

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Курганский государственный университет»
(КГУ)

Кафедра «Программное обеспечение автоматизированных систем»



УТВЕРЖДАЮ:
Первый проректор
Т.Р. Змызгова
31 августа 2023 г.

Рабочая программа учебной дисциплины

ТЕХНОЛОГИИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ

образовательной программы высшего образования –
программы специалитета

10.05.03 – Информационная безопасность автоматизированных систем

Специализация №5:

Безопасность открытых информационных систем

очная форма обучения

Курган 2023

Программа дисциплины «Технологии и методы программирования» составлена в соответствии с учебным планом программы специалитета 10.05.03 – Информационная безопасность автоматизированных систем (*Безопасность открытых информационных систем*), утвержденным 30.06.2023 года.

Рабочая программа дисциплины одобрена на заседании кафедры Программного обеспечения автоматизированных систем 30.08.2023 года, протокол № 1.

Рабочую программу разработал
доцент кафедры ПОАС



В.К. Волк

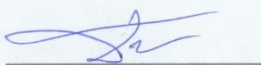
Согласовано:

Заведующий
кафедрой ПОАС



В.К. Волк

Заведующий
кафедрой БИАС



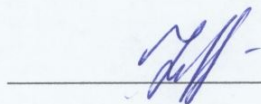
Д.И. Дик

Начальник
Управления
образовательной деятельности



И.В. Григоренко

Специалист
по учебно-методической работе
Учебно-методического отдела



Г.В. Казанкова

СОДЕРЖАНИЕ

1. ОБЪЕМ ДИСЦИПЛИНЫ	4
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ.....	4
3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ.....	4
3.1. Цели и задачи изучения дисциплины.....	4
3.2. Компетенции, формируемые в результате освоения дисциплины:.....	5
3.3. Результаты обучения.....	5
4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ.....	6
4.1 Учебно-тематический план	6
4.2 Содержание лекционных занятий.....	6
4.3 Лабораторный практикум.....	9
5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ СТУДЕНТОВ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ	12
5.1 Курс лекций	12
5.2 Лабораторный практикум.....	12
5.3 Самостоятельная работа	12
6. ПРОЦЕДУРА ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ	13
6.1 Состав и формы проведения контрольно-аттестационных мероприятий	13
Текущий контроль.....	14
Рубежный контроль	14
Промежуточная аттестация	14
6.2 Система балльно-рейтинговой оценки работы студентов.....	14
6.3 Критерии допуска к промежуточной аттестации	15
6.4 Фонд оценочных средств.....	16
Перечень оценочных средств.....	16
Примерные варианты компонентов фонда оценочных средств	16
7 ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА	19
7.1. Основная литература	19
7.2. Дополнительная литература.....	19
8 УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ	19
9. РЕСУРСЫ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ	19
10 ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ИНФОРМАЦИОННО-СПРАВОЧНЫЕ СИСТЕМЫ	22
11 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ	22
12 Для студентов, обучающихся с использованием ДОТ	22

1. ОБЪЕМ ДИСЦИПЛИНЫ

Виды учебной работы	Распределение трудоемкости по семестрам и видам учебных занятий, акад. часов	
	Всего	4-й семестр
Общая трудоемкость: 4 зач. ед.	144	144
Аудиторные занятия, все-	64	64
Лекции	32	32
Лабораторные работы	32	32
Самостоятельная работа, всего:	80	80
Подготовка к экзамену	27	27
Другие виды самостоятельной работы	53	53
Виды промежуточной аттестации	Экзамен	Экзамен

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина «Технологии и методы программирования» включена в модуль «Информатика и программирование» обязательной части образовательной программы, для ее освоения необходимы компетенции, сформированные при изучении дисциплин «Информатика», «Основы программирования», «Основы информационной безопасности».

Результаты изучения дисциплины необходимы для освоения технологий проектирования информационных систем различного назначения и используются при освоении дисциплин «Безопасность систем баз данных», «Методы проектирования защищенных распределенных информационных систем», «Технология построения защищенных информационных приложений».

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

3.1. Цели и задачи изучения дисциплины

Цель изучения дисциплины – ознакомление со стандартами программной инженерии и технологиями промышленной разработки программного обеспечения (далее – ПО).

Задачи дисциплины:

изучение:

- основных понятий, методологических основ и стандартов программной инженерии;
- структуры процессов жизненного цикла программного продукта;
- основных моделей жизненного цикла программного продукта.

практическое освоение:

- основ языка визуального моделирования (UML), используемого при анализе и проектировании ПО;
- CASE-средств поддержки программных проектов.

3.2. Компетенции, формируемые в результате освоения дисциплины:

- способность создавать программы на языках общего назначения, применять методы и инструментальные средства программирования для решения профессиональных задач, осуществлять обоснованный выбор инструментария программирования и способов организации программ (ОПК-7);
- разрабатывать компоненты систем защиты информации автоматизированных систем (ОПК-11);
- организовывать и проводить диагностику и тестирование систем защиты информации автоматизированных систем, проводить анализ уязвимостей систем защиты информации автоматизированных систем (ОПК-13)

3.3. Результаты обучения

В процессе освоения дисциплины студент должен демонстрировать следующие **результаты обучения**:

Должен знать:

- языки, системы и инструментальные средства, используемые в процессе проектирования программного обеспечения (для ОПК-7);
- стандарты и технологии промышленной разработки программного обеспечения и типовые проектные решения по обеспечению безопасности автоматизированных систем (для ОПК-11, ОПК-13);

Должен уметь:

- осваивать новые образцы программных, технических средств и информационных технологий (для ОПК-7);
- применять языки, системы и инструментальные средства проектирования программного обеспечения (для ОПК-7, ОПК-11);
- разрабатывать и анализировать проектные решения по обеспечению безопасности автоматизированных систем (для ПК-11);

