

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РЯЗАНСКИЙ ГОСУДАРСТВЕННЫЙ РАДИОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ИМЕНИ В.Ф. УТКИНА»

Кафедра «Вычислительная и прикладная математика»

«СОГЛАСОВАНО»
Директор ИМиА

«УТВЕРЖДАЮ»
Проректор РОПиМД

_____/ Бодров О.А.
«__» _____ 2020 г.

_____/ Корячко А.В.
«__» _____ 2020 г.

Заведующий кафедрой

_____/ Овечкин Г.В.
«__» _____ 2020 г.

Руководитель ОПОП

_____/ Пылькин А.Н.
«__» _____ 2020 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

**Б1.О.05 «МЕТОДЫ И СРЕДСТВА ПРОЕКТИРОВАНИЯ И РАЗРАБОТКИ
ПРОГРАММНЫХ СИСТЕМ»**

Направление подготовки
09.04.04 Программная инженерия

Направленность (профиль) подготовки
Разработка программно-информационных систем

Уровень подготовки
магистратура

Квалификация выпускника – магистр

Формы обучения – очная

Рязань 2020 г.

ЛИСТ СОГЛАСОВАНИЙ

Рабочая программа составлена с учетом требований федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.04.04 «Программная инженерия», утвержденного приказом Минобрнауки России от 19.09.2017 № 932.

Разработчик:
проф. каф. ВПИМ

_____ Белов В.В.

Программа рассмотрена и одобрена на заседании кафедры ВПИМ

«11» июня 2020 г., протокол № 11

Заведующий кафедрой ВПИМ
д.т.н., проф.

_____ Овечкин Г.В.

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины Б1.О.05 «Методы и средства проектирования и разработки программных систем» является: формирование у обучаемых компетенций в соответствии с Федеральным государственным образовательным стандартом, способности к логическому мышлению, анализу, восприятию и применению информации, воспитание профессиональной культуры, посредством формирования знаний, умений и навыков предусмотренных ФГОС, в части представленных ниже.

Основные задачи освоения учебной дисциплины: изучить современные направления, стандарты, инструменты проектирования, методика организации и управления проектированием, образцы проектирования; сформировать умения организовывать процесс проектирования и управления им, осуществлять создание детальных моделей программных систем в соответствии с требованиями качества и рекомендациями по проектированию; овладеть навыками планирования и управления проектированием, выбора архитектуры и использования имеющихся образцов проектирования, создания качественных детальных моделей программных систем, использования инструментов, поддерживающих работы по проектированию

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина Б1.О.05 «Методы и средства проектирования и разработки программных систем» относится к дисциплинам обязательной части Блока № 1 «Дисциплины (модули)» основной профессиональной образовательной программы (далее – образовательной программы) магистратуры «Программная инженерия» направления 09.04.04 «Программная инженерия». Дисциплина изучается во 2-м семестре 1-го курса.

Дисциплина базируется на знаниях, полученных при изучении дисциплин Б1.О.04 «Теория систем и системный анализ», Б1.О.10 «Методология программной инженерии», Б1.В.07 «Архитектура промышленных программных систем», устанавливаемыми ФГОС 3++ для высшего образования.

Для освоения дисциплины обучающиеся должны

знать: основные направления, проблемы, теории и методы современной математики, в том числе основы математического анализа, линейной алгебры, дискретной математики и математической логики, основные понятия информатики, используемые при проектировании и разработке программ,

уметь: решать стандартные задачи математического анализа, линейной алгебры, дискретной математики и математической логики и применять идеи, использованные в их решениях, для решения аналогичных задач, составлять решения задач на алгоритмических языках, включая C++ и/или C#,

владеть: навыками решения задач из следующих разделов современной математики: математический анализ, линейная алгебра и аналитическая геометрия, математическая логика, дискретная математики, знаниями основных алгоритмов при работе с массивами, списками и деревьями, навыками разработки программ и программных систем с использованием баз данных; программирования на языке высокого уровня типа C++ и/или C#.

