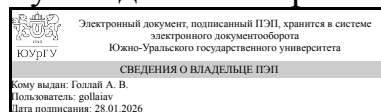


ЮЖНО-УРАЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ:
Руководитель направления



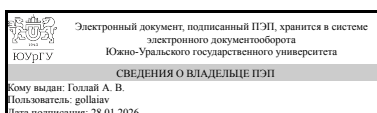
А. В. Голлай

РАБОЧАЯ ПРОГРАММА

дисциплины 1.О.26 Анализ требований и проектирование ПО
для направления 09.03.02 Информационные системы и технологии
уровень Бакалавриат
форма обучения очная
кафедра-разработчик Центр подготовки топ-специалистов в сфере ИТ "Цифровой Урал"

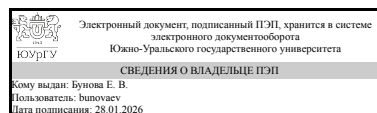
Рабочая программа составлена в соответствии с ФГОС ВО по направлению подготовки 09.03.02 Информационные системы и технологии, утверждённым приказом Минобрнауки от 19.09.2017 № 926

Зав.кафедрой разработчика,
д.техн.н., доц.



А. В. Голлай

Разработчик программы,
к.техн.н., доц., доцент



Е. В. Бунова

1. Цели и задачи дисциплины

формирование у обучающихся системного представления о процессах выявления, анализа, формализации и документирования требований к программному обеспечению, а также навыков проектирования архитектуры программных систем с использованием современных нотаций, стандартов и инструментальных средств. Задачи дисциплины: - формирование теоретических знаний о роли и месте анализа требований и проектирования в жизненном цикле программного обеспечения; - изучение методов сбора, классификации, анализа и управления требованиями; - освоение стандартов и регламентов разработки технического задания и проектной документации; - изучение нотаций и инструментов моделирования предметной области и бизнес-процессов (UML, BPMN, IDEF0); - формирование навыков объектно-ориентированного и предметно-ориентированного проектирования; - развитие умений проектирования архитектуры программных систем с учётом требований к качеству; - формирование практических навыков подготовки спецификаций, архитектурных описаний и проектной документации; - развитие компетенций командной работы при проектировании программных систем.

Краткое содержание дисциплины

Дисциплина посвящена изучению процессов анализа требований и проектирования программного обеспечения на этапах его жизненного цикла. Рассматриваются источники требований, методы их сбора и выявления, классификация функциональных и нефункциональных требований, техники интервьюирования, анкетирования, наблюдения и анализа документации. Изучаются методы формализации требований, разработка спецификаций требований к ПО, стандарты ГОСТ и ISO/IEC в области программной инженерии. Особое внимание уделяется моделированию бизнес-процессов в нотации BPMN, построению UML-диаграмм (диаграмм вариантов использования, классов, последовательностей, состояний, компонентов). Рассматриваются принципы архитектурного проектирования, архитектурные стили и шаблоны, влияние архитектурных решений на показатели качества программных систем. Изучаются основы подготовки технического задания, проектной и эксплуатационной документации. Практическая направленность дисциплины обеспечивается выполнением заданий по разработке требований и проектированию программного продукта.

2. Компетенции обучающегося, формируемые в результате освоения дисциплины

Планируемые результаты освоения ОП ВО (компетенции)	Планируемые результаты обучения по дисциплине
ОПК-4 Способен участвовать в разработке технической документации, связанной с профессиональной деятельностью с использованием стандартов, норм и правил	Знает: методы выявления, анализа и разработки требований при проектировании сложных программных систем; нотации моделирования бизнес-процессов в нотации BPMN для формирования требований к ПО; классификацию и структуру проектной документации на программные системы; стандарты и нотации моделирования (UML, BPMN IDEF0) для описания архитектуры и процессов; основы

	архитектуры программных систем и влияние архитектурных решений на качество ПО; правила оформления и ведения спецификаций, технических заданий Умеет: создавать и описывать объектно-ориентированные модели предметной области; разрабатывать техническую и проектную документацию на основе требований и стандартов; применять унифицированные языки моделирования (UML, IDEF0) для описания архитектуры и компонентов системы; использовать шаблоны проектирования при разработке архитектуры и спецификаций модулей; моделировать бизнес-процессы в нотации BPMN и анализировать требования, выявлять противоречия и формировать корректные спецификации; оформлять документы в соответствии с установленными нормами и шаблонами организации Имеет практический опыт: навыками подготовки и оформления документации (техническое задание, спецификация, концепция проекта, архитектурное описание) в соответствии со стандартами; инструментами и средствами CASE-технологий для создания диаграмм и спецификаций; навыками работы с инструментальными средствами моделирования бизнес-процессов и архитектуры ПО
ПК-1 Способен разрабатывать и отлаживать программный код	Знает: основные стандарты оформления технической документации на компьютерное программное обеспечение Умеет: применять заданные стандарты и шаблоны для составления и оформления технической документации
ПК-5 Способен разрабатывать требования и проектировать программное обеспечение	Знает: методы и средства проектирования компьютерного программного обеспечения, методы и средства проектирования программных интерфейсов Умеет: выбирать средства реализации требований к компьютерному программному обеспечению, вырабатывать варианты реализации компьютерного программного обеспечения, проводить оценку и обоснование рекомендуемых решений
ПК-7 Способен выполнять работы и управлять работами по созданию (модификации) и сопровождению информационных систем (ИС), автоматизирующих задачи организационного управления и бизнес-процессы	Знает: возможности типовой ИС, предметную область автоматизации, инструменты и методы моделирования бизнес-процессов Умеет: анализировать исходную документацию в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС, разрабатывать документы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС

