

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ ТАДЖИ-
КИСТАН
МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКО-ТАДЖИКСКИЙ (СЛАВЯНСКИЙ) УНИВЕРСИТЕТ»**

«Утверждаю»
Декан ЕНФ 
Муродзода Д.С.
« 31 » 08 2024 г.



РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ
Направление подготовки – 09.03.03. “Прикладная информатика”
профиль "Инженерия программного обеспечения"
Форма подготовки - очная
Уровень подготовки – бакалавриат

ДУШАНБЕ 2024

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования, утверждённого приказом Министерства образования и науки РФ от 19 сентября 2017 г. № 922

При разработке рабочей программы учитываются

- требования работодателей, профессиональных стандартов по направлению / специальности (при наличии) (для общепрофессиональных и профессиональных дисциплин);
- содержание программ дисциплин, изучаемых на предыдущих и последующих этапах обучения;
- новейшие достижения в данной предметной области.

Рабочая программа обсуждена на заседании кафедры Информатики и ИТ., протокол № 1 от 28 августа 2024 г.

Рабочая программа утверждена УМС естественнонаучного факультета, протокол № 1 от 29 августа 2024 г.

Рабочая программа утверждена Учёным советом естественнонаучного факультета, протокол № 1 от 30 августа 2024г.

Заведующий кафедрой, к.э.н., доцент



Лешукович А.И.

Зам. председателя УМС факультета
к. ф-м.н., доцент



Халимов И.И.

Разработчик, к.э.н., доцент



Мирзокаримов О.А.

Расписание занятий дисциплины

Ф.И.О. преподавателя	Аудиторные занятия				Место работы преподавателя
	Лекция	Практические занятия	(КСР, лаб.)	Приём СРС	
Мирзокаримов О.А.	Вторник	Четверг	Пятница	Пятница	РТСУ, кафедра информатики и ИТ, Корпус 2, 216 каб.
	08:00-09:20	09:30-10:50	11:00-12:40	14:10 15:30	
	второй корпус:	второй корпус:	второй корпус:		
	Ауд.213	Ауд.221	Ауд.223	Каб. 210	

1. ЦЕЛИ И ЗАДАЧИ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

1.1. Цели изучения дисциплины

Целями освоения дисциплины является приобретение комплекса теоретических знаний и методологических основ в области проектирования ИС, а также практических навыков необходимых для квалифицированной разработки требований и формированию технического задания на создание ИС. Дисциплина является важной составной частью подготовки специалиста в области информационных технологий. Основой курса являются методологии системного анализа и моделирования, позволяющие на этапе создания информационной системы решить следующие основные задачи: - обеспечение требуемой функциональности системы и адаптивности к изменяющимся условиям ее функционирования; - проектирование реализуемых в системе объектов данных; - проектирование программ и средств интерфейса (экранных форм, отчетов), которые будут обеспечивать выполнение запросов к данным; Программой курса предусматривается изучение CASE-инструментов поддержки проектирования информационных систем.

1.2. Задачи изучения дисциплины

Задачи дисциплины формулируются в соответствии с требованиями ФГОС, предъявляемые к компетенциям обучающегося.

1.3. В результате изучения дисциплины «Проектирование информационных систем» у обучающихся формируются следующие общепрофессиональные ко лекции:

Таблица 1.

Р1.3	Результаты освоения ОПОП	Перечень планируемых результатов обучения	Вид оценочного знания
УК-1	Способностью осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	ИУК-1.1. Анализирует задачу, выделяя ее базовые составляющие. Осуществляет декомпозицию задачи; ИУК-1.2. Находит и критически анализирует информацию,	Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.

		<p>необходимую для решения поставленной задачи;</p> <p>ИУК-1.3. Рассматривает различные варианты решения задачи, оценивая их достоинства и недостатки;</p> <p>ИУК-1.4. Грамотно, логично, аргументированно формирует собственные суждения и оценки. Отличает факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности;</p> <p>ИУК-1.5. Определяет и оценивает практические последствия возможных решений задачи.</p>	
ОПК-4	<p>Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью</p>	<p>ИОПК-4.1. Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.</p> <p>ИОПК-4.2. Применяет стандарты, нормы и правила оформления технической документации на различных стадиях жизненного цикла информационной системы.</p> <p>ИОПК-4.3. Разрабатывает техническую документацию на различных этапах жизненного цикла информационной системы.</p>	<p>Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.</p>
ОПК-6	<p>Способен анализировать и разрабатывать организационно-технические и экономические процессы с</p>	<p>ИОПК-6.1. Использует основы теории систем и системного анализа, дискретной математики, теории</p>	<p>Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контроль-</p>

		<p>вероятностей и математической статистики, методов оптимизации и исследования операций, нечетких вычислений, математического и имитационного моделирования.</p> <p>ИОПК-6.2. Применяет методы теории систем и системного анализа, математического, статистического и имитационного моделирования для автоматизации задач принятия решений, анализа информационных потоков, расчета экономической эффективности и надежности информационных систем и технологий.</p> <p>ИОПК-6.3. Проводит инженерные расчеты основных показателей результативности создания и применения информационных систем и технологий.</p>	<p>ная работа. Устный опрос.</p>
<p>ОПК-8</p>	<p>Способен принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла</p>	<p>ИОПК-8.1. Применяет основные технологии создания и внедрения информационных систем, стандарты управления жизненным циклом информационной системы.</p> <p>ИОПК-8.2. Осуществляет организационное обеспечение выполнения работ на всех стадиях и в процессах жизненного цикла информационной системы.</p> <p>ИОПК-8.3. Состав-</p>	

		ляет плановую и отчетную документацию по управлению проектами создания информационных систем на стадиях жизненного цикла.	
ОПК-9	Способен принимать участие в реализации профессиональных коммуникаций с заинтересованными участниками проектной деятельности и в рамках проектных групп	<p>ИОПК-9.1. Использует инструменты и методы коммуникаций в проектах; каналы коммуникаций в проектах; модели коммуникаций в проектах; технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии, технологии подготовки и проведения презентаций.</p> <p>ИОПК-9.2. Осуществляет взаимодействие с заказчиком в процессе реализации проекта; принимать участие в командообразовании и развитии персонала.</p> <p>ИОПК-9.3. Участвует в проведении презентаций, переговоров, публичных выступлений.</p>	

2.МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Данная дисциплина входит в базовый цикл вариативной части дисциплины Б1.0.26 ОПОП бакалавриата ФГОС ВО и является обязательной дисциплиной.

