологий.

Задачами выполнения курсовой работы являются:

- научиться применять полученные знания при решении реальной задачи;
- развить навыки самостоятельной работы и самостоятельного принятия решений.

Курсовая работа заключается в создании собственной программы с помощью современных средств программирования, анализа функциональности, надежности и тестирования разработанной программы и полного документирования всех этапов ее разработки. По итогам выполнения курсовой работы оформляется пояснительная записка, в которой описывается весь процесс разработки программного обеспечения.

Исходные данные: примерные предметные области, для которых можно разработать программный продукт, примерные типы программного обеспечения.

Примерный перечень предметных областей:

1. Аэропорт;
2. Фитнес-центр;
3. Поликлиника;
4. Ателье;
5. Гостиница;
6. Библиотека;
7. Магазин;
8. Прокат спортивного инвентаря;
9. Агентство недвижимости;
10. Станция технического обслуживания.

Примерный перечень типов программного обеспечения:

1. Система документооборота предприятия.
2. Система учета (материалов, энергоносителей, кадров на предприятии).
3. Информационно-справочная система в какой-либо области.
4. Информационная система предприятия.
5. Система поддержки принятия решений на предприятии. Объем курсовой работы – 30-35 стр.

Пояснительная записка курсовой работы должна содержать следующие основные структурные элементы:

1. Титульный лист
2. Задание к курсовой работе
3. Содержание
4. Введение, в котором описана актуальность работы и описание предметной области
5. Первый раздел, в котором описывается процесс анализа и моделирования предметной области (обязательно должен включать в себя требования к ПО; архитектуру ПО, включая три модели: информационную, состояний и процессов; спецификацию ПО; схему БД)
6. Второй раздел, в котором описываются алгоритмы работы и интерфейс программы (необходимо привести все диалоговые окна с подробным описанием функционала каждого элемента интерфейса)
7. Третий раздел, в котором документируется процесс тестирования ПО (обязательно должен включать в себя: стратегию тестирования; тест-план; отчет по проведенному тестированию; результаты выполнения автоматизированного теста)
8. Заключение
9. Приложение, содержащее листинг основных модулей программы и автоматизированного теста

Контрольная работа включает в себя расчетно-графическую на тему "Разработка технического задания" (ОПК-6-В1, ОПК-6-В2, ПК-1-В1, ПК-1-В2, ПК-3-В1).

Техническое задание представляет собой документ, в котором сформулированы основные цели разработки, требования к программному продукту, определены сроки и этапы разработки и регламентирован процесс приемосдаточных испытаний. Разработать техническое задание для программы, обеспечивающей:

1. работу регистратуры поликлиники.
2. учет книг в библиотеке.
3. учет запчастей в автосервисе.
4. учет успеваемости студентов.
5. планирование рабочего дня руководителя предприятия.
6. учет клиентской базы оптового магазина.
7. учет товаров продуктового магазина.
8. регистрацию результатов спортивного соревнования.
9. учет изменения погодных условий в населенном пункте.
10. анализ туристических предпочтений.

Техническое задание должно содержать следующие разделы:

1. Титульный лист;
2. Задание;
3. Содержание;

4. Введение;
5. Наименование и область применения;
6. Основание для разработки;
7. Назначение разработки;
8. Технические требования к программе или программному изделию;
9. Стадии и этапы разработки;
10. Порядок контроля и приемки;
11. Заключение;
12. Список использованных источников;
13. Приложения (при необходимости).

5.3. Оценочные материалы, используемые для экзамена (описание билетов, тестов и т.п.)

Формой промежуточной аттестации по дисциплине является экзамен.
Ниже представлен образец билета для экзамена, проводимого в устной форме.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ
«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.Е. Мажирина

Зав. кафедрой ЭиЭ: _____ Р.Е. Мажирина

«01» сентября 2020 г.

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

Образец заданий для экзамена, проводимого дистанционно в LMS Canvas:

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 посылаках
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:

- а) вопросы создания компьютерных программ и/или программного обеспечения
- б) бизнес-реинжиниринг
- в) вопрос поддержки жизненного цикла разработки ПО

17. Как называется процесс разбиения одной сложной задачи на несколько простых подзадач?