3. Место дисциплины в структуре ОП ВО

Перечень предшествующих дисциплин, видов работ учебного плана	Перечень последующих дисциплин, видов работ
1.О.08 Основы программирования на Python, 1.О.12 Алгоритмы и структуры данных, 1.О.11 Математическая логика и теория алгоритмов, 1.О.14 Архитектура ЭВМ, 1.О.15 Основы теории булевых функций, 1.О.27 Системный анализ в ИТ-проектировании, 1.О.29 Информационные технологии в управлении организационными системами, 1.О.07 Информатика, 1.О.13 Программирование на языках высокого уровня, 1.О.18 Объектно-ориентированное программирование	1.О.30 Вычислительные методы в ИТ

Требования к «входным» знаниям, умениям, навыкам студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин:

Дисциплина	Требования
1.О.07 Информатика	Знает: методы и приемы алгоритмизации поставленных задач; нотации и программное обеспечение для графического отображения алгоритмов, принципы работы современных информационных технологий; принципы работы программных средств, базовые понятия информатики и информационных технологий, основные форматы представления информации для автоматизированной обработки; основные принципы работы вычислительных систем и их компонентов; ОПК-2.1. 3-3. Знает основные принципы решения задач с помощью компьютера, понятие алгоритма, основные алгоритмические структуры; состав, назначение функциональных компонентов и программного обеспечения персонального компьютера; Умеет: использовать методы и приемы формализации и алгоритмизации поставленных задач, использовать современные информационные технологий при решении задач профессиональной деятельности; использовать программные средства при решении задач профессиональной деятельности, работать с информацией, представленной в различных формах; разрабатывать алгоритмы для решения типовых задач; работать с персональным компьютером, применять современное ПО, в т.ч., отечественного производства, для решения практических задач Имеет практический опыт: навыками использования современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении

	задач профессиональной деятельности, разработки алгоритмов для решения типовых задач
1.О.18 Объектно-ориентированное программирование	Знает: методы и средства проверки работоспособности компьютерного программного обеспечения, синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования; методологии разработки компьютерного программного обеспечения; технологии программирования; возможности используемой системы управления версиями и вспомогательных инструментальных программных средств; установленный регламент использования системы управления версиями, основные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования, возможности компиляторов программных проектов под различные операционные системы, наборы инструкций для системных утилит автоматической сборки программного обеспечения и установки программных пакетов объектно-ориентированных библиотек и фреймворков, принципы чистого кода, SOLID, DRY, KISS и др., принципы предметно-ориентированного проектирования (ПОП) программного обеспечения, паттерны проектирования и антипаттерны Умеет: применять методы и средства проверки работоспособности компьютерного программного обеспечения, применять выбранные языки программирования для написания программного кода; использовать выбранную среду программирования; использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода; регистрировать изменения исходного текста программного кода в системе управления версиями; сохранять изменения программного кода в соответствии с регламентом управления версиями; выполнять слияние, разделение и сравнение исходных текстов программного кода, использовать функциональные возможности современных интегрированных сред разработки программного обеспечения на объектно-ориентированных языках программирования для разработки прикладных программ, использовать утилиты автоматической сборки и развертывания программ в операционных системах, разрабатывать модульный и тестируемый программный код, выполнять модульное, интеграционное и нагрузочное тестирование,

	проводить рефакторинг для повышения качества кода, применять принципы ПОП при разработке программного обеспечения на языках программирования высокого уровня абстракций и в LowCode и NoCode системах Имеет практический опыт: работы с основными современными интегрированными средами разработки программного обеспечения на объектно-ориентированных языках, разработки, отладки и развёртывания программного обеспечения в операционных системах семейства Windows и Linux
1.О.11 Математическая логика и теория алгоритмов	Знает: основные понятия и законы математической логики и теории алгоритмов, необходимые для построения моделей профессиональных задач, алгоритмы решения типичных задач, области и способы их применения; методы и приемы алгоритмизации поставленных задач Умеет: формализовать профессиональные задачи средствами математической логики и строить модели процессов с использованием методов теории алгоритмов, использовать методы и приемы формализации и алгоритмизации поставленных задач Имеет практический опыт: разработки и проверки правильности логико-математических конструкций и алгоритмов, используемых в расчетах и проектировании
1.О.12 Алгоритмы и структуры данных	Знает: синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования; методологии разработки компьютерного программного обеспечения; технологии программирования, фундаментальных свойств алгоритмов и структур данных; методы решения алгоритмических задачи с применением структур данных в соответствии с особенностями предметной области Умеет: применять выбранные языки программирования для написания программного кода; использовать выбранную среду программирования; возможности имеющейся технической и/или программной архитектуры для написания программного кода, проектировать алгоритмическое решение на основе выбранной структуры данных; использовать средства разработки и отладки современной интегрированной среды программирования Имеет практический опыт: разработки программного кода с использованием языков программирования, определения и манипулирования данными, применяя выбранную систему контроля версий и инструментальные программные средства, применения широкого набора приемов, методов