Таблица 2.

№ п/п	Наименование дисциплины	Семестр	Место дисциплины в структуре ООП
1.	<i>Программирование</i>	1-2-3	Б1.О.13
2.	<i>Практикум по программированию</i>	2-4	Б1.О.21
3.	<i>Интеллектуальные информационные системы</i>	5	Б1.В.05
4.	<i>Проектирование информационных систем</i>	5-6	Б1.О.26
5.	<i>Информационные системы и технологии</i>	5	Б1.О.25
6.	<i>Теория систем и системный анализ</i>	7	Б1.В.12
7.	<i>Ознакомительная практика</i>	4	Б2.О.01(У)

8.	<i>Технологическая (проектно-технологическая) практика</i>	6	Б2.О.02(П)
9.	<i>Разработка системы электронного документооборота</i>	7	Б1.В.10
10.	<i>Преддипломная практика</i>	8	Б2.В.01(Пд)

При освоении данной дисциплины необходимы умения и готовность («входные» знания) обучающегося по дисциплинам 1-2, указанных в Таблице 2. Дисциплины 3-10 относятся к группе, которые должны использовать «входные» знания данной дисциплины.

3. СТРУКТУРА И СОДЕРЖАНИЕ КУРСА, КРИТЕРИИ НАЧИСЛЕНИЯ БАЛЛОВ

Объем дисциплины составляет 8 зачетных единиц, всего 288 часов, из которых: лекции 26 часов, практические занятия 26 часа, лабораторные работы 26 часов, КСР – 26 часа, всего часов аудиторной нагрузки - 104 часа, самостоятельная работа – 130 часов, контроль – 54 часов. Зачет –5-й семестр, экзамен –6-й семестр

3.1 Структура и содержание теоретической части курса 5-й семестр

Тема 1. Основы проектирования программного обеспечения. Выбор представления внутренних данных, разработка основного алгоритма, документирование ПО, тестирование и подбор тестов,, выбор представления входных данных.

Тема 2. Методы и технологии проектирования ИС. Классификация методов проектирования. Методы проектирования ИС. Концепции и теоретические основы. Нотации, используемые при построении статической структуры и динамического поведения ИС. Процедуры, определяющие практическое применение методов. Оценка результатов.

Тема 3. Методы быстрой разработки ПО. Подход RAD (направлен на быструю разработку приложений)— концепция организации технологического процесса разработки программных продуктов, ориентированная на максимально быстрое получение результата в условиях сильных ограничений по срокам и бюджету и нечётко определённых требований к продукту. Эффект ускорения разработки достигается путём использования соответствующих технических средств и непрерывного, параллельного с ходом разработки, уточнения требований и оценки текущих результатов с привлечением заказчика.

Тема 4. Проектирование структуры программного обеспечения. Операционный и декларативный Операционный подход. Основные операции, исполняемые компьютером. Декларативный подход. Логическое программирование.. Особенности языка Пролог. Этапы логической программы. Функциональное программирование. Списки и обработка списков. Логический принцип построения функциональных программ. Функциональное программирование на язык Лисп

Тема 5. Проектирование информационного обеспечения ИС. Определение сущности информационного, математического и программного обеспечения. Ознакомление с процессом классификации и кодирования информации. Рассмотрение и характеристика основных принципов проектирования унифицированной системы документации.

Тема 6. Структурный подход. Сущность структурного подхода к разработке ИС заключается в ее декомпозиции (разбиении) на автоматизируемые функции: система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, подразделяемые на задачи и так далее. Процесс разбиения продолжается вплоть до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимосвязаны. При разработке системы "снизу-вверх" от отдельных задач ко всей системе целостность теряется, возникают проблемы при информационной стыковке отдельных компонентов.

Тема 7. Объектно-ориентированный подход. Методология проектирования, соединяющая в себе процесс объектной декомпозиции и приемы представления как логической и физической, так и статической и динамической моделей проектируемой системы. Объектно-ориентированный анализ направлен на создание моделей, более близких к реальности, с использованием объектно-ориентированного подхода.

Тема 8. Классы. Создания классов. Наследование, встраивание и полиморфизм

3.2. Структура и содержание практической части курса

Структура и содержание практической части курса включает в себя тематику и содержание

практических занятий (ПЗ) и лабораторных работ.

Практические занятия (16 часов)

5-й семестр

- ПР1. Основные этапы разработки программного обеспечения. Построение модели. (2 часа).*
- ПР2. Методы и технологии проектирования ИС. Классификация методов проектирования. (2 часа).*
- ПР3. Методы быстрой разработки ПО. Подход RAD (2 часа).*
- ПР4. Проектирование структуры программного обеспечения. Обработка списков (2 часа).*
- ПР5. Проектирование информационного обеспечения ИС. Моделирование данных (2 часа).*
- ПР6. Моделирование потоков данных. (2 часа).*
- ПР7. Структурный подход. (2 часа).*
- ПР8. Метод функционального моделирования – SADT (2 часа).*

Лабораторные работы (16 часа)

5-й семестр

- Лабораторная работа №1. Технология проектирования ИС (2 часа).*
- Лабораторная работа №2. Методы проектирования ИС (2 часа).*
- Лабораторная работа №3. Модели жизненного цикла (2 часа).*
- Лабораторная работа №4. Методы быстрой разработки ПО-Подход RAD (2 часа).*
- Лабораторная работа №5. Проектирование структуры программного обеспечения (2 часа).*
- Лабораторная работа №6. Моделирование потоков данных (2 часа).*
- Лабораторная работа №7. Структурный подход (2 часа).*
- Лабораторная работа №8. Метод SADT (2 часа).*

Структура и содержание КСР (16 часов)

5-й семестр

- КСР-1.** Основные этапы разработки программного обеспечения (ПО). (2 часа).
- КСР-2.** Процедуры, определяющие практическое применение методов проектирования. (2 часа).
- КСР-3.** Построение модели. Дедуктивный подход. Индуктивный подход. (2 часа).
- КСР-4.** Инспекция и сквозной просмотр. Граничные условия. (2 часа).
- КСР-5.** Рассмотрение и характеристика основных принципов проектирования унифицированной системы документации. (2 часа).
- КСР-6.** Структурное проектирование. (2 часа).
- КСР-7.** Информационное моделирование 2 часа).
- КСР-8.** Объектно-ориентированное проектирование. (2 часа).