Должен владеть практическими навыками применения инструментальных средств, используемых для документирования программных проектов (для ОПК-7);

4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1 Учебно-тематический план

Разделы дисциплины		Часов контактной работы с преподавателем	
№	Наименование	Лекции	Лаб. работы
1	Программная инженерия: технологии промышленной разработки ПО	4	0
2	Стандарты программной инженерии	14	16
Рубежный контроль №1. Онлайн тестирование.		2	0
3	Визуальное моделирование при анализе и проектировании программных систем	12	12
Рубежный контроль №2. Защита учебного программного проекта		0	4
Всего часов аудиторных занятий по дисциплине:		32	32

4.2 Содержание лекционных занятий

Наименование и содержание лекции	Часов контактной работы с преподавателем
Раздел №1. ПРЕДМЕТ И БАЗОВЫЕ ПОНЯТИЯ ПРОГРАММНОЙ ИНЖЕНЕРИИ	
Лекция 1.1. Введение. Программная инженерия: базовые понятия Цели и задачи дисциплины, рабочая программа и учебно-методические материалы. История развития технологий создания программного обеспечения (ПО): модульное программирование, структурный и объектно-ориентированный подходы к программированию и проектированию. Понятие инженерной деятельности. Предмет программной инженерии.	2
Лекция 1.2. Жизненный цикл ПО Понятие и основные стадии жизненного цикла (ЖЦ) промышленного изделия. Особенности ЖЦ ПО. Понятие модели ЖЦ, типы моделей ЖЦ ПО (каскадная модель, V-модель, эволюционные модели, спиральная модель) и области их эффективного применения.	2
Раздел №2. СТАНДАРТЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ	
Лекция 2.1. Стандартизация в программной инженерии. Стадии создания автоматизированных систем (АС). Роль стандартизации в инженерной деятельности. Категории стандартов и организации - разработчики стандартов программной инженерии. Стадии создания АС.	2
Лекция 2.2. Стандарты документирования программных проектов – ГОСТ Р 59795-2021. Требования к содержанию документов. Единая система программной документации (ЕСПД): – ГОСТ 19.101. Виды программ и программных документов. – ГОСТ 19.102. Стадии разработки. – ГОСТ 19.103. Обозначения. – ГОСТ 19.105. Общие требования. – ГОСТ 19.201. Техническое задание. – ГОСТ 19.202. Спецификация. – ГОСТ 19.301. Программа и методика испытаний. – ГОСТ 19.401. Текст программы. – ГОСТ 19.402. Описание программы.	2

Наименование и содержание лекции	Часов контактной работы с преподавателем
<p align="center">Лекция 2.3. Стадия «Техническое задание»</p> <p>Управление требованиями. Классификация требований. Требования к защищенности и классы защищенности автоматизированных систем.</p>	2
<p align="center">Лекция 2.4. Процессы жизненного цикла ПО</p> <p>Структура процессов жизненного цикла ПО (стандарт ISO 12207). Классификация и иерархия процессов ЖЦ ПО: основные, вспомогательные и организационные процессы. Участники процессов ЖЦ ПО. Основные процессы ЖЦ ПО: "заказ", "поставка", "разработка", "эксплуатация", "сопровождение". Типовая структура процесса "разработка": подготовка процесса; анализ требований к системе; проектирование системной архитектуры; анализ требований к программным модулям; проектирование программной архитектуры; техническое проектирование программных модулей; программирование, тестирование, сборка и квалификационные испытания программных модулей; сборка системы; квалификационные испытания системы; ввод в действие и обеспечение приемки.</p>	2
<p align="center">Лекция 2.5. Управление программными проектами</p> <p>PM BOK. Управление конфигурациями. Управление качеством программного проекта. Два направления проектного менеджмента: CMMI Process Improvement и Agile Software Development. Методология MSF for Agile Software Development: концепции, элементы, модель проектной группы (ролевые группы и роли). Методология Scrum: принципы Scrum-разработки; команда проекта; Scrum-встречи и Sprint-встречи</p>	2
<p align="center">Лекция 2.6. Профессиональные и образовательные стандарты</p> <p>Обзор российских и зарубежных профессиональных стандартов программной инженерии. Образовательные стандарты: SWE BOK. Обзор российских образовательных стандартов УГС 09 (Информатика и вычислительная техника) и УГС 10 (Информационная безопасность).</p>	2
<p align="center">Лекция 2.7. Профессиональные стандарты информационной безопасности</p> <p>Российские профессиональные стандарты:</p> <ul style="list-style-type: none"> – 06.030 Специалист по защите информации в ТКС и сетях; – 06.032 Специалист по безопасности КС и сетей; – 06.033 Специалист по защите информации в АС; – 06.034 Специалист по технической защите информации; – 06.053 Специалист по информационной безопасности в кредитно-финансовой сфере; – 06.044 Консультант в области развития цифровой грамотности населения. <p>Зарубежные профессиональные стандарты:</p> <ul style="list-style-type: none"> – ISCO-08:2529.4 – Этичный хакер – ISCO-08:2529.6 – Администратор безопасности ИКТ – ISCO-08:2529.7 – Оперативник по кибер-безопасности 	4