Полученные знания используются далее, при выполнении НИР и при подготовке выпускной квалификационной работы.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
УК-3. Способен организовывать и руководить работой команды,	ИД-1 _{УК-3} Знать: методика формирования команд; методы эффек-

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
вырабатывая командную стратегию для достижения поставленной цели	<p>тивного руководства коллективами. ИД-2_{УК-3} Уметь: разрабатывать командную стратегию; организовывать работу коллективов; управлять коллективом; разрабатывать мероприятия по личностному, образовательному и профессиональному росту. ИД-3_{УК-3} Владеть методами организации и управления коллективом, планированием его действий.</p>
ОПК-2. Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	<p>ИД-1ОПК-2 Знать: современные интеллектуальные технологии для решения профессиональных задач; ИД-2ОПК-2 Уметь: обосновывать выбор современных интеллектуальных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач ИД-3ОПК-2 Владеть: иметь навыки разработки оригинальных программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач</p>
ОПК-5. Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем	<p>ИД-1ОПК-5 Знать: современное программное и аппаратное обеспечение информационных и автоматизированных систем ИД-2ОПК-5 Уметь: модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач; ИД-3ОПК-5 Владеть: иметь навыки разработки программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач.</p>

В результате изучения дисциплины студенты должны

знать: основные понятия, используемые при проектировании и создании приложений, программных систем (ПС), включая лучшие практики разработки программного обеспечения (ПО); основы визуального моделирования при проектировании и разработке архитектуры ПО, способы преобразования требований к ПС на основе сценариев использования в архитектурные представления ПО; основные задачи, выполняемые архитектором и разработчиком при проектировании по согласно методологии IBM RUP;

уметь: строить проектные модели ПО и ПС с использованием диаграмм визуальной нотации UML (Унифицированный язык моделирования); проектировать логическое представление архитектуры на UML; пошагово строить на UML диаграммы классов, представляющие модель предметной области и ключевые абстракции проекта; с использованием паттернов трансформировать модель предметной области; идентифицировать и разрабатывать спецификации сценариев использования ПС для создания классов-участников реализации; использовать диаграммы взаимодействия, состояний и активности при проектировании ПО;

владеть: навыками (приобрести опыт) пошагового построения проекта ПО как последовательного набора UML диаграмм в инструментальном CASE средстве; реализации лучших практик разработки программного обеспечения, включая построения компонентной архитектуры ПО и применение архитектурных механизмов анализа; использования визуальных диаграмм UML при проектировании архитектурных представлений.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Объем дисциплины по семестрам и видам занятий в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 4 зачетных единицы (ЗЕ), 144 часов.
Дисциплина читается на 1-м курсе во 2-м семестре

Объем дисциплины	Всего часов
Общая трудоемкость дисциплины, в том числе:	144
1. Контактная работа обучающихся с преподавателем (всего)	58,65
В том числе:	
Лекции	16
Лабораторные работы (ЛР)	16
Практические занятия (ПЗ)	24
Иная контактная работа (ИКР)	0,65
Консультация	2
2. Самостоятельная работа (СР)	34,3
3. Курсовой проект	15,7
4. Контроль	35,35
Вид промежуточной аттестации	Экзамен, КП

4.2 Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)

№	Раздел дисциплины	Общая трудоемкость, всего часов	Контактная работа обучающихся с преподавателем						Самостоятельная работа обучающихся	Контроль
			Всего	Лекции	ПЗ	ЛР	ИКР	Конс.		
Семестр 2										
1	Основы проектирования программных систем (ПС)	8	6	2	2	2			2	
2	Формирование технического задания на создание ПС	10	6	2	2	2			4	
3	Каноническое проектирование ПС	10	6	2	2	2			4	

4	Функционально-ориентированное проектирование ПС.	12	8	2	4	2			4	
5	Объектно-ориентированное проектирование ПС	12	8	2	4	2			4	
6	Проектирование обеспечивающих подсистем	12	8	2	4	2			4	
7	Индустриальное проектирование ПС	14	8	2	4	2			6	
8	Организация и управление проектом ПС на всех стадиях ЖЦ.	12,3	6	2	2	2			6,3	
9	Курсовое проектирование	16	0,3				0,3		15,7	
10	Экзамен	37,7	2,35				0,35	2		35,35
11	Всего	144	58,65	16	24	16	0,65	2	50	35,35

4.3. Содержание дисциплины

4.3.1. Лекционные занятия

№	Темы лекционных занятий	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
1	Основы проектирования программных систем (ПС).	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект
2	Формирование технического задания на создание ПС.	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект
3	Каноническое проектирование ПС.	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект
4	Функционально-ориентированное проектирование ПС.	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект
5	Объектно-ориентированное проектирование ПС.	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект
6	Проектирование обеспечивающих подсистем.	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект
7	Индустриальное проектирование ПС.	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект
8	Организация и управление проектом ПС на всех стадиях ЖЦ.	2	УК-3, ОПК-2, ОПК-5	Экзамен, курс. проект

Содержание лекций, структурированное по темам

Раздел 1. Основы проектирования программных систем (ПС).