3.3. Структура и содержание теоретической части курса

6-й семестр

Тема 9. Объектные модели. Разработка программного обеспечения тоже существует несколько подходов к моделированию. Важнейшие из них - алгоритмический и объектно-ориентированный. Алгоритмический метод представляет традиционный подход к созданию программного обеспечения. Основным строительным блоком является процедура или функция, а внимание уделяется, прежде всего вопросам передачи управления и декомпозиции больших алгоритмов на меньшие. Ничего плохого в этом нет, если не считать того, что системы не слишком легко адаптируются. При изменении требований или увеличении размера приложения (что происходит нередко) сопровождать их становится сложнее. Наиболее современным подходом к разработке программного обеспечения является объектно-ориентированный. Здесь в качестве основного строительного блока выступает объект или класс. В самом общем смысле объект - это сущность, обычно извлекаемая из словаря предметной области или решения, а класс является описанием множества однотипных объектов. Каждый объект обладает идентичностью (его можно поименовать или как-то по-другому отличить от прочих объектов), состоянием (обычно с объектом бывают связаны некоторые данные) и поведением (с ним можно что-то делать или он сам может что-то делать с другими объектами).

Тема 10. Универсальный язык моделирования - UML является языком широкого профиля, это — открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является

языком программирования, но на основании UML-моделей возможна генерация кода.

Тема 11. Диаграммы UML . В распоряжение проектировщика системы Rational Rose предоставляет следующие типы диаграмм, последовательное создание которых позволяет получить полное представление о всей проектируемой системе и об отдельных ее компонентах: Диаграмма прецедентов (Use-case diagram); Диаграмма классов (Class diagram); Диаграмма активностей (Activity diagram); Диаграмма последовательности (Sequence diagram); Диаграмма развертывания (Deployment diagram); Диаграмма сотрудничества (Collaboration diagram);

Диаграмма объектов (Object diagram); Диаграмма состояний (Statechart diagram).

Тема 12. Статическая структура ИС. Структурные диаграммы.

Диаграммы пакетов (Package); классов (Class); объектов (Object); компонентов (Component); распределения (Развертывания Deployment); композитных структур (Composite structure). Структурные диаграммы представляют статическую структуру системы в терминах классов объектно-ориентированного программирования, диаграмма классов относится к логическому представлению системы. Классы реализуют типы объектов, которыми манипулирует система. На диаграмме классов с помощью специальной системы обозначений изображается некоторый набор классов с описанием их внутренней структуры.

Тема 13. Диаграммы для описания динамического поведения ИС.

Диаграммы прецедентов использования (Use Case); деятельности (Activity); конечных автоматов (State Machine); Описывает функциональные возможности системы. Изображаются внешние *субъекты* (actors) и их связь с *аспектами использования* системы (use-cases). Диаграмма использования лишь внешнее представление поведения системы, и не имеет отношения к описанию реализации функциональных возможностей внутри системы. С помощью диаграммы использования описываются функциональные требования к конечному продукту. Диаграммы последовательностей (Sequence); коммуникации (Communication); обзора взаимодействий (Interaction overview); синхронизации (Timing). Используется, чтобы проследить выполнение сценария. Преимущество диаграммы взаимодействий в том, что на ней легче читается порядок послышки сообщений, а преимущество диаграммы объектов в том, что она лучше подходит для многих объектов со сложными вызовами и допускает включение другой информации: связи, значения атрибутов, роли, блок-схемы и видимость. Так как оба типа диаграмм имеют неоспоримые достоинства, мы пользуемся в нашем методе обоими

Структура и содержание практической части курса

Структура и содержание практической части курса включает в себя тематику и содержание практических занятий (ПЗ) и лабораторных работ.

Практические занятия (10 часов)

6-й семестр

- ПР1.** Основные понятия объектно - ориентированного подхода
- ПР2.** Классы. Создания классов. Наследование, встраивание и полиморфизм
- ПР3.** Отношения между классами
- ПР4.** Объектные модели UML
- ПР5.** Представление использования Use-Case-View.

Лабораторные работы (10 часа)

6-й семестр

- Лабораторная работа №1. Диаграмма Use-Case (2 часа).*
- Лабораторная работа №2. Диаграмма последовательности. (2 часа).*
- Лабораторная работа №3. Диаграмма развертывания. (2 часа).*
- Лабораторная работа №4. Диаграмма активности (2 часа).*
- Лабораторная работа №5. Диаграмма классов (2 часа).*

Структура и содержание КСР (10 часов)

6-й семестр

- КСР-1. Диаграммы коммуникации и последовательности (2 часа).*
- КСР-2. Диаграмма активности*
- КСР-3. Диаграмма обзора взаимодействия (2 часа).*
- КСР-4. Диаграмма синхронизации (2 часа).*
- КСР-5. Подход Rapid Application Development (RAD). (2 часа).*

Таблица 3.