Наименование и содержание лекции	Часов контактной работы с преподавателем
Раздел №3. ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ ПРИ АНАЛИЗЕ И ПРОЕКТИРОВАНИИ ПРОГРАММНЫХ СИСТЕМ	
<p>Лекция 3.1. <i>Проектирование как процесс информационного моделирования объекта</i></p> <p>Проектирование как процесс преобразования информационных моделей объекта. Задачи и базовые принципы проектирования сложных объектов (абстрагирование, декомпозиция, многоэтапность, итерационность). Визуализация при проектировании. Обзор систем графического моделирования. Структурные модели анализа бизнес-процессов: схема Захмана. Диаграммы структурного анализа систем: SADT - диаграммы функционального моделирования; ERD - диаграммы "Сущность – Связь"; DFD - диаграммы потоков данных. ООАП - объектно-ориентированный подход к анализу и проектированию программных систем. Язык графического моделирования UML: история стандартизации UML; структура и базовые понятия языка; модели и UML-диаграммы.</p>	2
<p>Лекция 3.2. <i>Пример документирования программного проекта</i></p> <p><i>Стадия технического задания (ТЗ):</i> терминологический словарь, базовые функциональные требования, обобщенная Use Case-диаграмма. <i>Стадия эскизного проекта (ЭП):</i> структурная декомпозиция (UML-диаграмма пакетов); функциональная декомпозиция (локальные Use Case-диаграммы и сценарии).</p> <p><i>Стадия технического проекта (ТП):</i> статическая модель проектируемой системы (структурная декомпозиция пакетов, UML-диаграммы классов); динамическая модель поведения проектируемой системы (UML-диаграммы состояний).</p>	2
<p>Лекция 3.3. <i>Разработка UML-диаграмм и сценариев вариантов использования</i></p> <p>Use Case-модель как результат функциональной декомпозиции проектируемой системы на начальной стадии программного проекта. Назначение и область применения Use-Case-моделей. Компоненты модели и графическая нотация. Сценарии вариантов использования. Примеры UseCase-диаграмм.</p>	2
<p>Лекция 3.4. <i>Разработка UML-диаграмм пакетов и UML-диаграмм классов</i></p> <p>Статические модели концептуального и логического уровней. «Пакеты» и «Классы» – два уровня структурной декомпозиции проектируемой системы. Компоненты моделей (пакеты, классы, интерфейсы, отношения), графическая нотация. Примеры диаграмм.</p>	2
<p>Лекция 3.5. <i>Разработка UML-диаграмм активности и UML-диаграмм состояний</i></p> <p>Назначение и область применения динамических UML-моделей. Модели логического уровня. Компоненты модели: простые и составные состояния; события и переходы; триггеры. Обозначения и графическая нотация. Примеры диаграмм.</p>	4
Всего часов лекционных занятий по дисциплине	32

4.3 Лабораторный практикум

Наименование и содержание лабораторной работы	Часов контактной работы с преподавателем
Раздел №2. СТАНДАРТЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ	
<p style="text-align: center;">Лабораторная работа № 2.1</p> <p style="text-align: center;">Стадии создания автоматизированной системы (АС)</p> <p>ГОСТ Р 59793: Стадии создания АС. ГОСТ 34.602: Техническое задание на разработку АС. Руководящий документ ФСТЭК. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации.</p>	2
<p style="text-align: center;">Лабораторная работа № 2.2</p> <p style="text-align: center;">Стандарты документирования программных проектов</p> <ul style="list-style-type: none"> – ГОСТ Р 59795-2021. Автоматизированные системы. Требования к содержанию документов. – Единая система программной документации (ЕСПД): <ul style="list-style-type: none"> ○ ГОСТ 19.101. Виды программ и программных документов. ○ ГОСТ 19.102. Стадии разработки. Требования к содержанию и оформлению документов: <ul style="list-style-type: none"> ○ ГОСТ 19.103. Обозначения. ○ ГОСТ 19.105. Общие требования. ○ ГОСТ 19.201 Техническое задание. ○ ГОСТ 19.202. Спецификация. ○ ГОСТ 19.301 Программа и методика испытаний. ○ ГОСТ 19.401 Текст программы. ○ ГОСТ 19.402 Описание программы. 	2
<p style="text-align: center;">Лабораторная работа № 2.3</p> <p style="text-align: center;">Процессы жизненного цикла ПО</p> <p>ГОСТ Р ИСО/МЭК 12207. Информационная технология. Процессы жизненного цикла программных средств.</p> <ul style="list-style-type: none"> – Процессы реализации ПС: анализ требований, проектирование архитектуры, детальное проектирование и конструирование, квалификационное тестирование. – Процессы поддержки ПС: управление конфигурацией; управление качеством; верификация и валидация; ревизия и аудит. 	2
<p style="text-align: center;">Лабораторная работа № 2.4</p> <p style="text-align: center;">Управление качеством программного проекта</p> <ul style="list-style-type: none"> – Стандарт СММ SW (Capability Maturity Model for Software). Оценка технологической зрелости процессов и организаций. – ГОСТ Р ИСО/МЭК 25010. Модели качества систем и ПО. – ГОСТ Р ИСО/МЭК 15504. Информационные технологии. Оценка процессов (Часть 2. Базовая модель процессов и их зрелости). 	1
<p style="text-align: center;">Лабораторная работа № 2.5.</p> <p style="text-align: center;">Профессиональные стандарты IT-отрасли. Разработка ПО</p> <ul style="list-style-type: none"> – 06.022 Системный аналитик – 06.003 Архитектор программного обеспечения – 06.001 Программист – 06.035 Разработчик Web и мультимедийных приложений 	1

