

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Б1.В.12
(индекс дисциплины)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Управление требованиями к программному обеспечению

(наименование дисциплины)

по направлению подготовки
09.03.03 Прикладная информатика

направленность (профиль)
Цифровая трансформация бизнеса

Форма обучения: заочная

Год набора: 2020

Общая трудоемкость: 6 ЗЕ

Распределение часов дисциплины по семестрам

Семестр		7	Итого
Вид занятий	Форма контроля	Экзамен	
Лекции		4	4
Лабораторные			
Практические		8	8
Руководство: курсовые работы (проекты) / РГР			
Промежуточная аттестация		0,35	0,35
Контактная работа		12,35	12,35
Самостоятельная работа		195	195
Контроль		8,65	8,65
Итого		216	216

Рабочую программу составил(и):

Доцент, доцент, к.п.н., Гущина О.М.

(должность, ученое звание, степень, Фамилия И.О.)

Рецензирование рабочей программы дисциплины:



Отсутствует



Рецензент

(должность, ученое звание, степень, Фамилия И.О.)

Рабочая программа дисциплины составлена на основании ФГОС ВО и учебного плана направления подготовки (специальности)

09.03.03 Прикладная информатика

Срок действия рабочей программы дисциплины до «31» августа 2025 г.

УТВЕРЖДЕНО

На заседании кафедры
«Прикладная математика и информатика»

(протокол заседания № 1 от «09» сентября 2019 г.).

1. Цель освоения дисциплины

Цель освоения дисциплины – формирование у студентов профессиональных знаний и практических навыков по управлению требованиями, предъявляемыми заинтересованными сторонами в процессе создания и/или внедрения автоматизированных информационных систем, программных продуктов.

В рамках курса изучаются основные этапы сбора и анализа требований, методы выявления, систематизации и описания требований

Курс нацелен на формирование у студента навыков сбора, обработки и представления исходных данных для описания требований к программным системам

2. Место дисциплины в структуре ОПОП ВО

Дисциплины и практики, на освоении которых базируется данная дисциплина: Технологии и средства конструирования программного обеспечения, Учебная практика (ознакомительная практика) 2.

Дисциплины и практики, для которых освоение данной дисциплины необходимо как предшествующее: Основы моделирования и проектирования программного обеспечения, Разработка предметно-ориентированных систем, Производственная практика (технологическая (проектно-технологическая) практика) 2

3. Планируемые результаты обучения

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1 ₁ - Знает принципы сбора, отбора и обобщения информации, методики системного подхода для решения профессиональных задач. УК-1 ₂ - Умеет анализировать и систематизировать разнородные данные, оценивать эффективность процедур анализа проблем и принятия решений в профессиональной деятельности. УК-1 ₃ - Владеет навыками научного поиска и практической работы с информационными источниками; методами принятия решений.	Знать: способы поиска, критического анализа и синтеза информации для формирования требований к программному обеспечению, методы системного анализа для решения поставленных задач прикладной предметной области
		Уметь: осуществлять поиск, критический анализ и синтез информации для формирования требований к программному обеспечению, применять системный подход для решения поставленных задач прикладной предметной области
		Владеть: навыками осуществления поиска, критического анализа и синтеза информации для формирования требований к программному обеспечению, методами системного анализа для решения поставленных задач прикладной предметной области
ПК-3 - Способен разрабатывать и адаптировать прикладное программное обеспечение	ПК-3 ₁ - Знает современные технологии разработки и адаптации прикладного программного обеспечения, их достоинства и недостатки ПК-3 ₂ - Умеет разрабатывать, адаптировать компоненты	Знать: способы и технологии описания требований при разработке и адаптации прикладного программного обеспечения для решения поставленных задач прикладной предметной области
		Уметь: разрабатывать и адаптировать

Формируемые и контролируемые компетенции (код и наименование)	Индикаторы достижения компетенций (код и наименование)	Планируемые результаты обучения
	прикладного программного обеспечения ПК-3 ₃ - Владеет навыками разработки прикладного программного обеспечения на современных языках программирования, методами адаптации прикладного программного обеспечения	прикладное программное обеспечение с четким формированием функциональных и нефункциональных требований
		Владеть: навыками формирования требований при разработке и адаптации прикладного программного обеспечения для решения поставленных задач прикладной предметной области