	и технологий программирования различных задач
1.О.27 Системный анализ в ИТ-проектировании	Знает: методы анализа бизнес-процессов (AS-IS), выявления точек автоматизации, цифровизации и трансформации, методы абстрактного мышления, анализа информации и синтеза проблемных ситуаций; основные способы осмысления и критического анализа проблемных ситуаций; методы нестандартных решений проблем, новых оригинальных проектов, методы анализа бизнес-процессов (AS-IS), выявления точек автоматизации, цифровизации и трансформации Умеет: применять полученные знания в области системного анализа и синтеза проблемных ситуаций, формализованных моделей процессов и явлений в профессиональной деятельности; анализировать исходные данные, необходимые для осмысления и критического анализа проблемных ситуаций; выявлять и обосновывать стратегию действий на основе системного подхода Имеет практический опыт:
1.О.14 Архитектура ЭВМ	Знает: методику настройки и наладки программно-аппаратных комплексов, возможности существующей программно-технической архитектуры, возможности современных и перспективных средств разработки программных продуктов, технических средств, методологии разработки компьютерного программного обеспечения и технологии программирования, методы оптимизации (кэш, память, тактовая частота процессора), инструменты профилирования, архитектуру компиляторов, оптимизации для выбранных архитектур процессоров, методику настройки и наладки программно-аппаратных комплексов Умеет: производить коллективную настройку и наладку программно-аппаратных комплексов, проводить сбор и систематизацию требований к компьютерному программному обеспечению, выявлять взаимосвязи и документировать требования к компьютерному программному обеспечению, вырабатывать варианты реализации требований к компьютерному программному обеспечению, анализировать бенчмарки, выявлять узкие места, разрабатывать код, оптимизированный для выбранной аппаратной архитектуры Имеет практический опыт: коллективной настройки и наладки программно-аппаратных комплексов
1.О.08 Основы программирования на Python	Знает: алгоритмы решения типичных задач, области и способы их применения; синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования, особенности синтаксиса

	языка Python и базовых библиотек для работы с информацией (NumPy, Pandas, Matplotlib) Умеет: применять выбранные языки программирования для написания программного кода, создавать эффективные программы на Python для решения простых и сложных задач обработки данных Имеет практический опыт: применения технологий написания читаемого и поддерживаемого кода на Python
1.О.29 Информационные технологии в управлении организационными системами	Знает: принципы работы современных информационных технологий; принципы работы программных средств, возможности типовой ИС; инструменты и методы моделирования бизнес-процессов; устройство и функционирование современных ИС; современные подходы и стандарты автоматизации организации Умеет: использовать современные информационные технологий при решении задач профессиональной деятельности; использовать программные средства при решении задач профессиональной деятельности; Имеет практический опыт: использования современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности
1.О.15 Основы теории булевых функций	Знает: теоретические основы и понятийный аппарат теории булевых функций; возможности и область применения аппарата теории булевых функций для понимания принципов работы современных информационных технологий и программных средств; алгоритмы решения типичных задач, области и способы их применения; методы и приемы алгоритмизации поставленных задач Умеет: применять аппарат теории булевых функций при решении задач профессиональной деятельности; , использовать методы и приемы формализации и алгоритмизации поставленных задач Имеет практический опыт: применения аппарата теории булевых функций при решении задач профессиональной деятельности
1.О.13 Программирование на языках высокого уровня	Знает: синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования; методологии разработки компьютерного программного обеспечения; технологии программирования; возможности используемой системы управления версиями и вспомогательных инструментальных программных средств; установленный регламент использования системы управления версиями, основные возможности современных интегрированных сред разработки программного обеспечения на языках высокого уровня,

	возможности компиляторов и компоновщиков под различные операционные системы, наборы инструкций для системных утилит автоматической сборки программ Умеет: применять выбранные языки программирования для написания программного кода; использовать выбранную среду программирования; использовать возможности имеющейся технической и/или программной архитектуры для написания программного кода; регистрировать изменения исходного текста программного кода в системе управления версиями; сохранять изменения программного кода в соответствии с регламентом управления версиями; выполнять слияние, разделение и сравнение исходных текстов программного кода, использовать функциональные возможности современных интегрированных сред разработки программного обеспечения на языках высокого уровня для разработки прикладных программ, использовать утилиты автоматической сборки и развертывания программ в операционных системах. Имеет практический опыт: работы с основными современными интегрированными средами разработки программного обеспечения на языках высокого уровня, разработки, отладки и развёртывания программного обеспечения в операционных системах семейства Windows и Linux
--	--

4. Объём и виды учебной работы

Общая трудоемкость дисциплины составляет 3 з.е., 108 ч., 70,25 ч. контактной работы