Наименование и содержание лабораторной работы	Часов контактной работы с преподавателем
<ul style="list-style-type: none"> – 06.025 Специалист по дизайну графических пользовательских интерфейсов – 06.004 Специалист по тестированию в области ИТ – 06.019 Технический писатель – 06.028 Системный программист – 06.041 Специалист по интеграции прикладных решений <p>ИТ-профессии и квалификационные уровни, состав трудовых функций, требования к образовательным уровням ИТ-специалистов.</p>	
<p style="text-align: center;">Лабораторная работа № 2.6. Профессиональные стандарты ИТ-отрасли. Руководство и управление проектами</p> <ul style="list-style-type: none"> – 06.016 Руководитель проектов в области ИТ – 06.017 Руководитель разработки программного обеспечения – 06.014 Менеджер по ИТ – 06.040 Специалист по контролю качества ИКС и сервисов – 06.012 Менеджер продуктов в области ИТ – 06.013 Специалист по информационным ресурсам – 06.029 Менеджер по продажам ИКС <p>ИТ-профессии и квалификационные уровни, состав трудовых функций, требования к образовательным уровням ИТ-специалистов.</p>	2
<p style="text-align: center;">Лабораторная работа № 2.7. Профессиональные стандарты ИТ-отрасли. Анализ данных</p> <ul style="list-style-type: none"> – 06.031 Специалист по автоматизации информационно-аналитической деятельности – 06.042 Специалист по большим данным – 06.043 Специалист по интернет-маркетингу – 06.046 Специалист по моделированию, сбору и анализу данных цифрового следа <p>ИТ-профессии и квалификационные уровни, состав трудовых функций, требования к образовательным уровням ИТ-специалистов.</p>	2
<p style="text-align: center;">Лабораторная работа № 2.8. Профессиональные стандарты ИТ-отрасли. Администрирование и информационная безопасность</p> <ul style="list-style-type: none"> – 06.011 Администратор баз данных – 06.015 Специалист по информационным системам – 06.026 Системный администратор ИКС – 06.027 Специалист по администрированию сетевых устройств – 06.030 Специалист по защите информации в ТКС и сетях – 06.032 Специалист по безопасности КС и сетей – 06.033 Специалист по защите информации в АС – 06.034 Специалист по ТЗИ – 06.053 Специалист по ИБ в кредитно-финансовой сфере – 06.044 Консультант в области развития цифровой грамотности <p>ИТ-профессии и квалификационные уровни, состав трудовых функций, требования к образовательным уровням ИТ-специалистов.</p>	2

Наименование и содержание лабораторной работы	Часов контактной работы с преподавателем
<p style="text-align: center;">Лабораторная работа № 2.9.</p> <p style="text-align: center;">Международные профессиональные стандарты IT-отрасли. ESCO - European Skills, Competences, Qualifications and Occupations ISCO - International Standard Classification of Occupations Группа ISCO-08:25 Специалисты-профессионалы по информационно-коммуникационным технологиям</p> <ul style="list-style-type: none"> – ISCO-08:251 – Разработчики и аналитики ПО <ul style="list-style-type: none"> ○ ISCO-08:2511 – Системные аналитики ○ ISCO-08:2512 – Разработчики ПО ○ ISCO-08:2513 – Разработчики Web- и мультимедийных приложений ○ ISCO-08:2514 – Программисты приложений ○ ISCO-08:2519 – Прочие разработчики и аналитики ПО: <ul style="list-style-type: none"> ▪ ISCO-08:2519.1 – Специалист по качеству данных ▪ ISCO-08:2519.3 – Менеджер по изменениям и конфигурации ИКТ ▪ ISCO-08:2519.4 – Аналитик по восстановлению после сбоев в ИКТ ▪ ISCO-08:2519.5 – Менеджер по обеспечению качества ИКТ ▪ ISCO-08:2519.7 – Тестировщик ПО – ISCO-08:252 – Специалисты по базам данных и сетям <ul style="list-style-type: none"> ○ ISCO-08:2521 – Дизайнеры баз данных и администраторы ○ ISCO-08:2522 – Системные администраторы ○ ISCO-08:2523 – Специалисты по компьютерным сетям ○ ISCO-08:2529 – Прочие специалисты по базам данных и компьютерным сетям: <ul style="list-style-type: none"> ▪ ISCO-08:2529.4 – Этичный хакер ▪ ISCO-08:2529.6 – Администратор безопасности ИКТ ▪ ISCO-08:2529.7 – Оперативник по кибербезопасности ▪ ISCO-08:2529.9 – Инженер знаний <p>ИТ-профессии и квалификационные уровни, состав трудовых функций, требования к образовательным уровням ИТ-специалистов.</p>	2
Раздел №3. ВИЗУАЛЬНОЕ МОДЕЛИРОВАНИЕ ПРИ АНАЛИЗЕ И ПРОЕКТИРОВАНИИ ПРОГРАММНЫХ СИСТЕМ	
<p style="text-align: center;">Лабораторная работа № 3.1.</p> <p style="text-align: center;">Стадия ЭП. Разработка UseCase-моделей и сценариев вариантов использования</p> <p>Выполнение индивидуальных практических заданий.</p>	4
<p style="text-align: center;">Лабораторная работа № 3.2.</p> <p style="text-align: center;">Стадия ТП. Разработка UML-диаграмм классов</p> <p>Выполнение индивидуальных практических заданий.</p>	4
<p style="text-align: center;">Лабораторная работа № 3.4.</p> <p style="text-align: center;">Стадия ТП. Разработка UML-диаграмм состояний</p> <p>Выполнение индивидуальных практических заданий.</p>	4
<p>Рубежный контроль №2 Защита учебного программного проекта.</p>	4
<p>Всего часов лабораторных занятий по дисциплине</p>	32.

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ СТУДЕНТОВ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

5.1 Курс лекций

Содержание лекционного материала представлено в учебном пособии [2], структура которого соответствует тематическому плану изучения дисциплины. Учебное пособие содержит перечни контрольных вопросов, ответы на которые должны быть получены студентами в процессе самостоятельной проработки материала соответствующей лекции.

5.2 Лабораторный практикум

Основная цель лабораторного практикума – ознакомление со стандартами программной инженерии и освоение технологий объектно-ориентированного проектирования программных систем и документирования программных проектов с использованием CASE-средств, поддерживающих язык UML.