№ п/п	Раздел дисциплины	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Литература	Кол-во баллов в неделю
		Лек.	Пр.	Лаб.	КСР		
5 семестр							
1.	Тема 1. Основы проектирования программного обеспечения Современный подход. Декомпозиция задачи. Абстракция через параметризацию. Основные этапы разработки программного обеспечения (ПО). Детальное проектирование. Построение модели. Дедуктивный подход. Индуктивный	2				1(5-15) 5 (с.11-16), 11 (с.31-49) 16 (с.19-33)	12,5
2.	ПР1. Основные этапы разработки программного обеспечения. Построение модели		2				
3.	Лабораторная работа №1. Технология проектирования ИС.			2			
4.	КСР-1. Основные этапы разработки программного обеспечения (ПО).				2		
5.	Тема 2. Методы и технологии проектирования ИС. Классификация методов проектирования. Методы проектирования ИС. Концепции и теоретические основы. Нотации, используемые при построении статической структуры и динамического поведения ИС. Процедуры, определяющие практическое применение методов. Оценка результатов.	2				4(с.46-51) 3(с.92-93) 4(с.51-58) 3(с.94-96)	12,5
6.	ПР2. Методы и технологии проектирования ИС.Классификация методов проектирования.		2				
7.	Лабораторная работа №2. Методы проектирования ИС			2			
8.	КСР-2. Процедуры, определяющие практическое применение методов проектирования.				2		
9.	Тема 3. Методы быстрой разработки ПО. Подход RAD направлен на быструю разработку приложений) — концепция организации технологического процесса разработки программных продуктов, ориентированная на максимально быстрое получение результата в условиях сильных ограничений по срокам и бюджету и нечётко определённых требований к продукту. Эффект ускорения разработки достигается путём использования соответствующих технических средств и непрерывного, параллельного с ходом разработ-	2				1(с.32-36) 5(с.213-220) 11(с.97-120) 4(с.99-102) 1(с.36-37)	12,5

	ки, уточнения требований и оценки текущих результатов с привлечением заказчика.						
10.	ПР3. Методы быстрой разработки ПО. Подход RAD		2				
11.	Лабораторная работа №3. Методы быстрой разработки ПО-Подход RAD			2			
12.	КСР-3. Построение модели. Дедуктивный подход. Индуктивный подход. (2 часа).				2		
13.	Тема 4. Проектирование структуры программного обеспечения. операционный и декларативный Операционный подход. Основные операции, исполняемые компьютером. Декларативный подход. Логическое программирование. Особенности языка Пролог. Этапы логической программы. Функциональное программирование. Списки и обработка списков. Логический принцип построения функциональных программ. Функциональное программирование на языке Лисп	2				1(с.37-39) 5(с.37-91) 11(с.97-120) 5 (с.91-93) 1 (с.115-116)	12,5
14.	ПР4. Проектирование структуры программного обеспечения. Обработка списков		2				
15.	Лабораторная работа №4. Модели жизненного цикла			2			
16.	КСР-4. Инспекция и сквозной просмотр. Граничные условия. (2 часа).				2		
17.	Тема 5. Проектирование информационного обеспечения ИС. Определение сущности информационного, математического и программного обеспечения. Ознакомление с процессом классификации и кодирования информации. Рассмотрение и характеристика основных принципов проектирования унифицированной системы документации.	2				1(с.50-59) 21(с.149-220) 5(с.162, 178, 221) 1(с.116-117)	12,5
18.	ПР5. Проектирование информационного обеспечения ИС. Моделирование данных		2				
19.	Лабораторная работа №5. Проектирование структуры программного обеспечения			2			
20.	КСР-5. Рассмотрение и характеристика основных принципов проектирования унифицированной системы документации. (2 часа).				2		
21.	Тема 6. Структурный подход. Сущность структурного подхода к разработке ИС (заключается в ее декомпозиции разбиении) на автоматизируемые функции: система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, подразделяемые на задачи и	2				1(с.39-49) 5(с.316-366) 11(с.135-154) 16(с.224-241)	12,5

	так далее. Процесс разбиения продолжается вплоть до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты заимовязаны. При разработке системы "снизу-вверх" от отдельных задач ко всей системе целостность					5(с.371-374) 1(с.118-119)	
22.	ПР6. Моделирование потоков данных.		2				
23.	Лабораторная работа №6. Моделирование потоков данных			2			
24.	КСР-6. Структурное проектирование.				2		
25.	Тема 7. Объектно-ориентированный подход. Методология проектирования, соединяющая в себе процесс объектной декомпозиции и приемы представления как логической и физической, так и статической и динамической моделей проектируемой системы. Объектно-ориентированный анализ направлен на создание моделей, более близких к реальности, с использованием объектно-ориентированного подхода.	2				1(с.64-82) 8 (с.57-105) 17(с.34-38) 17(с.39-41) 2(с.10-11)	12,5
26.	ПР7. Структурный подход.		2				
27.	Лабораторная работа №7. Структурный подход			2			
28.	КСР-7. Информационное моделирование				2		
29.	Тема 8. Классы. Создания классов. Наследование, встраивание и полиморфизм	2				1(с.83-96) 8 (с.106-125)	12,5
30.	ПР8. Метод функционального моделирования – SADT		2			17(с.42-75)	
31.	Лабораторная работа №8. Метод SADT			2		13(с.47-56) 17(с.76-85)	
32.	КСР-8. Объектно-ориентированное проектирование.				2	2(с.12-13)	
	ИТОГО:	16	16	16	16		200

№ п/п	Раздел дисциплины	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Литература	Кол-во баллов в неделю
		Лек.	Пр.	Лаб.	КСР		
6 семестр		Лек.	Пр.	Лаб.	КСР		
1.	Тема 1. Объектные модели Объектная модель Object Management Group (OMG). Модель объектно-ориентированных систем управления базами данных – Object Database Management Group. Основные этапы создания инфор-	2				1(с.124-126) 6(с.479-482) 2(с.16-17)	12,5