Вид учебной работы	Всего часов	Распределение по семестрам
		в часах
		Номер семестра
		4
Общая трудоёмкость дисциплины	108	108
<i>Аудиторные занятия:</i>	64	64
Лекции (Л)	16	16
Практические занятия, семинары и (или) другие виды аудиторных занятий (ПЗ)	48	48
Лабораторные работы (ЛР)	0	0
<i>Самостоятельная работа (СРС)</i>	37,75	37,75
Изучение литературы	37,75	37,75
Консультации и промежуточная аттестация	6,25	6,25
Вид контроля (зачет, диф.зачет, экзамен)	-	зачет

5. Содержание дисциплины

№ раздела	Наименование разделов дисциплины	Объем аудиторных занятий по видам в часах			
		Всего	Л	ПЗ	ЛР
1	Введение в программную инженерию и жизненный цикл программного обеспечения	2	2	0	0
2	Модели и стандарты жизненного цикла программного обеспечения	16	4	12	0
3	Сбор, анализ и управление требованиями к программному обеспечению	8	2	6	0
4	Моделирование бизнес-процессов и формирование требований к ПО. Проектирование архитектуры программных систем и техническое задание	16	4	12	0
5	Гибкие методологии разработки программного обеспечения. Анализ требований, проектирование, разработка ПО.	22	4	18	0

5.1. Лекции

№ лекции	№ раздела	Наименование или краткое содержание лекционного занятия	Кол- во часов
1	1	Программная инженерия как область профессиональной деятельности. Понятие программной инженерии, её цели, задачи и место в системе ИТ-дисциплин. Отличие программной инженерии от программирования. Программный продукт как объект инженерной деятельности. Основные характеристики программных систем: сложность, надежность, сопровождаемость, масштабируемость. Жизненный цикл программного обеспечения. Основные процессы ЖЦ ПО: инициация, анализ требований, проектирование, реализация, тестирование, внедрение, сопровождение и развитие. Роль анализа требований и проектирования в обеспечении качества ПО. Разбор примеров жизненного цикла разработки и внедрения типовых и уникальных программных систем. Выделение этапов разработки. Определение ролей участников проекта. Формирование схемы ЖЦПО для заданного программного продукта.	2
2	2	Модели жизненного цикла программного обеспечения. Понятие модели жизненного цикла. Каскадная модель: структура, этапы, достоинства и недостатки. Спиральная модель: итеративность, анализ рисков, области применения. Инкрементная модель. V-образная модель. Формирование каскадной модели жизненного цикла типовой и адаптируемой информационной системы. Анализ преимуществ и ограничений модели для различных типов проектов. Построение спиральной модели жизненного цикла. Выделение итераций, анализ и оценка рисков на каждой итерации. Области целесообразного применения различных моделей. Роль стандартов в управлении жизненным циклом ПО. ГОСТ 34: стадии создания автоматизированных систем, состав и содержание документации на каждой стадии. ГОСТ Р ИСО/МЭК 12207: структура процессов жизненного цикла, основные, вспомогательные и организационные процессы. Взаимосвязь стандартов с моделями ЖЦПО. Моделирование ЖЦ ИС по ГОСТ 34. Построение модели жизненного цикла автоматизированной системы в соответствии с ГОСТ 34. Определение стадий, этапов и комплектов документации	4
3	3	Требования к программному обеспечению. Понятие требования. Источники требований. Классификация требований: функциональные, нефункциональные, бизнес-требования, пользовательские требования, системные требования. Атрибуты качества ПО. Ограничения и	2

		предположения. Методы выявления требований: интервью, анкетирование, наблюдение, анализ документации, прототипирование. Предпроектные исследования. Обследование предметной области. Работа с заказчиком. Формирование перечня заинтересованных сторон. Процессы управления требованиями. Трассируемость требований. Верификация и валидация требований. Управление изменениями требований. Влияние изменений требований на архитектуру программного обеспечения. Обоснование проектных решений. Анализ изменений требований. Оценка влияния изменений на сроки, стоимость и архитектуру системы. Построение матрицы трассируемости требований.	
4	4	Моделирование бизнес-процессов и формирование требований к ПО. Проектирование архитектуры программных систем и техническое задание. Техническое задание. Основные разделы ТЗ по ГОСТ. Проектирование интерфейсов. Принципиальные архитектурные решения. Выбор архитектуры программного обеспечения. Выбор пользовательского интерфейса. Выбор средств реализации. Основы архитектурного проектирования.	4
5	5	Гибкие методологии разработки программного обеспечения. Принципы Agile. Методология Scrum. Роли: Product Owner, Scrum Master, команда разработки. Спринт. Бэклог продукта и бэклог спринта. Преимущества и ограничения гибких методологий. Разработка MVP-продукта Проектирование ЖЦ ИС по Scrum. Формирование бэклога продукта. Планирование спринта. Определение задач и сроков выполнения. Реинжиниринг и рефакторинг. Управление конфигурациями.	4