Лабораторные работы выполняются в соответствии с индивидуальными практическими заданиями, по результатам выполнения каждого задания оформляется отчет. Количество и состав заданий каждой лабораторной работы определяет преподаватель. Защита отчетов производится в форме публичных сообщений на аудиторных лабораторных занятиях.

По разделу №2 («Стандарты программной инженерии») предусмотрено 9 лабораторных работ аналитического характера, выполнение которых предполагает изучение и анализ определенных разделов соответствующих стандартов согласно полученным индивидуальным заданиям.

По разделу №3 («Визуальное моделирование при анализе и проектировании программных систем») предусмотрено 3 лабораторные работы проектного характера: выполнение каждой из работ требует реализации одной из стадий программного проекта согласно полученному индивидуальному заданию и предусматривает разработку соответствующей UML-диаграммы.

5.3 Самостоятельная работа

Самостоятельная работа студентов по освоению дисциплины включает:

- проработку лекционного материала и проведение экспресс-тестирования;
- выполнение лабораторных работ;
- подготовку и проведение рубежного контроля №1 в формате тестирования по материалу разделов №1 и №2 дисциплины;
- выполнение и подготовку к защите контрольной работы по материалу раздела №3;
- подготовку к промежуточной аттестации (зачету) по дисциплине.

Рекомендуемое распределение трудоемкости самостоятельной работы приведено в таблице 5.1.

Таблица 5.1 – Рекомендуемая трудоемкость самостоятельной работы

Виды самостоятельной работы	Рекомендуемая трудоемкость, акад. часов
Изучение материала лекционного курса и проведение экспресс-тестирования	16
Предмет и базовые понятия программной инженерии	4
Стандарты программной инженерии	8
Язык UML	4
Подготовка и выполнение лабораторных работ (по 2 часа на работу)	24
Подготовка к рубежному контролю	13
Рубежный контроль №1 (тестирование)	3
Рубежный контроль №2 (защита проекта)	10
Подготовка и проведение экзамена	27
Всего:	80

6. ПРОЦЕДУРА ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДИСЦИПЛИНЫ

6.1 Состав и формы проведения контрольно-аттестационных мероприятий

Программой изучения дисциплины предусмотрены мероприятия текущего и рубежного контроля и промежуточная аттестация (в форме экзамена).

График и формы проведения контрольных и аттестационных мероприятий приведены в таблице 6.1.

Таблица 6.1 - График проведения контрольных и аттестационных мероприятий

Виды	Содержание	Форма проведения	Неделя
Текущий контроль	Контроль освоения материала лекций	Тестирование	1 – 8
	Контроль выполнения лабораторных работ	Собеседование	2 – 16
Рубежный контроль	№1. Базовые понятия и стандарты программной инженерии.	Тестирование	9
	№2. Защита проекта	Публичный доклад с обсуждением	16
Промежуточная аттестация	Экзамен по дисциплине	Тестирование	17

Текущий контроль

Контроль освоения лекционного материала проводится в формате экспресс-тестирования. Тестирование проводится по каждой из 8 лекций, для допуска к тестированию по материалу очередной лекции студент должен проработать материал этой лекции и успешно пройти тестирование по материалам предыдущих лекций.

Контроль выполнения лабораторных работ проводится на аудиторных занятиях в форме демонстрации и последующей защиты отчетов по выполненным индивидуальным заданиям.

Рубежный контроль

Рубежный контроль №1 проводится в форме тестирования по тематическим разделам №1 и №2. К тестированию допускаются студенты, успешно прошедшие текущий контроль освоения материала лекций №1.1 – №2.4 и выполнившие лабораторные работы №2.1 – №2.9.

Рубежный контроль №2 проводится в форме публичной защиты результатов выполнения учебного программного проекта. К рубежному контролю допускаются студенты, успешно прошедшие текущий контроль освоения материала лекций №3.1 и №3.2 и выполнившие лабораторные работы №3.1 – №3.4.

Промежуточная аттестация

Экзамен по дисциплине проводится в форме тестирования по всем разделам дисциплины. Тест содержит 30 вопросов, расчетное время проведения тестирования – 45 минут.

6.2 Система балльно-рейтинговой оценки работы студентов

Оценивание результатов выполнения студентами очной формы обучения плановых контрольных и аттестационных мероприятий по дисциплине производится в соответствии с Положением о балльно-рейтинговой системе контроля о оценки академической активности студентов ФГБОУ ВО «Курганский государственный университет». Оценивание производится по 100-балльной шкале с последующим приведением итоговой 100-балльной рейтинговой оценки к традиционной четырех-балльной (таблица 6.3).

Рейтинговая оценка студента по дисциплине получается путем суммирования баллов, полученных им в течение семестра по результатам текущего и рубежного контроля (максимум 70 баллов) и баллов, полученных на промежуточной аттестации (максимум 30 баллов).

Максимальные балльные оценки по результатам проведения контрольных и аттестационных мероприятий (для студентов очной формы обучения) приведены в таблице 6.2. Минимальное количество баллов, которыми может быть оценен удовлетворительный ответ студента на промежуточной аттестации, равно 11. Неудовлетворительный ответ оценивается в 0 баллов.

Пересчет 100-балльной рейтинговой оценки студента по дисциплине в традиционную (4-балльную) оценку и в оценку ECTS (Общеввропейская система учета учебной работы) производится в соответствии с таблицей 6.3.