Тема 1.1. Назначение и виды ПС. Понятие ПС. Классификация ПС по различным признакам. Классификация ПС предприятия. Структура ПС. Состав функциональных и обеспечивающих подсистем. Функциональные подсистемы ПС. Обеспечивающие подсистемы ПС. Проектирование ПС: быстрый взгляд. Инвариантные составляющие жизненного цикла ПС.

Тема 1.2. Модели и процессы жизненного цикла (ЖЦ) ПС. Понятие модели жизненного цикла ПС. Процессы и этапы жизненного цикла ПС. Процессы жизненного цикла согласно ISO/IEC 12207 (основные, вспомогательные, организационные). Процессы жизненного цикла согласно ISO/IEC 15288 (договорные, процессы предприятия, проектные, технические, спе-

циальные). Стадии создания ПС. Стадии создания автоматизированных систем по ГОСТ 34.601 – 90. Стадии создания ИС согласно ISO/IEC 15288.

Тема 1.3. Методологии и технологии создания ПС. Основные понятия. Классификация технологий проектирования ПС.

Раздел 2. Формирование технического задания на создание ПС.

Тема 2.1. Методы анализа предметной области. Методы структурного анализа предметной области. Методы объектно-ориентированного анализа предметной области. Современное моделирование бизнес-процессов как средство автоматизации изложения требований. Определение бизнес-процесса.

Тема 2.2. Современная концепция процессного управления. Нотация BPMN (Business Process Modeling Notation). Элементы BPMN. Применение BPMN. Прототип системы как механизм поддержки процесса формирования технического задания. Методы анализа информационных потребностей.

Раздел 3. Каноническое проектирование ПС.

Тема 3.1. Стадии и состав работ канонического проектирования ПС. Понятие канонического проектирования. Стадии создания ПС. Содержание документов «Технико-экономическое обоснование проекта» и «Техническое задание на создание ПС». Практически сложившаяся система стадий и этапов проекта ПС. Эскизный и технический проекты. Рабочий проект.

Тема 3.2. Классификация объектов в ПС. Классификация объектов в ПС. Первичные определения. Типы систем классификации. Иерархическая система классификации. Фасетная система классификации. Выбор типа системы классификации. Дескрипторная система классификации. Система кодирования. Общие понятия. Классификационное кодирование. Регистрационное кодирование.

Тема 3.3. Классификация видов информации в ПС. Основные признаки информации, используемой в ПС. Элементы иерархичности в фасетной классификации информации. Классификационное кодирование видов информации в ПС.

Раздел 4. Функционально-ориентированное проектирование ПС.

Тема 4.1 Методологии SADT и IDEF0. Общие сведения о функционально-ориентированном проектировании. Терминология. Цель моделирования предметной области. Методологии и инструментальные средства моделирования. Методология SADT и семейство стандартов IDEF. Историческая справка. Стандарты IDEF. Методология моделирования IDEF0. Модель IDEF0. Область моделирования (Scope). Цель моделирования (Purpose). Точка зрения (Viewpoint). Типы диаграмм IDEF0. Деятельности (работы) в IDEF0-диаграммах. Стрелки в IDEF0-диаграммах. Диаграмма декомпозиции и ICOM-коды. Словарь стрелок. Система классификаторов. Связывание стрелок при декомпозиции работ. Внутренние связи функциональных блоков. Разветвления и параметры стрелок. Слияние, туннелирование стрелок. Противоречие туннелирования стрелок базовому принципу IDEF0. Идентификация функциональных блоков и диаграмм. Общие рекомендации по конструированию диаграмм.

Тема 4.2. Описание потоков данных. Назначение DFD. Элементы диаграмм потоков данных. Идентификация элементов DFD. Пример диаграммы в нотации DFD. Рекомендации по оформлению DFD.

Тема 4.3. Описание процессов и структуры системы. Описание процессов в нотации IDEF3. Назначение IDEF3-описаний. Элементы IDEF3-диаграмм. Парность перекрестков в IDEF3-диаграммах. Классы моделей: AS-IS, TO-BE, SHOULD-BE. Функциональные спецификации. Понятие функциональной спецификации. Стандарт IEEE 830. Описание структуры системы. Уровни описания структуры ПС. Структура бизнес-функций. Представление архитектурных решений.