5.2. Практические занятия, семинары

№ занятия	№ раздела	Наименование или краткое содержание практического занятия, семинара	Кол-во часов
1	2	Моделирование жизненного цикла ИС по ГОСТ 34, ГОСТ Р ИСО/МЭК 12207 для каскадной модели и спиральной модели жизненного цикла при разработке типовой ИС. Анализ структуры ГОСТ 34. Определение стадий создания автоматизированной системы. Формирование перечня проектной и эксплуатационной документации по стадиям. Построение схемы жизненного цикла ИС.	6
2	2	Моделирование жизненного цикла ИС по ГОСТ 34, ГОСТ Р ИСО/МЭК 12207 для каскадной модели и спиральной модели жизненного цикла при разработке уникальной ИС. Анализ структуры ГОСТ 34. Определение стадий создания автоматизированной системы. Формирование перечня проектной и эксплуатационной документации по стадиям. Построение схемы жизненного цикла ИС.	6
3	3	Моделирование бизнес-процессов и формирование требований к ПО. Построение модели бизнес-процессов AS-IS и TO-BE. Выделение основных бизнес-процессов предметной области. Определение проблемных зон. Проектирование целевого бизнес-процесса TO-BE. Определение точек автоматизации. Формирование требований к информационной системе.	6
4	4	Оформление структуры ТЗ в соответствии с ГОСТ 34. Определение разделов: назначение системы, требования к функциям, требования к надёжности, условия эксплуатации.	6
5	4	Проектирование архитектуры программной системы. Выбор архитектурного стиля. Определение компонентов системы. Формирование общей архитектурной схемы. Выбор инструментов разработки ПО	6
6	5	Основы Agile и Scrum. Определение ролей в Scrum-команде. Формирование бэклога продукта. Определение пользовательских историй. Декомпозиция	6

		пользовательских историй на задачи. Оценка трудоёмкости. Формирование бэклога спринта. Контроль выполнения задач. Анализ статуса проекта. Работа с доской задач (Kanban-подход).	
7	5	Формирование тестов и парное программирование. Контроль выполнения задач. Анализ статуса проекта. Работа с доской задач (Kanban-подход).	6
8	5	Управление изменениями в проекте. Спринты. Контроль выполнения задач. Анализ статуса проекта. Работа с доской задач (Kanban-подход). Реинжиниринг и рефакторинг. Документирование сопровождения ИС.	6

5.3. Лабораторные работы

Не предусмотрены

5.4. Самостоятельная работа студента

Выполнение СРС			
Подвид СРС	Список литературы (с указанием разделов, глав, страниц) / ссылка на ресурс	Семестр	Кол-во часов
Изучение литературы	Ботвинков, А. В. Технологии проектирования информационных систем и технологий : учебное пособие / А. В. Ботвинков, С. В. Моторин. — Новосибирск : СГУВТ, 2023. — 120 с. — ISBN 978-5-8119-0978-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/369893 (дата обращения: 08.12.2025). — Режим доступа: для авториз. пользователей.	4	37,75

6. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации

Контроль качества освоения образовательной программы осуществляется в соответствии с Положением о балльно-рейтинговой системе оценивания результатов учебной деятельности обучающихся.

6.1. Контрольные мероприятия (КМ)

№ КМ	Се-местр	Вид контроля	Название контрольного мероприятия	Вес	Макс. балл	Порядок начисления баллов	Учитывается в ПА
1	4	Текущий контроль	Контрольная работа 1	1	5	Оценка «5» (отлично) Контрольная работа выполнена в полном объеме и в строгом соответствии с заданием. Теоретический материал изложен глубоко, логично и системно, все понятия раскрыты корректно с использованием научной терминологии. Проведён обоснованный выбор модели жизненного цикла	зачет

					программного обеспечения с учётом особенностей предметной области, предложенной индустриальными партнёрами. Корректно построены стадии жизненного цикла по ГОСТ 34, сформирован полный комплект проектной и эксплуатационной документации. Установлено полное и корректное соответствие стадий ГОСТ 34 процессам стандарта ISO/IEC 12207. Работа оформлена в соответствии с установленными требованиями, отсутствуют фактические и логические ошибки. Оценка «4» (хорошо) Контрольная работа в целом выполнена правильно и в полном объёме. Теоретический материал раскрыт достаточно полно, однако отдельные положения изложены недостаточно глубоко или имеют незначительные неточности. Выбор модели жизненного цикла обоснован, но аргументация в отдельных местах носит фрагментарный характер. Стадии ГОСТ 34 и комплект документации сформированы корректно, допускаются отдельные неточности в их описании. Соответствие стандарту ISO/IEC 12207 установлено в основном верно. Оформление работы в целом соответствует требованиям, возможны незначительные замечания. Оценка «3» (удовлетворительно) Контрольная работа выполнена частично. Теоретический материал раскрыт поверхностно, допускаются пробелы в изложении ключевых положений и терминологии. Выбор модели жизненного цикла выполнен формально, без достаточного обоснования. Стадии ГОСТ 34 и комплект документации представлены неполно или с ошибками. Соответствие стандарту ISO/IEC 12207 установлено частично. В структуре и оформлении работы имеются существенные замечания. Работа в целом демонстрирует минимально допустимый уровень освоения материала. Оценка «2» (неудовлетворительно) Контрольная работа не соответствует требованиям задания. Теоретический материал раскрыт фрагментарно либо содержит значительное количество ошибок. Выбор модели жизненного цикла не обоснован или выполнен некорректно. Стадии ГОСТ 34 и документация представлены неправильно или отсутствуют. Соответствие стандарту ISO/IEC 12207 не раскрыто. Структура работы нарушена,	
--	--	--	--	--	---	--