Таблица 6.2 – Рейтинговые балльные оценки по дисциплине

Виды контроля /аттестации по дисциплине	Содержание	Максимальная оценка	
		За одну аттестацию	Всего
Текущий контроль	Контроль результатов освоения материала лекций (16 лекций)	0,5	8
	Контроль результатов выполнения лабораторных работ (12 работ)	4	48
Рубежный контроль	№1. Базовые понятия и стандарты программной инженерии.	6	6
	№2. Защита учебного программного проекта	8	8
Промежуточная аттестация (экзамен)		30	30
Максимальная итоговая оценка, баллов			100

Таблица 6.3 – Соответствие шкал оценивания

Рейтинговая оценка, баллов	Виды оценок промежуточной аттестации	
	Традиционная оценка	Оценка ECTS
91-100	5 (отлично)	A
84-90		B
74-83	4 (хорошо)	C
68-73		D
61-67		E
51-60	3 (удовлетворительно)	Fx
0-50		F
	2 (неудовлетворительно)	Не зачтено

6.3 Критерии допуска к промежуточной аттестации

Для допуска к экзамену студент должен набрать по итогам текущего и рубежного контроля в течение семестра не менее 51 балла.

Для получения оценки без проведения процедуры промежуточной аттестации студент должен набрать не менее 61 балла по результатам текущего и рубежного контроля в течение семестра, при этом он вправе повысить свою рейтинговую балльную оценку по дисциплине путем сдачи экзамена.

По усмотрению преподавателя рейтинговая балльная оценка студента по дисциплине может быть повышена за счет получения им дополнительных баллов за академическую активность и выполнение дополнительных заданий по дисциплине, а также за участие в течение семестра в учебной, научно-исследовательской, спортивной, культурно-творческой и общественной деятельности университета (не более 30 баллов).

Ликвидация академических задолженностей, возникших из-за разности в учебных планах при переводе или восстановлении, проводится путем выполнения дополнительных индивидуальных заданий. Количество и состав дополнительных заданий, формы их выполнения и требования к результатам определяются преподавателем и доводятся до студента в момент выдачи заданий.

6.4 Фонд оценочных средств

Перечень оценочных средств

Учебно-методический комплекс дисциплины содержит следующие компоненты фонда оценочных средств:

текущий контроль:

- задания для экспресс-тестирования по материалу курса лекций;
- индивидуальные задания и образцы отчетов по выполнению лабораторных работ;

рубежный контроль:

- задания для тестирования по разделам дисциплины «Базовые понятия программной инженерии» и «Стандарты программной инженерии»;
- индивидуальные задания по выполнению учебного программного проекта;

промежуточная аттестация:

- вопросы для подготовки к экзамену;
- задания для контрольного тестирования.

Примерные варианты компонентов фонда оценочных средств

Текущий контроль:

- Перечислите основные черты инженерной деятельности.
- В чем принципиальное отличие программной инженерии от традиционных инженерных отраслей?
- Какие преимущества дает модульный принцип построения ПО и какие проблемы должны быть решены для его реализации?
- Перечислите основные черты структурного подхода к проектированию программных систем; какие из проблем создания и эксплуатации сложных программных комплексов решаются методами структурного проектирования?
- Перечислите основные черты структурного программирования;
- Какие из проблем разработки ПО решаются методами структурного программирования?
- Перечислите и дайте определения базовым принципам объектно-ориентированной методологии к разработке ПО;
- Какие из проблем создания и эксплуатации сложных программных комплексов решаются методами объектно-ориентированного проектирования и программирования?
- Что такое CASE?
- Приведите примеры Upper CASE, Middle CASE и Low CASE.
- Перечислите основные стадии жизненного цикла программного продукта.
- Перечислите типовые стадии создания автоматизированной системы согласно ГОСТ Р 5979 и ГОСТ 19.102.

- Перечислите основные виды работ, выполняемых на стадии технического задания согласно ГОСТ 34.602 и ГОСТ 19.201
- Дайте определения следующим терминам:
 - квалификация;
 - верификация;
 - валидация.
- Дайте определения и приведите примеры использования следующих компонентов UseCase-модели языка UML:
 - actor;
 - use-case;
 - interface;
 - association;
 - generalization;
 - include;
 - extend;
 - dependency.
- Дайте определения и приведите примеры использования следующих компонентов State-Machine-модели языка UML:
 - событие, переход, триггерный переход, альтернативный переход, ветвление, слияние;
 - состояние, составное состояние, параллельное состояние, последовательное состояние, историческое состояние, глубокое историческое состояние.

Рубежный контроль:

- какие характеристики продукта определены в стандарте ГОСТ Р ИСО 25010-2015 для «модели качества при использовании»?
- какие термины модели качества продукта соответствуют следующим определениям стандарта ГОСТ 3 ИСО 25010-2015:
 - связь точности и полноты достижения пользователями целей с израсходованными ресурсами;
 - способность двух или более продуктов обмениваться информацией и использовать такую информацию;
 - степень выполнения продуктом определенных функций при указанных условиях в течение установленного времени.
- Перечислите основные трудовые функции (по квалификационным уровням) IT-специалистов, определенные в следующих профессиональных стандартах РФ:
 - 06.030 Специалист по защите информации в ТКС и сетях
 - 06.032 Специалист по безопасности КС и сетей
 - 06.033 Специалист по защите информации в АС
 - 06.034 Специалист по ТЗИ
 - 06.053 Специалист по ИБ в кредитно-финансовой сфере
- Разработайте обобщенную UseCase-диаграмму для проекта автоматизированной системы:

- оперативный учет выдачи-возврата книг читателям библиотеки и анализ читательского спроса;
- кадровый учет и управление персоналом малого предприятия;
- оперативный учет и анализ продаж интернет-магазина.

