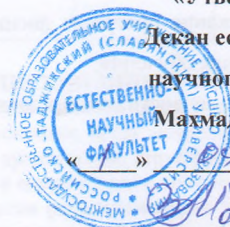


МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ  
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ  
ТАДЖИКИСТАН  
МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКО-ТАДЖИКСКИЙ (СЛАВЯНСКИЙ) УНИВЕРСИТЕТ»

«Утверждаю»

Декан естественно-  
научного факультета  
Махмадбегов Р.С.

2023 г.



**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**  
**ПРОЕКТИРОВАНИЕ ИНФОРМАЦИОННЫХ СИСТЕМ**  
Направление подготовки: 09.03.03- Прикладная информатика  
Профиль подготовки: Инженерия программного обеспечения  
Форма подготовки: очная  
Уровень подготовки: бакалавриат

Душанбе - 2023

Рабочая программа составлена в соответствии с требованиями федерального государственного образовательного стандарта высшего образования, утвержденного приказом Министерства образования и науки РФ № 922 от 19.09.2017 г.

При разработке рабочей программы учитываются

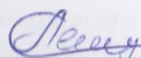
- требования работодателей, профессиональных стандартов по направлению / специальности (при наличии) (для общепрофессиональных и профессиональных дисциплин);
- содержание программ дисциплин/модулей, изучаемых на предыдущих и последующих этапах обучения;
- новейшие достижения в данной предметной области.

Рабочая программа обсуждена на заседании кафедры информатики и ИТ, протокол № 1 от 29 августа 2023 г.

Рабочая программа утверждена УМС естественнонаучного факультета, протокол № 1 от 30 августа 2023 г.

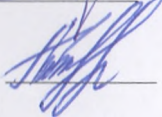
Рабочая программа утверждена Ученым советом естественнонаучного факультета, протокол № 1 от 31 августа 2022 г.

Заведующий кафедрой, к.э.н., доцент



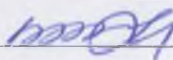
Лешукович А.И.

Зам. председателя УМС факультета, к.э.н.



Абдулхаева Ш.Р.

Разработчик, преподаватель



Каримов М.М.

### Расписание занятий дисциплины

Ф.И.О. преподавателя	Аудиторные занятия		Приём СРС	Место работы преподавателя
	лекция	Практические занятия, КСР, лаб.		
Каримов М. М.				РТСУ, кафедра информатики и ИТ, Корпус 2, 216 каб.

#### 1. ЦЕЛИ И ЗАДАЧИ ИЗУЧЕНИЯ ДИСЦИПЛИНЫ

Под проектированием информационных систем (ИС) понимается процесс создания в рамках обозначений, правил описания и совокупности документации, предусмотренной стандартами проекта ИС, т.е. однозначно понимаемого описания архитектуры, структуры и алгоритма функционирования. Проектирование ИС осуществляется в соответствии с определенной технологией проектирования, согласно масштабу и особенностям разрабатываемого проекта. Технология проектирования – это совокупность методов и средств проектирования и организации ИС, т.е. методы и средства, используемые в процессе создания и модернизации проекта ИС. В основе технологии проектирования лежит технологический процесс, который определяет действия, их последовательность, состав исполнителей, средства и ресурсы, необходимые для выполнения этих действий. Дисциплина «Проектирование информационных систем» изучает способы конструирования программного обеспечения, разработки структуры информационного обеспечения и эффективного распределения ресурсного обеспечения информационных систем с применением методов визуального моделирования на основе концепции объектно-ориентированной и структурной разработки.

##### 1.1. Цели изучения дисциплины

Целью дисциплины является освоение способов конструирования программ с применением методов структурного и объектно-ориентированного программирования, а также изучение методов визуального моделирования проектируемых информационных систем.