						требования к оформлению не соблюдены. Уровень знаний и умений не соответствует минимально допустимому. Оценка «1» (работа не зачтена) Контрольная работа не выполнена либо представленный материал не имеет отношения к заданной теме. Отсутствует теоретическая и практическая часть. Работа не подлежит оцениванию и подлежит обязательной переработке.	
2	4	Текущий контроль	Контрольная работа 2	1	5	Оценка «5» (отлично) Работа полностью выполнена по задаче индустриального партнёра. Корректно определены заинтересованные стороны. Сформированы не менее 10 функциональных и 10 нефункциональных требований, соответствующих стандартам IEEE и ГОСТ. Матрица трассируемости построена корректно. Процедура управления изменениями описана полностью. Оформление без замечаний. Оценка «4» (хорошо) Работа выполнена по задаче партнёра. Требования и заинтересованные стороны в целом определены верно, но имеются незначительные неточности. Матрица трассируемости и процедура управления изменениями представлены, но раскрыты недостаточно подробно. Оформление с несущественными замечаниями. Оценка «3» (удовлетворительно) Работа выполнена формально по задаче партнёра. Перечень заинтересованных сторон и требований неполный. Требования частично не соответствуют стандартам. Матрица трассируемости и процедура управления изменениями представлены частично. Имеются замечания по структуре и оформлению. Оценка «2» (неудовлетворительно) Работа слабо связана с задачей партнёра. Требования сформулированы некорректно или в недостаточном объёме. Матрица трассируемости отсутствует либо выполнена с грубыми ошибками. Процедура управления изменениями не раскрыта. Оценка «1» (работа не зачтена) Работа не выполнена по задаче партнёра либо не соответствует теме. Требования, матрица трассируемости и описание управления изменениями отсутствуют. Работа подлежит полной переработке.	зачет
3	4	Текущий контроль	Контрольная работа 3	1	5	Оценка «5» (отлично) Контрольная работа полностью выполнена по задаче индустриального партнёра.	зачет

					Модель AS-IS построена корректно и полно отражает реальный процесс. Проблемные зоны выявлены обоснованно. Модель TO-BE логически выстроена, содержит элементы автоматизации и оптимизации. Точки автоматизации определены корректно. Требования к информационной системе сформированы на основе модели TO-BE. Ожидаемый эффект от внедрения обоснован. Работа оформлена без замечаний. Оценка «4» (хорошо) Работа выполнена по задаче партнёра. Модель AS-IS и TO-BE в целом корректны, однако допускаются отдельные неточности в обозначении ролей, потоков или функций. Анализ проблем проведён, но не полностью раскрыт. Требования к ИС сформированы, но часть формулировок носит обобщённый характер. Оформление в основном соответствует требованиям. Оценка «3» (удовлетворительно) Работа выполнена формально по задаче партнёра. Модель AS-IS или TO-BE отражена частично. Проблемные зоны выявлены поверхностно. Точки автоматизации определены неполно. Требования к ИС сформированы без достаточной проработки. Имеются существенные замечания по структуре и оформлению работы. Оценка «2» (неудовлетворительно) Работа слабо связана с задачей партнёра. Модели AS-IS и TO-BE выполнены с грубыми ошибками или отсутствуют. Анализ проблем не проведён. Требования к ИС не сформированы либо некорректны. Оформление не соответствует установленным требованиям. Оценка «1» (работа не зачтена) Контрольная работа не выполнена по задаче партнёра либо не соответствует теме. Модели бизнес-процессов отсутствуют. Требования и обоснование эффекта не представлены. Работа подлежит обязательной полной переработке.		
4	4	Текущий контроль	Контрольная работа 4	1	5	Оценка «5» (отлично) Работа полностью выполнена по задаче индустриального партнёра. Структура технического задания соответствует требованиям ГОСТ 34. Разделы ТЗ разработаны корректно и полно.	зачет

					Архитектурная схема системы логична, обоснована и соответствует сформированным требованиям. Выбор архитектурных решений аргументирован с учётом показателей качества программного обеспечения. Оформление выполнено без замечаний. Оценка «4» (хорошо) Работа выполнена по задаче партнёра. В целом структура ТЗ соблюдена, однако отдельные разделы раскрыты недостаточно подробно. Архитектурная схема разработана корректно, но обоснование решений представлено не в полном объёме. Оформление в целом соответствует требованиям, допускаются незначительные замечания. Оценка «3» (удовлетворительно) Работа выполнена формально по задаче партнёра. Структура ТЗ представлена частично. Требования к функциям и надёжности раскрыты поверхностно. Архитектурная схема содержит упрощённое описание системы. Обоснование решений недостаточно аргументировано. Имеются существенные замечания по структуре и оформлению. Оценка «2» (неудовлетворительно) Работа слабо связана с задачей партнёра. Структура ТЗ нарушена. Требования к функциям и надёжности сформированы некорректно или неполно. Архитектурная схема отсутствует либо выполнена с грубыми ошибками. Обоснование решений не представлено. Оформление не соответствует требованиям. Оценка «1» (работа не зачтена) Контрольная работа не выполнена по задаче партнёра либо не соответствует теме. Разделы ТЗ отсутствуют. Архитектурные решения не представлены. Работа подлежит обязательной полной переработке.	
5	4	Промежуточная аттестация	Итоговая контрольная работа	-	5 Оценка «5» (отлично) Проект полностью выполнен по задаче индустриального партнёра. Представлен полный и логически связанный комплект проектных материалов. Все этапы (анализ, требования, AS-IS/TO-BE, архитектура, ТЗ, Scrum-бэклог, спринты, версии ПО) выполнены корректно. Применение методологии Scrum обосновано. Работа оформлена без замечаний. Оценка «4» (хорошо) Проект выполнен по задаче партнёра. В целом представлен полный набор проектных материалов, однако отдельные элементы	зачет