4. Структура и содержание дисциплины

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
Требования к программному обеспечению	Лек	Теоретические основы разработки требований	4	4	8	-	Тестовые задания 1
	Ср	Требования с точки зрения клиента		7		-	
	Ср	Рекомендуемые приемы формулирования требований		7		-	
	Ср	Роль бизнес-аналитика в проектах		7		-	
	Ср	Разработка описания программного обеспечения		7		-	
	Ср	Формирование функциональных требований к ПО		7		-	
	Пр	Оформление технического задания		2	15	-	Отчет по практическому заданию 1
Разработка требований	Ср	Определение бизнес-требований		4	8	-	Тестовые задания 2
	Ср	Анализ требований пользователей		4		-	
	Ср	Бизнес-правила и требования		4		-	
	Ср	Документирование требований		4		-	
	Ср	Принципы создания требований		4		-	
	Ср	Моделирование требований		4		-	
	Ср	Определение требований к данным		4		-	
	Ср	Сбор детальной информации от пользователей		4		-	
	Ср	Описание спецификаций полученной информации		4		-	

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
Требования в проектах определенных классов	Ср	Разработка требований к программному обеспечению		4		-	Тестовые задания 3
	Ср	Проекты гибкой разработки (agile)		10	8	-	
	Ср	Проекты автоматизации бизнес-процессов		10		-	
	Ср	Построение функциональной модели проекта		20		-	Отчет по практическому заданию 2
	Пр	Построение диаграммы вариантов использования		2	15	-	
	ПА	Промежуточное тестирование по пройденным темам		0,35		-	
Управление требованиями	Ср	Приемы управления требованиями к ПО		10	8	-	Тестовые задания 4
	Ср	Инструментальные средства разработки требований		10		-	
	Ср	Построение диаграммы классов проекта		20		-	
	Пр	Построение объектно-ориентированной модели проекта		2	15	-	Отчет по практическому заданию 3
Реализация процесса построения требований	Ср	Совершенствование процессов работы с требованиями		10	8	-	Тестовые задания 5
	Ср	Требования к ПО и управление рисками		10		-	
	Ср	Построение диаграммы Ганта по управлению проектом разработки ПО		10			
	Ср	Создание прототипа программного обеспечения проекта по выбранной теме		10		-	

Модуль (раздел)	Вид учебной работы	Наименование тем занятий (учебной работы)	Семестр	Объем, ч.	Баллы	Интерактив, ч.	Формы текущего контроля (наименование оценочного средства)
	Пр	Прототипирование программного обеспечения проекта по выбранной теме		2	15	-	Отчет по практическому заданию 4
	Контроль	Итоговое тестирование по курсу		8,65			Вопросы к экзамену
Итого:				216	100		

Схема расчета итогового балла

Текущий рейтинг (все занятия и промежуточные тесты) + Результат итогового теста и все делится на 2

5. Образовательные технологии

При изучении дисциплины (учебного курса) используются дистанционные образовательные технологии.

6. Методические указания по освоению дисциплины

6.1. Рекомендации по подготовке к практическим занятиям

Студентам следует:

- при подготовке к занятиям обязательно использовать не только учебную литературу, но и другие источники;
- в начале занятий задать преподавателю вопросы по материалу, вызвавшему затруднения в его понимании и освоении при решении задач, заданных для самостоятельного решения;
- на занятии доводить каждую задачу до окончательного решения, демонстрировать понимание проведенных расчетов (анализов, ситуаций), в случае затруднений обращаться к преподавателю.

При самостоятельном решении задач нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если студент видит несколько путей решения проблемы (задачи), то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы (задачи). Решение проблемных задач или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что решение каждой учебной задачи должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данной задачи. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение задач данного типа нужно продолжать до приобретения твердых навыков в их решении.

6.2. Рекомендации по подготовке к тестированию по темам курса

Тесты – это вопросы или задания, предусматривающие конкретный, краткий, четкий ответ на имеющиеся эталоны ответов.