	мационных систем. Вертикальное и горизонтальное проектирование. Функциональность приложения. Масштабность приложения. Модель, реализованная в среде CORBA. Брокер запросов. Система типов CORBA. Создание и уничтожение объектов. Идентификация объектов. Метаданные. Определение динамики поведения объектов. Модули или компоненты						
2.	ПР1. Основные понятия объектно - ориентированного подхода		2				
3.	Лабораторная работа №1. Введение в UML			2			
4.	КСР-1. Подход Rapid Application Development (RAD).				2		
5.	Тема 2. Универсальный язык моделирования Универсальный язык моделирования Unified Modeling Language – UML. Цели разработки. Стандартизация UML. CASE – средства: Microsoft Visual Modeler. Нотация UML. Диаграммы UML для моделирования. Варианты использования. Последовательность транзакций. Взаимодействие между компонентами системы и взаимодействие между пользователем и системой.	2				8 (с.175-186) 17(с.17-24) 17(с.17-24) 2(с.21-22)	12,5
6.	ПР2. Классы. Создания классов. Наследование, встраивание и олиморфизм		2				
7.	Лабораторная работа №2. Диаграмма Use-Case			2			
8.	КСР-2. Диаграммы коммуникации и последовательности				2		
9.	Тема 3. Диаграммы UML. В распоряжение проектировщика системы Rational Rose предоставляет следующие типы диаграмм, последовательное создание которых позволяет получить полное представление о всей проектируемой системе и об отдельных ее компонентах: Диаграмма прецедентов (Use-case diagram); Диаграмма классов (Class diagram); Диаграмма активностей (Activity diagram); Диаграмма последовательности (Sequence diagram); Диаграмма развёртывания (Deployment diagram); Диаграмма сотрудничества (Collaboration diagram); Диаграмма объектов (Object diagram); Диаграмма состояний (Statechart diagram).	2				8 (с.186-195) 10(с.584-601) 17(с.34-42) 13(с.26-33) 2(с.25-26)	12,5
10.	ПР3. Объектные модели UML		2				
11.	Лабораторная работа №3. Диаграмма последовательности.			2			
12.	КСР-3. Диаграмма активности				2		

13.	Тема 4. Статическая структура ИС. Структурные диаграммы. Диаграммы пакетов (Package); классов (Class); объектов (Object); компонентов (Component); распределения (Развертывания Deployment); композитных структур (Composite structure). Структурные диаграммы представляют статическую структуру системы в терминах классов объектно-ориентированного программирования, диаграмма классов относится к логическому представлению системы. Классы реализуют типы объектов, которыми манипулирует система. На диаграмме классов с помощью специальной системы обозначений изображается некоторый набор классов с описанием их внутренней структуры.	2				8 (с.186-275) 13(с.38-45) 2(с.25-26) 13(с.26-33) 2(с.25-26)	12,5
14.	ПР4. Представление использования Use-Case-View.		2				
15.	Лабораторная работа №4. Диаграмма активности			2			
16.	КСР-4. Диаграмма обзора взаимодействия				2		
17.	Тема 5. Диаграммы для описания динамического поведения ИС. Диаграммы прецедентов использования (Use Case); деятельности (Activity); конечных автоматов (State Machine); описывает функциональные возможности системы. Изображаются внешние <i>субъекты</i> (actors) и их связь с <i>аспектами использования</i> системы (use-cases). Диаграмма использования лишь внешнее представление поведения системы, и не имеет отношения к описанию реализации функциональных возможностей внутри системы. С помощью диаграммы использования описываются функциональные требования к конечному продукту. Диаграммы последовательностей (Sequence); коммуникации (Communication); обзора взаимодействий (Interaction overview); синхронизации (Timing). Используется, чтобы проследить выполнение сценария. Преимущество диаграммы взаимодействий в том, что на ней легче читается порядок послышки сообщений, а преимущество диаграммы объектов в том, что она лучше подходит для многих объектов со сложными вызовами и допускает включение другой информации: связи, значения атрибутов, роли, блок-схемы и видимость. Так как оба типа диаграмм имеют неоспоримые достоинства, мы пользуемся в нашем методе обоими	2				13(с.47-52) 17(с.110-115) 2(с.29-30) 2 8 (с.123-148) 17(с.42-109)	12,5

18.	ПР5. Логическое представление (Logical View) .		2				
19.	Лабораторная работа №5. Диаграмма классов			2			
20.	КСР-5. Диаграмма синхронизации				2		
	ИТОГО:	10	10	10	10		

Формы контроля и критерии начисления баллов

Контроль усвоения студентом каждой темы осуществляется в рамках балльно-рейтинговой системы (БРС), включающей текущий, рубежный и итоговый контроль. Студенты **1-го курса**, обучающиеся по кредитно-рейтинговой системе обучения, могут получить максимально возможное количество баллов - 300. Из них на текущий и рубежный контроль выделяется 200 баллов или 49% от общего количества.

На итоговый контроль знаний студентов выделяется 51% или 100 баллов.

Порядок выставления баллов: 1-й рейтинг (1-7 недели до 12,5 баллов+12,5 баллов (8 неделя – Рубежный контроль №1) = 100 баллов), 2-й рейтинг (9-15 недели до 12,5 баллов+12,5 баллов (16 неделя – Рубежный контроль №2) = 100 баллов), итоговый контроль 100 баллов.

К примеру, за текущий и 1-й рубежный контроль выставляется 100 баллов: лекционные занятия – 21 балл, за практические занятия (КСР, лабораторные) – 31,5 балл, за СРС – 17,5 баллов, требования ВУЗа – 17,5 баллов, рубежный контроль – 12,5 баллов.

В случае пропуска студентом занятий по уважительной причине (при наличии подтверждающего документа) в период академической недели деканат факультета обращается к проректору по учебной работе с представлением об отработке студентом баллов за пропущенные дни по каждой отдельной дисциплине с последующим внесением их в электронный журнал.

Итоговая форма контроля по дисциплине (зачет, экзамен) проводится как в форме тестирования, так и в традиционной (устной) форме. Тестовая форма итогового контроля по дисциплине предусматривает: для естественнонаучных направлений – 10 тестовых вопросов на одного студента, где правильный ответ оценивается в 10 баллов, для гуманитарных направлений – 25 тестовых вопросов, где правильный ответ оценивается в 4 балла. Тестирование проводится в электронном виде, устный экзамен на бумажном носителе с выставлением оценки в ведомости по аналогичной системе с тестированием.

Таблица 4.