##### 1.2. Задачи изучения дисциплины

Задачи дисциплины формулируются в соответствии с требованиями ФГОС, предъявляемые к компетенциям обучающегося. В результате освоения дисциплины «Проектирование информационных систем» формируются следующие компетенции обучающегося:

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	<b>УК-1.1.</b> Знать принципы сбора, отбора и обобщения информации, методики системного подхода для решения профессиональных задач.	Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.
		<b>УК-1.2.</b> Уметь анализировать и систематизировать разнородные данные, оценивать эффективность процедур анализа проблем и принятия решений в профессиональной деятель-	

		ности.	
		<b>УК-1.3.</b> Владеть навыками научного поиска и практической работы с информационными источниками; методами принятия решений.	
<b>ОПК-4</b>	Способностью участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	<b>ОПК-4.1.</b> Знать основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.	Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.
		<b>ОПК-4.2.</b> Уметь применять стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы.	
		<b>ОПК-4.3.</b> Владеть навыками составления технической документации на различных этапах жизненного цикла информационной системы.	
<b>ОПК-6</b>	Способен анализировать и разрабатывать организационно-технические и экономические процессы с применением методов системного анализа и математического моделирования	<b>ОПК-6.1.</b> Знать основы теории систем и системного анализа, дискретной математики, теории вероятностей и математической статистики, методов оптимизации и исследования операций, нечетких вычислений, математического и имитационного моделирования.	Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.
		<b>ОПК-6.2.</b> Уметь применять 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.	
		<b>ОПК-6.3</b> <b>Владеет</b> навыками проведения инженерных расчетов основных показателей результативности создания и применения информационных систем и технологий.	

<b>ОПК-8</b>	Способностью принимать участие в управлении проектами создания информационных систем на стадиях жизненного цикла	<b>ОПК-8.1.</b> Знать основные технологии создания и внедрения информационных систем, стандарты управления жизненным циклом информационной системы.	Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.
		<b>ОПК-8.2.</b> Уметь осуществлять организационное обеспечение выполнения работ на всех стадиях и в процессах жизненного цикла информационной системы.	
		<b>ОПК-8.3.</b> Владеть навыками составления плановой и отчетной документации по управлению проектами создания информационных систем на стадиях жизненного цикла.	
<b>ОПК-9</b>	Способностью принимать участие в реализации профессиональных коммуникаций с заинтересованными участниками проектной деятельности и в рамках проектных групп	<b>ОПК-9.1.</b> Знать 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.	Тестирование. Контроль самостоятельной работы. Отчеты по практическим работам. Контрольная работа. Устный опрос.
		<b>ОПК-9.2.</b> Уметь осуществлять взаимодействие с заказчиком в процессе реализации проекта; принимать участие в командообразовании и развитии персонала.	
		<b>ОПК-9.3.</b> Владеть навыками проведения презентаций, переговоров, публичных выступлений.	

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП

Дисциплина «Проектирование информационных систем» изучает способы конструирования программного обеспечения, разработки структуры информационного обеспечения и распределения ресурсного обеспечения информационных систем с применением методов визуального моделирования. Она является обязательной дисциплиной (Б1.О.26), изучается в 5 и 6 семестрах. Логически и содержательно-методически взаимосвязана с дисциплинами ОПОП, указанных в таблице 1:

Таблица 1.

№	Название дисциплины	Семестр	Место дисциплины в структуре ОПОП
1.	<i>Программирование</i>	2,3	<i>Б1.О.06</i>
2.	<i>Математика</i>	1-2	<i>Б1.О.07</i>
3.	<i>Основы алгоритмизации и языки программирования</i>	3-4	<i>Б1.В.10</i>

4.	<i>Базы данных</i>	3-4	<i>Б1.О.15</i>
5.	<i>Программная инженерия</i>	4-5	<i>Б1.О.17</i>
6.	<i>Интеллектуальные информационные системы</i>	5	<i>Б1.В.12</i>
7.	<i>Управление программными проектами</i>	8	<i>Б1.В.20</i>
8.	<i>Технологическая (проектно-технологическая) практика</i>	6	<i>Б2.О.02(П)</i>
9.	<i>Преддипломная практика</i>	8	<i>Б2.В.01(Пд)</i>

При освоении данной дисциплины необходимы умения и готовность («входные» знания) обучающегося по дисциплинам 1-5, указанных в Таблице 1. Дисциплины 7- 8 относятся к группе «входных» знаний, вместе с тем определенная ее часть изучается параллельно с данной дисциплиной («входные-параллельные» знания). Теоретическими дисциплинами и практиками, для которых освоение данной дисциплины необходимо как предшествующее являются: 9-10.

### 3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Объем дисциплины «Проектирование информационных систем» составляет 8 зачетных единиц, всего 288 часов, из которых: лекции – 26 часов, практические занятия – 26 часов, лабораторные работы – 26 часа, КСР – 26 часов, всего часов аудиторной нагрузки - 104 часов, в том числе в интерактивной форме 42 часа, самостоятельная работа – 130 часа.