При самостоятельной подготовке к тестированию студенту необходимо:

а) готовясь к тестированию, проработайте информационный материал по дисциплине. Проконсультируйтесь с преподавателем по вопросу выбора учебной литературы;

б) четко выясните все условия тестирования заранее. Вы должны знать, сколько тестов Вам будет предложено, сколько времени отводится на тестирование, какова система оценки результатов и т.д.;

в) приступая к работе с тестами, внимательно и до конца прочтите вопрос и предлагаемые варианты ответов. Выберите правильные (их может быть несколько). На отдельном листке ответов выпишите цифру вопроса и буквы, соответствующие правильным ответам;

г) в процессе решения желательно применять несколько подходов в решении задания. Это позволяет максимально гибко оперировать методами решения, находя каждый раз оптимальный вариант.

д) если Вы встретили чрезвычайно трудный для Вас вопрос, не тратьте много времени на него. Переходите к другим тестам. Вернитесь к трудному вопросу в конце.

е) обязательно оставьте время для проверки ответов, чтобы избежать механических ошибок.

Тестирование - позволяет оценить знание фактического материала, умение логически мыслить, способность к рефлексии и творчески подходить к решению поставленной задачи.

6.3. Рекомендации по подготовке к экзамену

Подготовка к экзамену способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На экзамен студент демонстрирует то, что он приобрел в процессе обучения по конкретной учебной дисциплине.

Необходимо ориентировать студентов на систематическую подготовку к занятиям в течение семестра, что позволит использовать время экзаменационной сессии для систематизации знаний.

7. Оценочные средства

7.1. Паспорт оценочных средств

Семестр	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
4	УК-1	Тестовые задания 1-5 Вопросы к экзамену
	ПК-3	Отчет по практическим заданиям 1-4 Вопросы к экзамену

7.2. Типовые задания или иные материалы, необходимые для текущего контроля

7.2.1. Тестовые задания

(наименование оценочного средства)

Типовой пример задания

Как вы получаете информацию от пользователей для требований?

- а) Разработчики уже знают, что создавать.
- б) Маркетологи, менеджеры по продукту или менеджеры пользователей уверены, что способны выразить точку зрения пользователей.
- в) Проводится анкетирование или опрос фокус-групп пользователей.
- г) В проекте участвуют конкретные представители различных классов пользователей, наделенные конкретной ответственностью обязанностями и полномочиями.

Насколько хорошо обучены и насколько опытны ваши аналитики требований?

- а) Они — разработчики или бывшие пользователи, имеющие мало опыта и не проходившие обучения в области конструирования требований к ПО.
- б) Разработчики, опытные пользователи или менеджеры проектов, имеющие некоторый опыт конструирования требований, исполняют роль аналитиков.
- в) Аналитики прошли обучение в течение нескольких дней и имеют значительный опыт сотрудничества с пользователями.
- г) У нас есть профессиональные бизнес-аналитики или конструкторы требований, имеющие подготовку и опыт в области приемов проведения собеседования, ведения собраний групп и написания технических текстов.

Как системные требования размещаются по программным частям продукта?

- а) Предполагается, что ПО будет работать, несмотря на любые недостатки оборудования.
- б) Конструкторы ПО и оборудования обсуждают, какие подсистемы должны выполнять конкретные функции.
- в) Конструктор или архитектор системы анализирует системные требования и решает, какие из них будут реализованы в каждой подсистеме.
- г) Части системных требований размещаются по программным подсистемам и трассируются до отдельных программных требований. Интерфейсы опытные ваши аналитики требований?

Какие методы используются для понимания проблем клиента?

- а) Наши разработчики очень умные; они и так понимают все проблемы.
- б) Мы спрашиваем пользователей, что они хотят, а потом это создаем.
- в) Мы обсуждаем с пользователями их бизнес-нужды и системы, с которыми они работают в настоящее время, а затем составляем спецификацию требований.
- г) Мы наблюдаем за тем, как пользователи выполняют свои задачи, моделируем их текущие рабочие процессы и узнаем, что они собираются делать при помощи новой системы.

Какие приемы используются для выявления всех отдельных требований к ПО?