Раздел 5. Объектно-ориентированное проектирование ПС.

Тема 5.1. Язык UML. Назначение UML. История UML. Основные изобразительные средства. Понятие заинтересованных лиц и детальности проекта. Архитектурные паттерны и их использование в проектировании. Использование модели предметной области в задаче.

Связи между пакетами UML и их использование в представлении архитектуры. Диаграмма классов (Class diagram). Диаграмма прецедентов (Use case). Диаграмма деятельности (Activity diagram). Диаграмма коммуникации (Communication diagram). Диаграмма последовательности (Sequence diagram). Диаграмма компонентов (Component diagram). Диаграмма развёртывания (Deployment diagram).

Тема 5.2. Представление проектных решений в виде диаграмм UML. Бизнес-моделирование, формулировка требований, проектирование, реализация. От бизнес-модели к системной диаграмме прецедентов. Разработка диаграммы классов. Правила использования стереотипов «boundary» «control» «entity». Разработка моделей базы данных. Представление физической реализации системы. CASE-средства для объектно-ориентированного проектирования. Исторический лидер объектно-ориентированных CASE-средств. Альтернативы Rational Rose.

Тема 5.3. Семантическая целостность UML-модели и семантическое единство UML-диаграмм. Понятия семантической целостности UML-модели и семантического единства UML-диаграмм. Смысловое единство пар диаграмм: прецедентов – классов; прецедентов – деятельности/автоматов; прецедентов – последовательности/коммуникаций; последовательности – классов. Отображение элементов диаграмм модели проектирования в элементы диаграмм модели реализации.

Раздел 6. Проектирование обеспечивающих подсистем. Состав обеспечивающих подсистем. Технологии разработки программного обеспечения ПС. Основные понятия. Технология RUP (Rational Unified Process). Технология Oracle. Технология Borland. Технология Computer Associates. Технология Microsoft Solution Framework (MSF). Проектирование информационного, технического и организационного обеспечения. Требования к математическому и лингвистическому обеспечению.

Раздел 7. Индустриальное проектирование ПС.

Тема 7.1. Прототипное и типовое проектирование. Прототипное проектирование ПС. RAD-технология. Фазы RAD-разработки. Ограничения на применение RAD. Инструментальные средства RAD-технологии. Типовое проектирование ПС. Концепция типового проектирования. Параметрически-ориентированное проектирование. Модельно-ориентированное проектирование.

Тема 7.2. Процессно-ориентированное проектирование ПС. Процессно-ориентированное проектирование ПС. Специфика процессно-ориентированного проектирования. Концепция сервисно-ориентированной архитектуры ПС (SOA). BPEL – инструмент оркестровки бизнес-процессов. От BPMN к BPEL. Интеграция приложений: оркестровка и хореография.

Раздел 8. Организация и управление проектом ПС на всех стадиях ЖЦ.

Тема 8.1. Организация системы управления IT-проектами. Схема организации работ по проектированию ПС. Оценка затрат проекта программной системы. Цели оценки затрат. Стоимость времени и структура затрат в современных индустриальных проектах. Цена проекта создания (приобретения) ПС. Основы менеджмента качества ПС. Организация системы управления IT- проектами. Проблемы: дефицит сроков, фондов и компетенций. Стандарты и модели управления жизненным циклом ПС. Онтологии как средство формализации знаний в системах управления IT-проектами.

Тема 8.2. Методы управления портфолио IT-проектов. История появления термина портфолио IT-проектов. Суть и преимущества управления портфелем IT-проектов. Реализация управления портфелем IT-проектов и его сравнение с системой сбалансированных показателей. Матрица МакФарлана. План игры управления портфелем IT-проектов.