Зачет – 5-й семестр

Экзамен – 6-й семестр

Защита курсового проекта – 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**

**Тема 9. Отношения между классами. Ассоциация. Агрегация и композиция. Обобщение/Расширение (наследование)**

#### 6-й семестр

**Тема 10. 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 11. 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 12. Представления (Виды, Views) UML**

**Тема 13. 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 14. 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 15. Диаграммы для описания динамического поведения ИС.**

Диаграммы прецедентов использования (Use Case); деятельности (Activity); конечных автоматов (State Machine); Описывает функциональные возможности системы. Изображаются внешне *субъекты* (actors) и их связь с *асpekтами 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 посылки 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

### **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 (26 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).

##### **6-й семестр**

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

#### **Лабораторные работы (26 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).

6-й семестр

- Лабораторная работа №1. Введение в UML (2 часа).  
 Лабораторная работа №2. Диаграмма Use-Case (2 часа).  
 Лабораторная работа №3. Диаграмма последовательности. (2 часа).  
 Лабораторная работа №4. Диаграмма развертывания. (2 часа).  
 Лабораторная работа №5. Диаграмма активности (2 часа).

### 3.3 Структура и содержание КСР (26 часов)

5-й семестр

- КСР-1. Основные этапы разработки программного обеспечения (ПО). (2 часа).  
 КСР-2. Жизненный цикл программ (2 часа).  
 КСР-3. Построение модели. Дедуктивный подход. Индуктивный подход. (2 часа).  
 КСР-4. Инспекция и сквозной просмотр. Граничные условия. (2 часа).  
 КСР-5. Методы проектирования ПО. (2 часа).  
 КСР-6. Структурное проектирование. (2 часа).  
 КСР-7. Информационное моделирование (2 часа).  
 КСР-8. Объектно-ориентированное проектирование. (2 часа).

6-й семестр

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

№ п/п	Раздел дисциплины	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)				Литература	Кол-во баллов в неделю
		Лек. 18	Лаб18.	Пр. 18	КСР18		
<b>5 семестр</b>							
1.	<b>Тема 1. Основы проектирования программного обеспечения</b> Современный подход. Декомпозиция задачи. Абстракция через параметризации. Основные этапы разработки программного обеспечения (ПО). Детальное проектирование. Построение модели. Дедуктивный подход. Индуктивный подход. Инспекция и сквозной просмотр. Граничные условия. <b>Лабораторная работа №1.</b> Технология проектирования ИС	2	2			1(5-15) 5 (с.11-16), 11 (с.31-49) 16 (с.19-33)	11, 5
2	<b>КСР-1.</b> Основные этапы разработки программного обеспечения (ПО).			2	2	4(с.46-51) 3(с.92-93)	11, 5

	<b>ПР1.</b> Основные этапы разработки программного обеспечения. Построение модели.						
3	<b>Тема 2. Методы и технологии проектирования ИС. Классификация методов проектирования.</b> Методы проектирования ИС. Концепции и теоретические основы. Нотации, используемые при построении статической структуры и динамического поведения ИС. Процедуры, определяющие практическое применение методов. Оценка результатов. Технология проектирования для реализации методов. Классификация методов проектирования ПО: по степени автоматизации, по методологии процесса разработки. Подходы проектирования алгоритмов и программ: структурное проектирование, информационное моделирование, объектно-ориентированное проектирование. <b>Лабораторная работа №2.</b> Методы проектирования ИС	2	2			1(15-24) 11(с.65-79) 16(с.36-51) 17(с.45-69)	11, 5
4	<b>КСР-2.</b> Жизненный цикл программы. <b>ПР2.</b> Методы и технологии проектирования ИС. Классификация методов проектирования.			2	2	4(с.51-58) 3(с.94-96)	11, 5
5	<b>Тема 3. Методы быстрой разработки ПО. Подход RAD</b> Подход Rapid Application Development (RAD). Основные составляющие подхода: небольшая группа, производственный график, взаимодействие с заказчиком. Функциональные точки. Ограничения в применении подхода RAD <b>Лабораторная работа №3.</b> Модели жизненного цикла	2	2			1(с.32-36) 5(с.213-220) 11(с.97-120)	11, 5
6	<b>КСР-3.</b> Построение модели. Дедуктивный подход. Индуктивный подход. <b>ПР3.</b> Методы быстрой разработки ПО. Подход RAD			2	2	4(с.99-102) 1(с.36-37)	11, 5
7.	<b>Тема 4. Проектирование структуры программного обеспечения. Операционный и декларативный подходы.</b> 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 <b>Лабораторная работа №4.</b> Методы быстрой разработки ПО - Подход RAD	2	2			1(с.37-39) 5(с.37-91) 11(с.97-120)	11, 5
8	<b>КСР-4.</b> Инспекция и сквозной просмотр. Граничные условия. <b>ПР4.</b> Проектирование структуры программного обеспечения. Обработка списков			2	2	5(с.91-93) 1(с.115-116)	11, 5