- а) Мы начинаем с общего понимания, пишем код, а потом модифицируем его, пока все не получится.

- б) Руководство или отдел маркетинга задает концепцию продукта, а разработчики пишут требования. Маркетологи говорят разработчикам, не упустили ли те чего-либо. Если направление продукта меняется, отдел маркетинга не забывает сказать об этом разработчикам.
- в) Представители отдела маркетинга или клиентов говорят разработчикам, какие функции и возможности должен содержать продукт.
- г) Мы проводим организованные собеседования или семинары по выявлению требований с участием представителей различных классов пользователей продукта.

Как документируются требования к ПО?

- а) Наши требования к ПО складываются из воспоминаний очевидцев, электронных и голосовых сообщений и заметок, сделанных во время собеседований и собраний.
- б) Мы составляем описания в произвольной текстовой форме или рисуем диаграммы вариантов использования и классов.
- в) Мы записываем требования на структурированном естественном языке с последовательным уровнем детализации, в соответствии со стандартным шаблоном спецификации требований к ПО. Иногда мы дополняем эти требования графическими моделями анализа с применением стандартных пояснений.
- г) Мы храним свои требования в базе данных или коммерческом инструментальном средстве управления требованиями, а модели анализа — в коммерческом инструментальном средстве, вместе с каждым требованием хранятся несколько его атрибутов.

Как выявляются и документируются такие нефункциональные требования, как атрибуты качества ПО?

- а) А что такое «Атрибуты качества ПО»?
- б) Мы проводим бета-тестирование, чтобы получить ответ пользователей, насколько им понравился продукт.
- в) Мы документируем некоторые атрибуты, такие, как производительность, простота использования и требования к защищенности.
- г) Мы работаем с клиентами для выявления важных атрибутов качества каждого продукта, которые затем документируем методом, обеспечивающим точность и возможность проверки.

Как идентифицируются отдельные функциональные требования?

- а) Мы пишем абзацы пояснительного текста; конкретные требования не идентифицируются по отдельности.
- б) Мы используем маркированные или нумерованные списки.
- в) Мы используем иерархическую схему нумерации, например «3.1.2.4».
- г) Каждое конкретное требование имеет свой уникальный, содержательный идентификатор, не изменяющийся при добавлении, перемещении или удалении других требований.

Как определяются приоритеты требований?

- а) Все требования важны, иначе мы бы вообще их не стали записывать.
- б) Клиенты говорят нам, какие требования для них более важные.
- в) Клиенты приходят к единому мнению о разделении всех требований на категории высокого, среднего и низкого приоритета,
- г) Мы принимаем решения о приоритетах при помощи аналитического процесса, посредством которого определяем ценность для клиента, стоимость и технический риск каждого варианта использования, функции или функционального требования

Какие методы используются для подготовки частичного решения и подтверждения единого понимания проблемы?

- а) Никакие. Мы просто создаем систему.
- б) Мы создаем несколько простых прототипов и запрашиваем ответную реакцию пользователей. Иногда нам приходится выпускать прототип как окончательный продукт.
- в) Мы создаем прототипы и для имитации пользовательского интерфейса, и для технического испытания концепции, когда это целесообразно.
- г) Наши планы по проектам включают задачи создания электронных или бумажных одноразовых прототипов для уточнения требований.

Как проверяются требования?

- а) Мы думаем, что неплохо пишем требования с первого раза.
- б) Мы даем почитать документацию требований разным людям, чтобы узнать их мнения.
- в) Аналитик и некоторые заинтересованные в проекте лица проводят неформальные проверки.

- г) Мы проверяем свою документацию и модели требований, привлекая к этому клиентов, разработчиков и тестировщиков.

Как обозначаются различные версии документации требований?

- а) Автоматически генерируется дата распечатки документа.
- б) Мы используем последовательный номер, например 1.0, 1.1 и т.д., для каждой версии документа.
- в) У нас есть ручная схема идентификации, отделяющая черновые версии от основных и крупные изменения от мелких.
- г) Управление версиями документации требований выполняет система управления конфигурацией, в которой они хранятся, либо требования хранятся в инструментальном средстве управления требованиями, ведущем историю изменений каждого требования

Как производится трассирование требований к оригиналу?