4.3.2. Лабораторные занятия

№	Наименование лабораторных работ	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
1	Методологии SADT и IDEF0	2	УК-3, ОПК-2, ОПК-5	экзамен
2	Описание потоков данных. Диаграммы DFD	2	УК-3, ОПК-2, ОПК-5	экзамен

№	Наименование лабораторных работ	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
3	Описание процессов. Диаграммы IDEF3	2	УК-3, ОПК-2, ОПК-5	экзамен
4	Диаграммы прецедентов и спецификация потоков	2	УК-3, ОПК-2, ОПК-5	экзамен
5	Диаграммы деятельности и автоматов	2	УК-3, ОПК-2, ОПК-5	экзамен
6	Диаграммы классов последовательности и коммуникаций	2	УК-3, ОПК-2, ОПК-5	экзамен
7	Диаграммы компонентов и размещения	2	УК-3, ОПК-2, ОПК-5	экзамен
8	Процессно-ориентированное проектирование ПС	2	УК-3, ОПК-2, ОПК-5	экзамен

4.3.3. Практические занятия

Практические имеют своей целью изучение методологий и технологий проектирования информационных систем, основанных как на классических, так и современных принципах.

Цель занятий. Освоить методологии и технологии проектирования обеспечивающих подсистем, прототипного и типового проектирования, организации и управления проектом, формирования и управления портфолио IT-проектов.

Задачи закрепления теоретических знаний и практических умений и навыков: обучаемый должен знать основные понятия, связанные с проектированием обеспечивающих подсистем, прототипным и типовым проектированием, организацией и управлением проектами, формированием и управлением портфолио IT-проектов

Форма проведения: обсуждение и анализ типовых процессов и задач проектирования ИС на аудиторных занятиях, самостоятельное изучение студентами опыта проектирования ИС в ведущих IT-компаниях в аудитории и в домашних условиях.

Очная форма обучения

№	Темы практических занятий	Трудоемкость (час.)	Формируемые компетенции	Форма контроля
1	Проектирование обеспечивающих подсистем	4	УК-3, ОПК-2, ОПК-5	экзамен
2	Индустриальное проектирование ИС. Прототипное и типовое проектирование	4	УК-3, ОПК-2, ОПК-5	экзамен
3	Организация и управление проектом ИС на всех стадиях жизненного цикла	4	УК-3, ОПК-2, ОПК-5	экзамен
4	Методы управления портфолио IT-проектов	4	УК-3, ОПК-2, ОПК-5	экзамен

4.3.4. Курсовой проект

В ходе выполнения курсового проекта студенты должны спроектировать программу для заданной предметной области и реализовать проект на изучаемом алгоритмическом языке высокого уровня. Пояснительная записка к проекту должна отражать выполнение работ важнейших этапов жизненного цикла программного обеспечения – Анализ и Синтез, включая: 1) анализ задачи и формулировку требований к создаваемой программе; 2) проектирование; 3) кодирование (программирование, реализацию); 4) тестирование; 5) отладку.

Во время проектирования необходимо: 1) осуществить выбор методов решения и логических структур данных; 2) обосновать способ реализации используемых структур данных (при необходимости необходимо создать логическую full attributed ЗНФ модель базы данных); 3) структурировать создаваемую программу – выделить подсистемы (модули, классы), выбрать архитектуру программы, определить режим управления и взаимодействия

компонентов; 4) разработать алгоритмы реализации модулей.

На этапе кодирования должны быть реализованы методы классов и необходимые структуры данных, при необходимости – создана физическая модель базы данных.

Разработку программы необходимо осуществить с использованием средств моделирования, шаблонов проектирования и прагматики используемого языка программирования. В частности, систему классов событийно-ориентированного приложения необходимо создавать с учетом рекомендаций, содержащихся в документации используемой системы программирования.

Выполняя проект необходимо создать UML-модели требований, проектирования и реализации, включая следующие диаграммы: 1) диаграмма бизнес-прецедентов (при необходимости); 2) диаграмма системных прецедентов; 3) диаграмма деятельности и/или автомата; 4) диаграммы взаимодействия (коммуникации и/или последовательности); 5) диаграмма классов; 6) диаграмма компонентов; 7) диаграмма развёртывания.

Некоторые из создаваемых диаграмм могут декомпозироваться, при этом возникает не одна, а несколько диаграмм конкретного вида. Диаграмма классов должна содержать не менее трех тематических пакетов (сущности, граничные классы и управляющие классы).

Диаграмма компонентов должна содержать три модели: модель исходного кода, модель исполняемого кода и модель артефактов, поставляемых заказчику.

Тема курсового проекта формулируется по схеме:

Разработка программной системы [заданной предметной области]

где вместо словосочетания [заданной предметной области] указывается конкретная предметная область – название предприятия или базовая функция ПС. Например:

Разработка программной системы автовокзала «Центральный» г. Рязани

Разработка программной системы для контроля знаний по языку UML

В процессе выполнения проекта осуществляется промежуточная аттестация по следующим этапам проектирования.