- а) абстракция
- б) декомпозиция
- в) реинжиниринг
- г) верификация

18. Интерфейс пользователя — это

- а) набор методов взаимодействия компьютерной программы и пользователя этой программы
- б) набор методов для взаимодействия между программами
- в) способ взаимодействия между объектами
- г) прежде всего, набор правил

19. Что из приведенного является критериями оценки удобства интерфейсов?

- а) скорость обучения
- б) адаптация к стилю работы пользователя
- в) абстракция
- г) реинжиниринг

20. Легкость применения программного обеспечения это

- а) характеристики ПО, позволяющие минимизировать усилия пользователя по подготовке исходных данных, применению ПО
- б) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов
- г) характеристики ПО, позволяющие минимизировать усилия по внесению изменений для устранения в нем ошибок и по его модификации
- д) способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей

21. Мобильность программного обеспечения это

- а) способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей
- б) способность ПО безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени
- в) способность ПО быть перенесенным из одной среды (аппаратного / программного) в другое
- г) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов

22. Под отладкой программного средства понимают

- а) Деятельность, направленная на обнаружение и исправление ошибок в ПС с использованием процессов выполнения его программ
- б) Процесс выполнения его программ на некотором наборе данных, для которого заранее известен результат применения или известны правила поведения этих программ
- в) Отладка = Тестирование + Поиск ошибок + Редактирование
- г) Процесс поиска и исправления ошибок (без тестирования)
- д) Настройка ПС на требуемые наборы данных

23. Устойчивость программного обеспечения — это

- а) свойство, характеризующее способность ПС завершать автоматически корректное функционирование ПК, несмотря на неправильные (ошибочные) входные данные

- б) свойство, способное противостоять преднамеренным или непреднамеренным деструктивным действиям пользователя
- в) свойство, характеризующее способность ПС продолжать корректное функционирование, несмотря на неправильные (ошибочные) входные данные
- г) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов
24. UML — это:
- а) язык программирования, имеющий синтаксис схожий с C ++
- б) унифицированный язык визуального моделирования, использует нотацию диаграмм
- в) набор стандартов и спецификаций качества программного обеспечения
- г) 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 инсталляцию 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 Canvas

$90 \leq$ 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 приведены не полностью, отсутствует описание процесса приемосдаточных мероприятий.

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, эл. адрес
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, http://biblioclub.ru/index.php?page=book&id=363975

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л2.1	Абдулаев В.И.	Программная инженерия: учебное пособие		Йошкар-Ола : ПГТУ, 2016, http://biblioclub.ru/index.php?page=book&id=459449
Л2.2	В.Т. Николаев, С.В. Купцов, В.Н. Тикменов	Практика программирования в инженерных расчётах: учебное пособие		Москва : Физматлит, 2018, http://biblioclub.ru/index.php?page=book&id=485295
Л2.3	Э.Р. Ипатов, Ю.В. Ипатов	Методологии и технологии системного проектирования информационных систем: учебник		Москва : Издательство «Флинта», 2016, http://biblioclub.ru/index.php?page=book&id=79551
Л2.4	Флоренсов А.Н.	Системное программное обеспечение: учебное пособие		Омск : Издательство ОмГТУ, 2017, http://biblioclub.ru/index.php?page=book&id=493301

6.1.3. Методические разработки

	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л3.1	А.С.Измайлова	Методические указания по выполнению курсовых работ: Методические указания		НФ НИТУ «МИСиС», 2020, www.nf.misis.ru
Л3.2	Долженко, А.И.	Технологии командной разработки программного обеспечения информационных систем		Москва : Национальный Открытый Университет «ИНТУИТ», 2016, http://biblioclub.ru/index.php?page=book&id=428801

6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

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

6.3 Перечень программного обеспечения**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		

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 Canvas. Он доступен по URL адресу <https://lms.misis.ru> и позволяет использовать специальный контент и элементы электронного обучения и дистанционных образовательных технологий. LMS Canvas используется преимущественно для асинхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет».

Чтобы эффективно использовать возможности LMS Canvas, 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.