					проработаны недостаточно глубоко. Возможны отдельные неточности в логике связи требований, архитектуры и спринтов. Оформление в целом соответствует требованиям. Оценка «3» (удовлетворительно) Проект выполнен формально по задаче партнёра. Анализ предметной области и требования представлены поверхностно. Модели AS-IS/TO-BE, архитектура и Scrum-артефакты разработаны частично. Логическая связность проекта нарушена. Имеются существенные замечания по структуре и оформлению. Оценка «2» (неудовлетворительно) Проект слабо связан с задачей партнёра. Отсутствуют ключевые проектные разделы (AS-IS/TO-BE, архитектура, ТЗ, Scrum-планирование). Представленные материалы фрагментарны и содержат грубые ошибки. Оформление не соответствует требованиям. Оценка «1» (работа не зачтена) Итоговая контрольная работа не выполнена по задаче партнёра либо не соответствует тематике дисциплины. Проектные материалы отсутствуют. Работа подлежит обязательной полной переработке.	
--	--	--	--	--	--	--

6.2. Процедура проведения, критерии оценивания

Вид промежуточной аттестации	Процедура проведения	Критерии оценивания
зачет	Зачёт по дисциплине «Анализ требований и проектирование программного обеспечения» проводится в соответствии с утверждённым учебным планом и рабочей программой дисциплины и является формой промежуточной аттестации обучающихся. К прохождению зачёта допускаются обучающиеся, выполнившие в полном объёме все виды текущей учебной работы, предусмотренные рабочей программой дисциплины, включая выполнение и защиту контрольных работ №1–№4, а также представление итоговой контрольной (проектной) работы №5. Наличие академических задолженностей по текущему контролю является основанием для недопуска к зачёту. Зачёт проводится в очной форме в виде защиты итоговой контрольной работы и устного собеседования по основным разделам дисциплины. По решению кафедры зачет может проводиться также с использованием электронных образовательных технологий и дистанционных форм обучения. В ходе зачёта обучающийся представляет результаты итоговой контрольной работы, выполненной на основе задачи, предоставленной индустриальным партнёром кафедры. Студент кратко докладывает цель проекта, предметную область, основные требования, архитектурные решения, структуру технического задания, применяемые методологии и полученные результаты. После доклада обучающийся отвечает	В соответствии с пп. 2.5, 2.6 Положения

	на вопросы преподавателя, направленные на проверку теоретических знаний и практических навыков в области жизненного цикла программного обеспечения, анализа требований, бизнес-моделирования, архитектурного проектирования, стандартов ГОСТ 34 и ISO/IEC 12207, а также принципов Agile и Scrum. Оценивание результатов зачёта осуществляется на основе совокупности следующих показателей: полнота и качество выполнения итоговой контрольной работы, степень соответствия работы задаче индустриального партнёра, корректность применения стандартов и методологий, уровень сформированности профессиональных компетенций, а также ответы обучающегося в ходе устного собеседования. Решение о выставлении зачёта принимается преподавателем по результатам защиты итоговой работы и ответов обучающегося. Итоговая оценка выставляется по 5-балльной шкале и заносится в ведомость и зачётную книжку обучающегося в установленном порядке. Обучающиеся, не прошедшие зачёт по уважительной причине, допускаются к повторной аттестации в сроки, установленные расписанием пересдач. Обучающиеся, не прошедшие зачёт по неуважительной причине, проходят повторную аттестацию в порядке, установленном локальными нормативными актами вуза. В случае неудовлетворительного результата зачёта обучающемуся предоставляется возможность повторного прохождения промежуточной аттестации после устранения выявленных недостатков в установленный вузом срок.	
--	--	--