Неделя	Активное участие на лекционных занятиях, написание конспекта и выполнение других видов работ*	Активное участие на практических (семинарских) занятиях, КСР	СРС Написание реферата, доклада, эссе Выполнение других видов работ	Выполнение положения высшей школы (установленная форма одежды, наличие рабочей папки, а также других пунктов устава высшей школы)	РК №1	Всего
1	2	3	4	5	6	7
1	3	4,5	2,5	2,5	-	12,5
2	3	4,5	2,5	2,5	-	12,5
3	3	4,5	2,5	2,5	-	12,5
4	3	4,5	2,5	2,5	-	12,5
5	3	4,5	2,5	2,5	-	12,5
6	3	4,5	2,5	2,5	-	12,5
7	3	4,5	2,5	2,5	-	12,5
8	-	-	-	-	12,5	12,5
Первый рейтинг	21	31,5	17,5	17,5	12,5	100

Формула вычисления результатов дистанционного контроля и итоговой формы контроля по дисциплине за семестр для студентов 4-го курсов:

$$ИБ = \left[\frac{(P_1 + P_2)}{2} \right] \cdot 0,49 + Эи \cdot 0,51, \text{ где } ИБ - \text{итоговый балл, } P_1 - \text{итоги первого рейтинга,}$$

P_2 - итоги второго рейтинга, $Эи$ – результаты итоговой формы контроля (экзамен).

4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Методы обработки информации» включает в себя:

1. план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;
2. характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;
3. требования к представлению и оформлению результатов самостоятельной работы;
4. критерии оценки выполнения самостоятельной работы.

План-график выполнения самостоятельной работы по дисциплине

4.1. План-график выполнения самостоятельной работы по дисциплине

№ п/п	Объем СРС в ч.	Тема СРС	Форма и вид результатов самостоятельной работы	Форма контроля
1.	4	Основные этапы разработки программного обеспечения. Построение модели.	Вопросы 1-4. Описание технологии разработки, реферат	Опрос
2.	4	Методы и технологии проектирования ИС. Классификация методов	Вопросы 5-8. Презентация методов	Выступление
3.	6	Методы быстрой разработки ПО. Подход	Вопросы 8-10. Презентация, доклад	Выступление
4.	6	Проектирование структуры программного обеспечения. Обра-	Вопросы 11-13. Выполнение задания 1 (1-10).	Защита работы. Вы-
5.	4	Проектирование информационного обеспечения ИС. Моделирование дан-	Выполнение задания 1. Конспект, презентация (вопросы 14-15)	Опрос, Выступление
6.	4	Моделирование потоков данных.	Выполнение задания 2	Защита работы.
7.	6	Структурный подход.	Вопросы 16-17. Выполнение задания 3	Защита работы.
8.	6	Метод функционального моделирования – SADT	Вопросы 16-17. Выполнение задания 4	Защита работы.
9.	4	Структурное программирование	Выполнение задания 5	Защита работы.
10.	4	Основные понятия объектно-ориентированного подхода	Вопросы 18-25. Выполнение задания 6	Защита работы.

11.	4	Классы. Создания классов. Наследование, встраивание и полиморфизм	Вопросы 26-29. Выполнить задания 2 и описать в терминах классов.	Опрос. Защита работы
12.	4	Отношения между классами	Вопросы 30-31. Реферат. Выполнение задания 7	Защита реферата. Защита работы
13.	4	Объектные модели	Вопросы 32-37. Презентация	Опрос. Выступление
14.	4	Универсальный язык моделирования Unified Modeling Language – UML.	Вопросы 38-40. Выполнение задания 8 (1-4)	Защита работы
15.	4	Представление использования Use-Case- View.	Вопросы 41-44. Выполнение задания 9	Защита работы
16.	4	Логическое представление (Logical View) .	Вопросы 45-46. Выполнение задания 8 (4-10)	Защита работы
17.	4	Основные диаграммы UML. (Use-Case)	Вопросы 50-51. Выполнение задания 10	Защита работы
18.	4	Статическая структура ИС. Структурные диаграммы. Диаграммы пакетов (Package); классов (Class); объектов (Object); компонентов (Component); распределе-	Вопросы 52-54. Выполнение задания 11	Защита работы
19.	4	Диаграммы для описания динамического поведения ИС. Диаграммы преце-	Вопросы 55-59. Выполнение задания 11	Защита работы
20.	4	Диаграмм в реализации, диаграммы размещения,	Вопросы 60-62. Выполнение задания 12	Защита работы
21.	4	Диаграммы деятельности (Activity).	Вопросы 63-64. Выполнение задания 13	Защита работы
22.	4	Диаграммы состояний	Вопросы 65-66. Выполнение задания 14	Защита работы
23.	4	Конечных автоматов (State Machine)	Вопросы 67-68. Выполнение задания 15	Защита работы
24.	4	Принципы и стандарты документирования	Вопросы 69-74. Презентация, доклад	Выступление

4.2 Характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению;

Для выполнения задания, прежде всего, необходимо ознакомиться и изучить основные положения теоретических материалов соответствующей темы из литературных источников. Они указаны в разделе «Содержание и структура дисциплины». Конспекты и задания можно выполнить в отдельном тетради или в лекционной (практической) тетради в произвольной форме.

4.3 Критерии оценки выполнения самостоятельной работы.

Критерии оценки выполнения самостоятельной работы является полнота освещения вопроса, логичность изложения, проявления самостоятельность в обработке материала.

5. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература

1. Грекул В. И., Коровкина Н. Л., Левочкина Г. А. Основы проектирования информационных систем (учебник) Национальный исследовательский университет «Высшая школа экономики» (г. Москва). 2021 385 стр.
2. Черткова Е. А. Введение в программную инженерию; Инженерия ПО Национальный исследовательский университет «Высшая школа экономики» (г. Москва). 2021 147 стр
3. Иркаев Б.Н., Умаров М.А., Бахтеев К.С. Основы информационных технологий, Учебник, РТСУ, Душанбе, 2018, 370 стр.
4. Кудрина Е. В., Огнева М. В. Основы алгоритмизации и программирования на языке C#. Национальный исследовательский Саратовский государственный университет имени Н.Г. Чернышевского (г. Саратов). УМО ВО 2021 286 стр.
5. Огнева М. В., Кудрина Е. В. ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ C++: ПРАКТИЧЕСКИЙ КУРС. Учебное пособие для вузов УМО ВО 2021 335 с.
6. Маркин А. В. ПРОГРАММИРОВАНИЕ НА SQL В 2 Ч. ЧАСТЬ 2 2-е изд., испр. и доп. Учебник и практикум для вузов УМО ВО 2021 340 с.
7. Григорьев М. В., Григорьева И. И. Методы и средства проектирования информационных систем и технологий; Тюменский государственный университет (г. Тюмень). 2021 318 стр.
8. Умаров М.А., Бахтеев К.С., Мирзокаримов О.А, «Проектирование информационных

5.2 Дополнительная литература

9. Умаров М.А. Проектирование информационных систем. Часть 1. Методологические основы проектирования информационных систем. Учебное пособие. Душанбе: - РТСУ, 2011. 125с.
10. Брауде Э.Дж., Технология разработки программного обеспечения. – СПб.: Питер, 2004. – 655с.
11. Бобровский С.И. Delphi 7. Учебный курс – СПб.: Питер, 2004. – 736с.
12. Буч Г., Максимчук Р., Энгл М. и др. Объектно-ориентированный анализ и проектирование с примерами приложений, 3-е изд.: Пер. с англ.- М.: ООО «И.Д.Вильямс», 2008.-720с.
13. Буч Г., Рамбо Д., Джекобсон И. Язык UML. Руководство пользователя: Пер. с англ.- М.: ДМК, 2010. 432 с.
14. Информационные системы в экономике и управлении: Учебник / Под ред. Проф. В.В. Трифонова. – М.: Высшее образование, 2009. – 480 с.
15. Гарнаев А. И др. Microsoft Office 2000. Разработка приложений. СПб.: ВHV, 2000, 656 с. С ил
16. Вендров А.М. Проектирование программного обеспечения экономических информационных систем.- М.2000
17. Ефимов Е.Н., Петрушкина С.М., Панферова Л.Ф., Хашиева Л.И. Информационные системы в экономике.- М.: ИКЦ «МарТ», 2004. – 352с.
18. Кватрани Т. Rational Rose 2000 и UML. Визуальное моделирование: Пер с англ. – М.: ДМК Пресс, 2001.-176с.
19. Ли И.Т., Умаров М.А., Методические рекомендации по выполнению дипломных проектов для специальности 010502 «Прикладная информатика (в экономике)» Душанбе: РТСУ, 2009.-101с.
20. [http:// www.citforum.ru](http://www.citforum.ru) – материалы сайта Сервер информационных технологий.
21. <http://www.computer.org/tab/seprof/code.htm>

5.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

22. [http:// www.citforum.ru](http://www.citforum.ru) – материалы сайта Сервер информационных технологий.
23. <http://www.makasin.info/system/files>

5.5. Перечень информационных технологий и программного обеспечения

Используются лицензионное программное обеспечение ОС Windows -/11 и программное

обеспечение открытого доступа (Open source), среды программирования (Denwer, CodeBlock, Dev_C++ и др.). Для разработки моделей проекта ИС используются CASE – средства: ERWin, Visual UML, Rational Rose и т.д.

6. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Студенты, изучающие курс «Проектирование информационных систем», должны в первую очередь обратить внимание на современных подходах разработки информационных систем принципиальную разницу между структурным и объектно-ориентированным подходами. Необходимо больше внимания уделять использованию возможностей CASE – технологий. Четко представлять основные понятия ОПОП. Структурирование программы на модули особого вида, объединяющие данные и процедуры их обработки. Кроме того студенты должны достаточно хорошо владеть нотациями UML. Знать основные его элементы. Представление использования Use-Case-View. Назначение и основные используемые диаграммы. Логическое представление (Logical View) – взгляд на систему изнутри. Логическая структура, основные модули и их алгоритмы реализации. Компонентное представление. Представление взаимодействия процессов и т.д.

Общую схему изучения предмета «Проектирование информационных систем» можно представить в следующем виде:

- Приобретение необходимых знаний по общим методологиям и технологиям проектирования информационных систем
- Приобретение необходимых знаний и навыков по проектированию, разработке и созданию ИС на базе объектно-ориентированных языков
- Приобретение необходимых знаний и навыков по использованию основных элементов UML для проектирования и создания ИС.
- Приобретение необходимых знаний и навыков для тестирования программных продуктов.
- Приобретение необходимых умений документирования и оценки качества программных продуктов.

Самостоятельная работа студентов запланирована в п. 4. данной рабочей программы. Там указаны названия тем, номера заданий, объемы выполняемых работ и формы контроля со стороны преподавателя.

На лабораторных занятиях с использованием средств вычислительной техники студенты выполняют задания, по проектированию отдельных этапов ИС.

Основой обучения являются аудиторские занятия – лекции и лабораторные занятия по выполнению заданий и подготовка и защита курсовой работы. Вся тематика курса условно разбита на 6 основных разделов.

В первом разделе (темы №№ 1-4) рассматривая общие принципы проектирования ИС уделяется внимание следующим вопросам: Основные этапы разработки программного обеспечения (ПО). Детальное проектирование. Инспекция и сквозной просмотр. Граничные условия. Вертикальное и горизонтальное проектирование. Функциональность приложения. Масштабность приложения. Методы проектирования ИС. Концепции и теоретические основы. Нотации, используемые при построении статической структуры и динамического поведения ИС. Процедуры, определяющие практическое применение методов. Классификация методов проектирования ПО: по степени автоматизации, по методологии процесса разработки. Подходы проектирования алгоритмов и программ: структурное проектирование, информационное моделирование, объектно-ориентированное проектирование. Подход Rapid Application Development (RAD). Операционный подход. Декларативный подход. Логическое программирование. Функциональное программирование на языке Лисп и обработка списков.