Промежуточная аттестация:

- Перечислите стадии разработки ПО и приведите основное содержание каждой из них.
- Прокомментируйте понятия: "*жизненный цикл ПО*" и "*модель жизненного цикла ПО*". Перечислите известные Вам модели ЖЦ ПО.
- Опишите основные черты и области эффективного применения моделей ЖЦ ПО:
 - каскадной модели и V-модели;
 - эволюционной модели;
 - модели пошаговой разработки;
 - спиральной модели.
- Для чего используют модели сложных систем при их проектировании? Перечислите основные задачи, решаемые проектировщиками систем с помощью моделирования.
- Какие цели преследует визуальное моделирование систем?
- Расшифруйте сокращенные названия диаграмм *SADT*, *ERD*, *DFD* и переведите на русский язык. Для чего используются диаграммы перечисленных типов?
- Какие задачи позволяет решать методология *SADT*? Для чего используются элементы модели *activity* и *arrow*? Что такое *ICOM*? Опишите *SADT*-диаграммами процесс выдачи книг читателю абонента публичной библиотеки (на двух уровнях декомпозиции).
- Перечислите компоненты ER-модели, дайте определения всем компонентам. Разработайте ER-модель данных для учета книжного фонда публичной библиотеки (студенческого абонента университетской библиотеки, читального зала научных работников).
- Дайте общую характеристику языка UML. Какова область применения этого языка? Чем принципиально отличается язык UML от языка программирования высокого уровня?
- Определите понятие "*пакет*" языка UML. Для чего могут быть использованы *пакеты* и как они изображаются на UML-диаграммах?
- Перечислите основные типы UML-диаграмм.
- Определите понятие "*интерфейс*" как компонент *UseCase*-диаграммы.

- Определите понятие "связь (отношение)" как компонент UseCase-диаграммы.
- Понятие "Класс" в языке UML, графическое представление этого элемента на UML-диаграммах классов.
- Понятие и формат спецификаций *атрибута и метода класса*.
- Отношения *ассоциации* и *обобщения* между классами: понятие, характеристики, обозначение на UML-диаграммах
- UML-диаграмма состояний: назначение и использование, основные компоненты и их графические обозначения.

7 ОСНОВНАЯ И ДОПОЛНИТЕЛЬНАЯ УЧЕБНАЯ ЛИТЕРАТУРА

7.1. Основная литература

1. Бозм Б. Программная инженерия. Основы и принципы. — 2-е изд. — М.: Вильямс, 2017. — 320 с.
2. Волк В.К. Практическое введение в программную инженерию: учебное пособие. – СПб.: Изд-во «Лань», 2019. – 100 с.: ил. – (Учебники для вузов. Специальная литература).
3. Леоненков А. Самоучитель UML – 2-е издание.: – Санкт-Петербург: БХВ-ПетербургСоммервилл И. Инженерия программного обеспечения – 6-е издание : пер. с англ. – Москва : Издательский дом «Вильямс», 2002. – 624 с. : ил.

7.2. Дополнительная литература

4. Брауде Э. Технология разработки программного обеспечения. пер. с англ. – Санкт-Петербург: Питер, 2004.
5. Буч Г. и др. Объектно-ориентированный анализ и проектирование с примерами приложений на C++, 3-е издание: пер. с англ. – Москва: Издательский дом «Вильямс», 2010 – 720 с.
6. Липаев В.В. Программная инженерия. Методологические основы : учебник – Москва : ТЕИС, 2006. – 608 с.

8 УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

7. Волк В.К. Основы программной инженерии. Методические указания по выполнению лабораторных работ и контрольных заданий (на правах рукописи).

9. РЕСУРСЫ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫЕ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Профессиональные стандарты IT-отрасли

8. Профессиональные стандарты РФ. ОПД 06: Связь, информационные и коммуникационные технологии.
URL:<https://classinform.ru/profstandarty /06-sviaz-informatcionnye-i-kommunikatcionnye-tekhnologii.html>.

9. Международный стандарт классификации профессий ISCO-08.
URL: <https://ilostat.ilo.org/methods/concepts-and-definitions/classification-occupation/>

Процессы жизненного цикла ПО

10. ГОСТ Р 59793-2021. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания. URL: <https://docs.cntd.ru/document/1200181804>.
11. ГОСТ Р ИСО/МЭК 12207-99. Информационная технология. Процессы жизненного цикла программных средств.
URL: <https://docs.cntd.ru/document/1200009075>
12. ГОСТ Р ИСО/МЭК 15504-1-2009. Информационная технология. Оценка процессов. URL: <https://docs.cntd.ru/document/1200076921>
13. Интегрированная модель оценивания зрелости продуктов и процессов разработки программных средств CMMI (Capability Maturity Model Integration for Product and Process Development).
URL: <https://intuit.ru/studies/courses/2262/160/lecture/4428?page=2>

Управление качеством

14. ГОСТ Р ИСО 9000-2001. Система менеджмента качества. Основные положения и словарь. URL: <https://docs.cntd.ru/document/1200015260>
15. ГОСТ Р ИСО/МЭК 9126-93. Информационная технология. Оценка программного продукта. Характеристики качества и руководство по их применению. URL: <https://docs.cntd.ru/document/1200009076>.
16. ГОСТ Р ИСО/МЭК 12119-2000. Информационная технология. Пакеты программ. Требования к качеству и тестирование. URL: <https://docs.cntd.ru/document/1200025075>.
17. ГОСТ Р ИСО/МЭК 25010-2015. Информационная технология. Системная и программная инженерия. Требования и оценка качества систем и программного обеспечения (SQuaRE). Модели качества систем и программных продуктов. URL: <https://docs.cntd.ru/document/1200121069>.