9	<p><b>Тема 5. Проектирование информационного обеспечения ИС</b>          Описание предметной области. Понятие «предметная область». Объектное ядро предметной области. Траектории предметной области. Семантика предметной области. Информационно-логическое (инфологическое) моделирование предметной области. Абстрактность средства описания предметной области. Концептуальная схема проектирования БД. Понятие типа и объединение объектов.          Фазы методологии проектирования баз данных. Процесс конструирования общей программной модели базы данных на основе отдельных моделей данных, которая является независимой от особенностей реально используемой СУБД и других физических условий. Этапы проектирования. Выделение сущностей и связей между ними, построение диаграмм ER – типа с учетом всех сущностей и их связей, формирование набора предварительных отношений с указанием предполагаемого первичного ключа для каждого отношения с использованием ER – диаграмм. Использование CASE – средства для проектирования базы данных (ErWin)  <b>Лабораторная работа №5.</b>          Проектирование структуры программного обеспечения</p>	2	2			1(с.50-59) 21(с.149-220)	8
10	<p><b>КСР-5. Методы проектирования ПО.</b>  <b>ПР5.</b> Проектирование информационного обеспечения ИС. Моделирование данных</p>			2	2	5(с.162, 178, 221) 1(с.116-117)	11, 5
11	<p><b>Тема 6. Структурный подход.</b>          Структурный подход. 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. Моделирование данных: сущность, связь, атрибут, экземпляр атрибута.  <b>Лабораторная работа №6.</b>          Моделирование потоков данных</p>	2	2			1(с.39-49) 5(с.316-366) 11(с.135-154) 16(с.224-241)	11, 5
12	<p><b>КСР-6. Структурное проектирование.</b>  <b>ПР6.</b> Моделирование потоков данных.</p>			2	2	5(с.371-374) 1(с.118-119)	11, 5
13	<p><b>Тема 7. Объектно-ориентированный подход.</b>          Принципиальное различие между структурным и объектно-ориентированным подходом. Основные понятия объектно-ориентированного программирования. Языки объектно-</p>	2	2			1(с.64-82) 8(с.57-105) 17(с.34-38)	11, 5

	<p>ориентированного программирования. C++, Delphi, Smaltalk. Объект – предмет или явление, имеющее четко определяемое поведение. Объектно-ориентированное программирование, наиболее популярная в настоящее время парадигма программирования, – развитие структурного программирования. Основные понятия объектно-ориентированного 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</p> <p><b>Лабораторная работа №7.</b> Структурный подход</p>						
14	<p><b>КСР-7. Информационное моделирование ПР7 Структурный подход.</b></p>			2	2	17(с.39-41) 2(с.10-11)	11, 5
15	<p><b>Тема 8. Классы. Создания классов. Наследование, встраивание и полиморфизм</b> Множество объектов, связанных общностью структуры и поведения. Класс. Интерфейс и реализация. Экземпляры класса. Класс как тип данных. Абстрактные и конкретные классы. Базовые классы. Метаклассы и метаданные. 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.</p> <p><b>Лабораторная работа №8.</b> Метод SADT</p>	2	2			1(с.83-96) 8(с.106-125) 17(с.42-75)	11, 5
16	<p><b>КСР-8.Объектно-ориентированное проектирование.</b> <b>ПР8. Метод функционального моделирования – SADT</b></p>			2	2	13(с.47-56) 17(с.76-85) 2(с.12-13)	11, 5
<b>Итого:</b>		<b>16</b>	<b>16</b>	<b>16</b>	<b>16</b>		