Второй раздел (тема №5) посвящен вопросам проектирования информационного обеспечения ИС. Изучаются структуры и модели данных. В качестве инструмента рассматривается E-R – диаграммы и понятия сущность, связь, атрибут, экземпляр атрибута. Приводятся сведения о CASE средстве ERWin.

В третьем разделе (темы №№ 6-9) подробно раскрываются методы и технологии структурного и объектно-ориентированного подхода к проектированию ИС. Основные принципы структурного подхода: разделяй и властвуй, иерархическое упорядочивание, абстрагирование, непротиворечивость, структурирование данных. Метод функционального моделирования -

SADT. Состав функциональной модели, построение иерархии диаграмм, типы функции и связи между ними. Моделирование потоков данных. Излагаются принципиальные различия между структурным и объектно-ориентированным подходами. Основные понятия объектно-ориентированного программирования. Определяются понятия ОПОП: объект – предмет или явление, имеющее четко определяемое поведение. Объектно-ориентированное программирование, наиболее популярная в настоящее время парадигма программирования, – развитие структурного программирования. Инкапсуляция. Структурирование программы на модули особого вида, объединяющие данные и процедуры их обработки. Процесс отделения отдельных элементов объекта. Абстрагирование. Клиент. Сервер. Состояние и идентификация объекта. Взаимодействие объекта с окружающим миром - интерфейс объекта. Время жизни объекта. Создание объекта и уничтожение объекта. Действия, выполняемые объектом – методы. Множество объектов, связанных общностью структуры и поведения. Класс. Интерфейс и реализация. Экземпляры класса. Класс как тип данных. Абстрактные и конкретные классы. Базовые классы. Отношения между классами. Метаклассы и метаданные.

Четвертый раздел (тема №10) рассматривает объектные модели такие как OMG, CORBA и др., а также среды объектно-ориентированного программирования. C++, Delphi, Smalltalk. Объявление и создание классов. Свойства и методы класса. Переопределение методов. Область видимости. (Protected, Private, Public, Published, Automated, Friend). Уничтожение класса. Механизм наследования. Наследование как средство специализации. Наследование как способ задания интерфейса. Механизм встраивания. Встраивание одного класса в другой. Полиморфизм. Способность класса принадлежать более чем одному типу. Использование наследования и полиморфизма.

Пятый раздел (темы №№ 11-15) посвящен описанию метода проектирования ИС на базе элементов визуально-графического языка моделирования Unified Modeling Language – UML. Излагаются цели разработки и стандартизация UML. Описываются CASE – средства: Microsoft Visual Modeler и Rational Rose. Нотация UML. Диаграммы UML для моделирования. Варианты использования. Последовательность транзакций. Взаимодействие между компонентами системы и взаимодействие между пользователем и системой. Рассматриваются основные элементы языка. Представление использования Use-Case-View. Назначение и основные используемые диаграммы. Логическое представление (Logical View) – взгляд на систему изнутри. Логическая структура, основные модули и их алгоритмы реализации. Компонентное представление. Представление взаимодействия процессов. Представление распределения. Основные диаграммы UML. Диаграммы вариантов использования Use-Case-Diagram. Диаграммы классов Class Diagram. Диаграммы поведения системы Behavior Diagram. Диаграммы взаимодействия Interaction Diagram. Диаграммы последовательности Sequence Diagram. Кооперативные диаграммы Collaboration diagram. Диаграммы состояний Statechart Diagram. Диаграммы деятельности Activity Diagram. Диаграмма реализации Implementation Diagram. Диаграммы компонентов Component Diagram. Диаграммы размещения Deployment Diagram.

В шестом разделе (темы № 16) рассматриваются принципы и стандарты документирования. Технологическая документация процесса разработки. Типовая структура и содержание базовых комплектов эксплуатационных документов на программы и данные. Документация администрирования при эксплуатации ИС. Особенности использования документов на импортные программные средства. Типовая структура и содержание базовых комплектов технологических документов разработчиков в жизненном цикле прикладных программных средств ИС. Документы при проектировании, испытаний, сопровождении и конфигурационное управление.

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для реализации дисциплины при кафедре информатики и ИС РТСУ имеются 5 компьютерных классов, 2 из которых обеспечены электронными досками. В трех компьютерных классах реализованы облачные технологии на базе блейд-серверной системы.

Для занятий используются лицензионное программное обеспечение ОС Windows -7/8/10 и программное обеспечение открытого доступа (Open source), среды программирования (Dewer, CodeBlock, Dev_C++ и др.). Для разработки моделей проекта ИС используются CASE – средства: ERWin, Visual UML, Rational Rose и т.д.

8. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМО-

СТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

Промежуточная аттестация осуществляется: для зачета – контрольная работа и опрос. Экзамен проводится в форме тестирования. Защита курсового проекта: представляется пояснительная записка и презентация выступления.

Текущий контроль студентов осуществляется путем защиты лабораторных работ, выполнения самостоятельного задания, обсуждения теоретических вопросов

Контролирующие материалы по дисциплине содержат:

Контрольные вопросы и задания для текущего контроля знаний по дисциплине (Приложение 1);

Тестовые задания для промежуточного контроля знаний по дисциплине (Приложение 2);
Методические рекомендации и тематика курсового проектирования. Также указаны критерии оценки курсового проекта (Приложение 3).

Итоговая система оценок по кредитно-рейтинговой системе с использованием буквенных символов

Оценка по буквенной системе	Диапазон соответствующих наборных баллов	Численное выражение оценочного балла	Оценка по традиционной системе
A	10	95-100	Отлично
A-	9	90-94	
B+	8	85-89	Хорошо
B	7	80-84	
B-	6	75-79	
C+	5	70-74	Удовлетворительно
C	4	65-69	
C-	3	60-64	
D+	2	55-59	
D	1	50-54	
Fx	0	45-49	Неудовлетворительно
F	0	0-44	

Содержание текущего контроля, промежуточной аттестации, итогового контроля раскрываются в фонде оценочных средств, предназначенных для проверки соответствия уровня подготовки по дисциплине требованиям ФГОС ВО. ФОС по дисциплине является логическим продолжением рабочей программы учебной дисциплины. ФОС по дисциплине прилагается.