Разработка требований к ПО

18. ГОСТ 34.602-2020. Информационные технологии. Комплекс стандартов на автоматизированные системы. Техническое задание на разработку автоматизированной системы.
URL: <https://docs.cntd.ru/document/1200181804>.
19. Стандарт ISO/IEC/IEEE 29148:2018. Программная и системная инженерия. Процессы жизненного цикла. Разработка требований.
URL: <https://www.standards.ru/document/6439527.aspx>

20. ISO 15408-1-3.1999. (ГОСТ Р – 2002). Методы и средства обеспечения безопасности. Критерии оценки безопасности информационных технологий. Ч. 1. Введение и общая модель. Ч. 2. Защита функциональных требований. Ч. 3. Защита требований к качеству.
21. Классы защищенности для автоматизированных систем. URL: <http://infoprotect.net/note/klassy-zashhity-informatsii#2>
22. Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации. – Руководящий документ ФСТЭК.
URL: <https://fstec.ru/component/attachments/download/296>.

Управление программными проектами

23. Справочник менеджера проектов: главное о PMBOK. URL: <https://practicum.yandex.ru/blog/chto-takoe-pmbok-i-dlya-chego-on-nuzhen/>
<https://skillbox.ru/media/management/glavnoe-o-pmbok-svode-znaniy-po-upravleniyu-proektami-soderzhanie-i-osnovnye-idei/>
24. ISO 14764:1999. (ГОСТ Р–2002). Информационные технологии. Сопровождение программных средств.
25. ISO 15846:1998. ТО. Процессы жизненного цикла программных средств. Конфигурационное управление программными средствами.
26. ISO 16085:2004. Характеристики процессов управления рисками при разработке, применении и сопровождении программных средств.

Стандарты документирования программных проектов

27. ЕСПД (ГОСТ 19). Единая система программной документации.
URL: <https://www.standards.ru/collection.aspx?control=40&id=868075&catalogid=temat-sbor>.
28. ГОСТ Р 59795-2021. Комплекс стандартов на автоматизированные системы. Требования к содержанию документов.
URL: <https://protect.gost.ru/v.aspx?control=8&baseC=6&page=1&month=12&year=2021&search=&RegNum=1&DocOnPageCount=15&id=231352&pageK=E96B00A6-8F56-47D4-B377-400166AC3DC0>
29. РД 50-34.698-90. Методические указания. Информационная технология. Автоматизированные системы. Требования к содержанию документов.
URL: <https://docs.cntd.ru/document/1200006978>

**10 ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
И ИНФОРМАЦИОННО-СПРАВОЧНЫЕ СИСТЕМЫ**

№	Наименование	Использование
1	StarUML™. The Open Source UML/MDA Platform.	Используются в качестве Case-средства документирования программных проектов при выполнении лабораторных и контрольных работ.
2	Справочник по диаграммам языка UML. UML 2.5 Diagrams Overview. URL: https://www.uml-diagrams.org/	Используется для подготовки к выполнению лабораторных работ №10 – №13 по разделу №3 и для подготовки к выполнению контрольной работы.
3	ЭБС «Лань». URL: https://e.lanbook.com/books/	
4	ЭБС «Znanium». URL: https://znanium.ru/catalog/books/theme/06	

11 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

№	Наименование	Использование
1	Комплект: ноутбук, медиа-проектор, экран	Для демонстрации иллюстративного материала на лекционных занятиях.
2	Персональный компьютер стандартной комплектации	Для выполнения лабораторных и контрольных работ.

12 ДЛЯ СТУДЕНТОВ, ОБУЧАЮЩИХСЯ С ИСПОЛЬЗОВАНИЕМ ДИСТАНЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ (ДОТ)

При использовании электронного обучения и дистанционных образовательных технологий (далее ЭО и ДОТ) учебные занятия полностью или частично проводятся в режиме онлайн, при этом объем дисциплины и распределение нагрузки по видам работ соответствует тематическому плану (п. 4.1 рабочей программы).

При использовании ЭО и ДОТ распределение балльных оценок (таблица 6.2 рабочей программы) может быть изменено соответствующим решением кафедры, которое принимается с учетом мнения ведущего преподавателя и доводится до сведения студентов.

Согласно п.5.1, п.5.3 и п.6.1 рабочей программы использование ДОТ предусмотрено для онлайн-доступа к курсу лекций, проведения текущего контроля освоения лекционного материала в формате экспресс-тестирования и проведения рубежного контроля №1 в формате тестирования по материалу тематических разделов №1 и №2 дисциплины.

Аннотация
рабочей программы учебной дисциплины
ТЕХНОЛОГИИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ

образовательной программы высшего образования –
программы специалитета 10.05.03

Информационная безопасность автоматизированных систем

Направленность:

Безопасность открытых информационных систем

Форма обучения: **очная**

Трудоемкость дисциплины: 4 ЗЕ (144 академических часа)

Семестры: 4-й

Форма промежуточной аттестации: Экзамен,

Содержание дисциплины

Дисциплина «Технологии и методы программирования» включена в модуль «Информатика и программирование» обязательной части образовательной программы, для ее освоения необходимы компетенции, сформированные при изучении дисциплин «Информатика», «Основы программирования», «Основы информационной безопасности».

Результаты изучения дисциплины необходимы для освоения технологий проектирования информационных систем различного назначения и используются при освоении дисциплин «Безопасность систем баз данных», «Методы проектирования защищенных распределенных информационных систем», «Технология построения защищенных информационных приложений».

Цель изучения дисциплины – ознакомление со стандартами программной инженерии (в том числе – с требованиями к уровню защищенности автоматизированных систем и с профессиональными стандартами в сфере информационной безопасности) и технологиями промышленной разработки программного обеспечения.

Задачи дисциплины:

изучение:

- основных понятий, методологических основ и стандартов программной инженерии;
- структуры процессов жизненного цикла программного продукта;
- основных моделей жизненного цикла программного продукта.

практическое освоение:

- основ языка визуального моделирования (UML), используемого при анализе и проектировании программного обеспечения;
- CASE-средств поддержки программных проектов.