<b>6 семестр</b>							
1	<p><b>Тема 10. Объектные модели</b> Объектная модель Object Management Group (OMG). Модель объектно-ориентированных систем управления базами данных – Object Data-</p>	2	2		2	1(с.97-123) 6(с.452-478)	18

	base Management Group. Основные этапы создания информационных 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 <b>Лабораторная работа №1. Введение в UML</b>					
2	<b>Лабораторная работа №2. Диаграмма Use-Case</b> <b>ПР1.</b> Основные понятия объектно - ориентированного подхода	2	2		1(с.124-126) 6(с.479-482) 2(с.16-17)	18
3.	<b>Тема 11. Универсальный язык моделирования</b> 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	2		8 (с.175-186) 17(с.17-24)	10
4	<b>Лабораторная работа №3. Диаграмма последовательности</b>		2		17(с.17-24) 2(с.21-22)	18
5	<b>Тема 12. Представления (Виды, Views) UML</b> 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. Типы связи: один к одному, один ко многим и много ко многим. Представление распределения <b>ПР2.</b> Классы. Создания классов. Наследование, встраивание и полиморфизм	2	2		8 (с.186-195) 10(с.584-601) 17(с.34-42)	18
6	<b>Лабораторная работа №4. Диаграмма развертывания.</b> <b>КСР-1.</b> Диаграммы коммуникации и последовательности		2	2	13(с.26-33) 2(с.25-26)	18
7	<b>Тема 13. Диаграммы UML</b> 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	2	2		8 (с.186-275) 13(с.38-45) 2(с.25-26)	18

	мы деятельности Activity Diagram. Диаграмма реализации Implementation Diagram. Диаграммы компонентов Component Diagram. Диаграммы размещения Deployment Diagram						
8	Лабораторная работа №5. Диаграмма активности ПР3. Отношения между классами		2	2		13(с.26-33) 2(с.25-26)	18
9	Тема 14. 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. ПР4. Объектные модели UML	2		2		13(с.47-52) 17(с.110-115) 2(с.29-30)	18
10	Лабораторная работа №6. Диаграмма классов КСР2. Диаграмма активности		2		2	8 (с.123-148) 17(с.42-109)	18
<b>Итого по семестру</b>		<b>10</b>	<b>10</b>	<b>10</b>	<b>10</b>		

#### Формы контроля и критерии начисления баллов

В случае пропуска студентом занятий по уважительной причине (при наличии подтверждающего документа) в период академической недели, деканат факультета обращается к профессору 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 проводится в традиционной устной форме на бумажном носителе с выставлением оценки в ведомости.

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	Административный балл за примерное поведение	Балл за рубежный и итоговый контроль	Всего
1	2	3	4	5	6	7
1	3,5	4	4	-	-	11,5
2	3,5	4	4	-	-	11,5
3	3,5	4	4	-	-	11,5
4	3,5	4	4	-	-	11,5

5	3,5	4	4	-	-	11,5
6	3,5	4	4	-	-	11,5
7	3,5	4	4	-	-	11,5
8	3,5	4	4	-	-	11,5
9(первый рубеж- ный кон- троль)					8	8
<b>Первый рейтинг</b>				-	8	100/2*0 .49
10	3,5	4	4	-	-	11,5
11	3,5	4	4	-	-	11,5
12	3,5	4	4	-	-	11,5
13	3,5	4	4	-	-	11,5
14	3,5	4	4	-	-	11,5
15	3,5	4	4	-	-	11,5
16	3,5	4	4	-	-	11,5
17	3,5	4	4	-	-	11,5
18 (второй рубежный кон- троль)					8	8
<b>Второй рейтинг</b>					8	100/2*0 .49
<b>ИТОГОВЫЙ КОНТРОЛЬ (зачет, зачет с оценкой, экзамен)</b>					100	100/2*0 .51
<b>ИТОГО:</b>	<b>56</b>	<b>64</b>	<b>64</b>		<b>16+100</b>	<b>300/100</b>

#### 4. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

##### 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 технологий разработки, реферат	Опрос
2.	4	Методы и технологии проектирования ИС. Классификация методов проектирования.	Вопросы 5-8. Презентация методов	Выступление
3.	6	Методы быстрой разработки ПО. Подход RAD	Вопросы 8-10. Презентация, доклад	Выступление
4.	6	Проектирование структуры программного обеспечения. Обработка списков	Вопросы 11-13. Выполнение зада-	Защита работы. Вы-