6.3. Паспорт фонда оценочных средств

Компетенции	Результаты обучения	№ КМ				
		1	2	3	4	5
ОПК-4	Знает: методы выявления, анализа и разработки требований при проектировании сложных программных систем; нотации моделирования бизнес-процессов в нотации BPMN для формирования требований к ПО; классификацию и структуру проектной документации на программные системы; стандарты и нотации моделирования (UML, BPMN IDEF0) для описания архитектуры и процессов; основы архитектуры программных систем и влияние архитектурных решений на качество ПО; правила оформления и ведения спецификаций, технических заданий	+	+	+	+	
ОПК-4	Умеет: создавать и описывать объектно-ориентированные модели предметной области; разрабатывать техническую и проектную документацию на основе требований и стандартов; применять унифицированные языки моделирования (UML, IDEF0) для описания архитектуры и компонентов системы; использовать шаблоны проектирования при разработке архитектуры и спецификаций модулей; моделировать бизнес-процессы в нотации BPMN и анализировать требования, выявлять противоречия и формировать корректные спецификации; оформлять документы в соответствии с установленными нормами и шаблонами организации	+	+	+	+	
ОПК-4	Имеет практический опыт: навыками подготовки и оформления документации (техническое задание, спецификация, концепция проекта, архитектурное описание) в соответствии со стандартами; инструментами и средствами CASE-технологий для создания диаграмм и спецификаций; навыками работы с инструментальными средствами моделирования бизнес-	+	+	+	+	

	процессов и архитектуры ПО					
ПК-1	Знает: основные стандарты оформления технической документации на компьютерное программное обеспечение	++			++	
ПК-1	Умеет: применять заданные стандарты и шаблоны для составления и оформления технической документации	++			++	
ПК-5	Знает: методы и средства проектирования компьютерного программного обеспечения, методы и средства проектирования программных интерфейсов	++			++	
ПК-5	Умеет: выбирать средства реализации требований к компьютерному программному обеспечению, вырабатывать варианты реализации компьютерного программного обеспечения, проводить оценку и обоснование рекомендуемых решений	++			++	
ПК-7	Знает: возможности типовой ИС, предметную область автоматизации, инструменты и методы моделирования бизнес-процессов	++	++		++	
ПК-7	Умеет: анализировать исходную документацию в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС, разрабатывать документы в рамках выполнения работ и управления работами по созданию (модификации) и сопровождению ИС	++	++		++	

Типовые контрольные задания по каждому мероприятию находятся в приложениях.

7. Учебно-методическое и информационное обеспечение дисциплины

Печатная учебно-методическая документация

а) основная литература:

Не предусмотрена

б) дополнительная литература:

Не предусмотрена

в) отечественные и зарубежные журналы по дисциплине, имеющиеся в библиотеке:

Не предусмотрены

г) методические указания для студентов по освоению дисциплины:

1. ФОС для студентов
2. УММ для студентов

из них: учебно-методическое обеспечение самостоятельной работы студента:

1. ФОС для студентов
2. УММ для студентов

Электронная учебно-методическая документация

№	Вид литературы	Наименование ресурса в электронной форме	Библиографическое описание
1	Основная литература	ЭБС издательства Лань	Туманова, М. Б. Проектирование программных систем : учебное пособие / М. Б. Туманова, Е. К. Михайлова, Е. А. Муравьева. — Москва : РТУ МИРЭА, 2023. — 138 с. — ISBN 978-5-7339-2050-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL:

			https://e.lanbook.com/book/398273 (дата обращения: 08.12.2025). — Режим доступа: для авториз. пользователей.
2	Дополнительная литература	ЭБС издательства Лань	Павлов, В. Д. Технология программирования. Управление разработкой : учебное пособие / В. Д. Павлов. — Санкт-Петербург : СПбГУ ГА им. А.А. Новикова, 2025. — 74 с. — ISBN 978-5-907860-25-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/509319 (дата обращения: 08.12.2025). — Режим доступа: для авториз. пользователей.
3	Основная литература	ЭБС издательства Лань	Рочев, К. В. Информационные технологии. Анализ и проектирование информационных систем : учебное пособие для вузов / К. В. Рочев. — 4-е изд., стер. — Санкт-Петербург : Лань, 2025. — 128 с. — ISBN 978-5-507-50803-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/465164 (дата обращения: 08.12.2025). — Режим доступа: для авториз. пользователей.
4	Основная литература	ЭБС издательства Лань	Ивановский, М. А. Методы и средства проектирования информационных систем и технологий : учебное пособие / М. А. Ивановский, И. А. Глазкова. — Тамбов : ТГТУ, 2024. — 130 с. — ISBN 978-5-8265-2787-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/472319 (дата обращения: 08.12.2025). — Режим доступа: для авториз. пользователей.
5	Дополнительная литература	ЭБС издательства Лань	Зубкова, Т. М. Технология разработки программного обеспечения : учебное пособие / Т. М. Зубкова. — Санкт-Петербург : Лань, 2022. — 324 с. — ISBN 978-5-8114-3842-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/206882 (дата обращения: 08.12.2025). — Режим доступа: для авториз. пользователей.
6	Методические пособия для самостоятельной работы студента	ЭБС издательства Лань	Ботвинков, А. В. Технологии проектирования информационных систем и технологий : учебное пособие / А. В. Ботвинков, С. В. Моторин. — Новосибирск : СГУВТ, 2023. — 120 с. — ISBN 978-5-8119-0978-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/369893 (дата обращения: 08.12.2025). — Режим доступа: для авториз. пользователей.

Перечень используемого программного обеспечения:

Нет

Перечень используемых профессиональных баз данных и информационных справочных систем:

Нет

8. Материально-техническое обеспечение дисциплины

Не предусмотрено