1. Создание модели требований. Студент предъявляет модель требований, включающую следующие элементы:
 - аспект функциональности
 - 1) UML-диаграмма бизнес-прецедентов;
 - 2) UML-диаграмма системных прецедентов;
 - 3) спецификация бизнес-прецедентов;
 - 4) спецификация системных прецедентов;
 - аспект данных
 - 5) пакет классов-сущностей диаграммы классов;
 - аспект поведения
 - 6) пакет граничных классов диаграммы классов.
2. Создание первой части модели проектирования. Студент предъявляет первую часть модели проектирования, включающую следующие элементы:
 - аспект функциональности
 - 1) UML-диаграммы детализации системных прецедентов (при необходимости);
 - 2) спецификации прецедентов диаграмм детализации (при наличии диаграмм детализации системных прецедентов);
 - 3) пакет управляющих классов диаграммы классов;
 - аспект данных
 - 4) дополненный пакет классов-сущностей диаграммы классов (входные и выходные параметры методов управляющих классов);
 - аспект поведения
 - 5) дополненный пакет граничных классов диаграммы классов (при наличии диаграмм детализации системных прецедентов);
 - 6) описания форм определенных на текущий момент граничных классов;

- 7) диаграммы деятельности и/или автоматов, алгоритмически реализующие определенные на текущий момент прецеденты системы.
3. Создание второй части модели проектирования. Студент предъявляет вторую часть модели проектирования, включающую следующие элементы:
 - аспект данных
 - 1) логическая full attributed ЗНФ модель базы данных (если база создается);
 - аспект поведения
 - 2) диаграммы коммуникации и/или последовательности на уровне спецификаций и/или экземпляров, представляющие роли и/или объекты классов, участвующие в реализации прецедентов и/или методов классов;
 - 3) описания форм всех граничных классов;
 - аспект структуризации
 - 4) диаграммы коммуникации на уровне спецификаций, представляющие роли классов, участвующие в реализации других классов;
4. Создание модели реализации. Студент предъявляет модель реализации, включающую следующие элементы:
 - аспект структуризации
 - 1) диаграммы компонентов, представляющие модели исходного и исполняемого кодов, а также модель артефактов, поставляемых заказчику;
 - 2) диаграмма развертывания.

4.3.5. Самостоятельная работа

Самостоятельное изучение тем учебной дисциплины способствует закреплению знаний, умений и навыков, полученных в ходе аудиторных занятий, углублению и расширению знаний по отдельным вопросам и темам дисциплины; освоению умений практического использования полученных знаний при моделировании и анализе различных функциональных узлов систем передачи данных, расчете их основных характеристик.

Самостоятельная работа обучающихся по данной дисциплине заключается в следующем:

- при подготовке к лекциям в изучении и доработке конспекта лекции с применением учебно-методической литературы, в решении заданных и подборе дополнительных примеров к теоретическим положениям дисциплины по данной теме;
- при подготовке к лабораторным работам в разработке, отладке и выполнении программного проекта своего варианта задания по данной теме, подготовке отчета и подготовке к защите лабораторного задания;
- в самостоятельном изучении отдельных вопросов и тем курса с применением рекомендуемой учебно-методической литературы;
- при подготовке к экзамену в изучении, осмыслении и повторении пройденного теоретического материала и выполненных практических заданий с применением конспекта лекций и учебно-методической литературы.

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Оценочные материалы приведены в приложении к рабочей программе дисциплины (см. документ «09.04.04 Б1.О.05 «Методы и средства проектирования и разработки программных систем Оценочные материалы.docx»

6. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1 Основная литература

1. Белов В.В., Чистякова В.И. Проектирование информационных систем: учебник – М. : КУРС, 2018. – 400 с. ISBN 978-5-906923-53-0 (КУРС) (45 экз. в БФ РГРТУ).

2. Белов В.В., Чистякова В.И. Проектирование информационных систем: учебник для студ. учреждений высш. образования / Под ред. В.В. Белова. – 2-е изд., стер. – М.: Издательский центр «Академия», 2015. – 352 с. (Сер. Бакалавриат). ISBN 978-5-4468-2440-3 (132 экз. в БФ РГРТУ)

3. Введение в программную инженерию: Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин, В.К. Столчнев. – М.: КУРС: ИНФРА-М, 2017. – 336 с. (40 экз. в БФ РГРТУ).