			ния 1 (1-10).	ступление
5.	4	Проектирование информационного обеспечения ИС. Моделирование данных	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.	6	Структурное программирование	Выполнение задания 5	Защита работы.
10.	6	Основные понятия объектно - ориентированного подхода	Вопросы 18-25. Выполнение задания 6	Защита работы.
11.	6	Классы. Создания классов. Наследование, встраивание и полиморфизм	Вопросы 26-29. Выполнить задания 2 и описать в терминах классов.	Опрос. Защита работы
12.	6	Отношения между классами	Вопросы 30-31. Реферат. Выполнение задания 7	Защита реферата. Защита работы
13.	6	Объектные модели	Вопросы 32-37. Презентация	Опрос. Выступление
14.	6	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.	6	Основные диаграммы UML. (Use-Case)	Вопросы 50-51. Выполнение задания 10	Защита работы
18.	6	Статическая структура ИС. Структурные диаграммы. Диаграммы пакетов (Package); классов (Class); объектов (Object); компонентов (Component); распределения (Развертывания Deployment); композитных структур (Composite structure).	Вопросы 52-54. Выполнение задания 11	Защита работы
19.	6	Диаграммы для описания динамического поведения ИС. Диаграммы прецедентов использования.	Вопросы 55-59. Выполнение задания 11	Защита работы
20.	6	Диаграммы реализации, диаграммы размещения, диаграммы распределения, диаграммы коммуникации, диаграммы	Вопросы 60-62. Выполнение задания 12	Защита работы



		обзора взаимодействий и синхронизации		
21.	6	Диаграммы деятельности (Activity).	Вопросы 63-64. Выполнение задания 13	Защита работы
22.	6	Диаграммы состояний	Вопросы 65-66. Выполнение задания 14	Защита работы
23.	6	Конечных автоматов (State Machine)	Вопросы 67-68. Выполнение задания 15	Защита работы
24.	6	Принципы и стандарты документирования	Вопросы 69-74. Презентация, доклад	Выступление
	132			

#### 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 3 «Содержание и структура дисциплины». Большинство заданий выполняются в виде разработки модели отдельных аспектов проектируемой системы, некоторые задания необходимо программировать.

Отчет по разработке модели отдельных аспектов содержит следующие разделы:

1. Стандартный титульный лист
2. Отчет о решенных задачах (с указанным номером варианта):
  - постановку задачи;
  - описание задачи и инструментов для ее решения (например диаграмма SADT, UML и .т.п);
  - схематическое представление;
  - выводы.

#### 4.3. Требования к представлению и оформлению результатов самостоятельной работы;

При выполнении самостоятельной работы студент должен предварительно изучить теоретические сведения по используемым методам для решения соответствующих задач. По выполнению индивидуального домашнего задания по практическим занятиям студенты должны представить отчеты в соответствии с содержанием, приведенным в пункте 4.2, которые должны быть представлены преподавателю.

### 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 С#. 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.: BHV, 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> – 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 подходах. Необходимо больше внимание уделять использованию возможностей CASE –технологии. 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, COR-BA и др., а также среды о 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 взаимодействии 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 конфигурационном управлении.

#### **7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Для реализации дисциплины при кафедре информатики и ИС РГСУ 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 (Deweg, CodeBlock, Dev\_C++ и др.). Для разработки моделей проекта ИС используются CASE – средства: ERWin, Visual UML, Rational Rose и т.д.

В Университете созданы специальные условия обучающихся с ограниченными возможностями здоровья – специальные учебники, учебные пособия и дидактические материалы, специальные технические средства обучения коллективного и индивидуального пользования, предоставление услуг ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, проведение групповых и индивидуальных коррекционных занятий, обеспечение доступа в здания организаций и другие условия. без которых невозможно или затруднено освоение дисциплины обучающимися с ограниченными возможностями здоровья.

Обучающимся с ограниченными возможностями здоровья предоставляются бесплатно специальные учебники и учебные пособия, иная учебная литература, а также обеспечивается:

наличие альтернативной версии официального сайта организации в сети "Интернет" для слабовидящих;

присутствие ассистента, оказывающего обучающемуся необходимую помощь;

обеспечение выпуска альтернативных форматов печатных материалов (крупный шрифт или аудиофайлы);

возможность беспрепятственного доступа обучающихся в учебные помещения, столовые, туалетные и другие помещения организации, а также пребывания в указанных помещениях (наличие пандусов, поручней, расширенных дверных проёмов, лифтов).

## **8. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, 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	Неудовлетворительно
Fx	0	45-49	
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.*