6.2. Дополнительная литература

4. Программная инженерия: Учебник / В.А. Антипов, А.А. Бубнов, А.Н. Пылькин. – М.: Издательский центр «Академия», 2014. – 288 с. (50 экз. в БФ РГРТУ).

5. Разработка и анализ требований к программному обеспечению: Учебник / А.А. Бубнов, С.А. Бубнов, К.А. Майков. – М.: КУРС: ИНФРА-М, 2018. – 176 с. (40 экз. в БФ РГРТУ).

6. Иванов, Денис Юрьевич. Унифицированный язык моделирования UML [Электронный ресурс]: учебное пособие для вузов по направлению подготовки "Системный анализ и управление" / Д.Ю. Иванов, Ф.А. Новиков; Санкт-Петербургский государственный политехн. ун-т. – Электрон. текстовые дан. (1 файл : 1,83 Мб). – Санкт-Петербург, 2011. – Загл. с титул. экрана. – Электронная версия печатной публикации. – Свободный доступ из сети Интернет (чтение, печать, копирование). – Текстовый документ. – Adobe Acrobat Reader 7.0. Доступно по URL: <http://elib.spbstu.ru/dl/2962.pdf>, <http://elib.spbstu.ru/dl/2962.pdf/download/2962.pdf>

7. Орлов С.А. Программная инженерия. Технологии разработки программного обеспечения: Учебник для вузов. –5-е изд. обновл. и доп. Стандарт третьего поколения. – СПб.: Питер, 2016. – 640 с. Электрон. текстовые дан. (1 файл : 37,58 Мб). — Текстовый документ. — Adobe Acrobat Reader, Internet Explorer. Доступно по URL <https://www.twirpx.com/file/2378219/>.

6.3. Методические указания к лабораторным и практическим занятиям

При подготовке к лабораторным работам используются основные учебники по данной дисциплине [1, 2]т.

6.4. Методические указания к самостоятельной работе

1. Учебно-методическая литература [1 – 5].

2. Электронные учебники, учебные пособия и учебно-методическое обеспечение по данной дисциплине в учебных классах кафедры в папке

//FS/Work/Docs/МО_дисциплин_кафедры.

7. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1. Электронно-библиотечная система «Лань». – Режим доступа: свободный доступ из корпоративной сети РГРТУ, после регистрации -- доступ из сети Интернет. – URL: <https://e.lanbook.com/>

2. Электронная библиотечная система «IPRbooks». – Режим доступа: с любого компьютера РГРТУ без пароля, из сети Интернет по паролю. – URL: <https://iprbookshop.ru/>.

3. Электронная библиотечная система РГРТУ. Режим доступа: свободный доступ из корпоративной сети РГРТУ, после регистрации – доступ из сети Интернет – URL: <http://elib.rsreu.ru>

4. Научная электронная библиотека eLibrary. – URL: <http://e.lib/vlsu.ru/www.uisrussia.msu.ru/elibrary.ru>

5. Библиотека и форум по программированию. – URL: <http://www.cyberforum.ru>

6. Национальный открытый университет ИНТУИТ. – URL: <http://www.intuit.ru/>

7. Информационно-справочная система. – URL: <http://window.edu.ru>

8. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ЛИЦЕНЗИОННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ

В учебном процессе применяются следующие информационные технологии:

- чтение лекций с использованием презентаций;
- выполнение студентами заданий с использованием лицензионного или свободно распространяемого программного обеспечения, установленного на рабочих местах студента в компьютерных классах и в помещениях для самостоятельной работы, а также для выполнения самостоятельной работы в домашних условиях.

Перечень используемого программного обеспечения: операционные системы WINDOWS XP, WINDOWS 7, средства разработки ПО Eclipse, FreePascal, SharpDevelop, свободно распространяемое программное обеспечение (сайт <http://www.>).

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Для освоения дисциплины необходимы:

- 1) лекционная аудитория, оборудованная средствами отображения презентаций и других лекционных материалов на экран;
- 2) компьютерные классы для проведения лабораторных работ с необходимым установленным лицензионным программным обеспечением.

№	Наименование специальных помещений и помещений для самостоятельной работы	Перечень специализированного оборудования
1		
2		
3		
4		

Программу составил

д-р. техн. наук, профессор кафедры

вычислительной и прикладной математики _____ В.В. Белов