- а) Никак,
- б) Мы знаем, как появилось большинство требований.
- в) У каждого требования есть установленный источник.
- г) У нас установлено двустороннее трассирование между каждым требованием к ПО и каким-либо заявлением представителей клиентов, системным требованием, вариантом использования, бизнес-правилом, архитектурным требованием или другим источником.

Как требования используются для разработки планов проекта?

- а) Дата выпуска продукта устанавливается до того, как мы начинаем собирать требования. Мы не можем изменить ни расписание работы по проекту, ни требования.
- б) Мы проходим через фазу быстрого свертывания масштабов незадолго до даты выпуска.
- в) Первая итерация плана работы над проектом устанавливает сроки сбора требований. Оставшаяся часть плана проекта разрабатывается после того, как получено предварительное понимание требований.
- г) Мы вырабатываем сроки и планы на предполагаемых затратах, необходимых для реализации требуемой функциональности. Эти планы обновляются по мере изменения требований.

Как требования используются для конструирования?

- а) Если бы мы имели записанные требования, мы бы обращались к ним при программировании.
- б) В документации требований описаны решения, которые мы намереемся реализовать.
- в) Каждое функциональное требование трассируется к элементу конструкции,
- г) Конструкторы проверяют, может ли спецификация требований служить основой для конструирования.

Как требования используются для тестирования?

- а) Между тестированием и требованиями нет прямой взаимосвязи.
- б) Тестировщики проверяют то, что, по словам разработчиков, реализовано.
- в) Мы составляем варианты тестирования системы на основе вариантов использования и функциональных требований.
- г) Тестировщики проверяют, поддаются ли требования, документированные в спецификации, проверке, и создают планы тестирования. Один из параметров измерения прогресса тестирования системы — по охвату требований.

Как определяется основная версия требований для каждого проекта и как ею управляют?

- а) А что такое «основная версия»?
- б) Клиенты и менеджеры объявляют требования законченными, но разработчикам все равно приходит много запросов на изменения и жалоб.
- в) Мы определяем в спецификации первоначальную основную версию требований, но не всегда вовремя обновляем ее по мере внесения изменений.
- г) Требования хранятся в базе данных, когда определяется первоначальная основная версия. База данных и спецификация требований обновляются каждый раз, когда утверждаются изменения в требованиях.

Как управляют изменениями в требованиях?

- а) Бесконтрольные изменения попадают в проект всякий раз, как у кого-то появляется новая идея или когда кто-то понимает, что он что-то забыл.
- б) Мы боремся с изменениями, замораживая требования после завершения фазы разработки требований, но все равно неформальные соглашения об изменениях имеют место.

- в) Мы используем утвержденный формат и центральный пункт подачи запросов на изменения. Менеджер проекта решает, какие изменения утвердить.
- г) Изменения вносятся согласно нашему документированному технологическому процессу управления изменениями. Последствия каждого изменения оцениваются прежде, чем совет по управлению изменениями принимает решение о его одобрении.

Краткое описание и регламент выполнения

К тестам допускаются все студенты.

По результатам итогового теста студент может набрать максимально 100 баллов.

Критерии оценки:

Максимальное количество баллов – 8 б. (баллы студенту начисляются автоматически пропорционально выполненным тестовым заданиям)

7.2.2. Отчет по практическим заданиям

(наименование оценочного средства)

Типовой пример задания

Практическая работа 1. Оформление технического задания

Форма отчета по практической работе № 1. В отчет по практической работе должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы;
- ответы на контрольные вопросы.

Практическая работа 2. Построение диаграммы вариантов использования.

Форма отчета по практической работе № 2. В отчет по практической работе должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы;
- ответы на контрольные вопросы.

Практическая работа 3. Построение объектно-ориентированной модели проекта

Форма отчета по практической работе № 3. В отчет по практической работе должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы;
- ответы на контрольные вопросы.

Практическая работа 4. Создание прототипа программного обеспечения проекта по выбранной теме

Форма отчета по практической работе № 4. В отчет по практической работе должны быть включены следующие пункты:

- титульный лист;
- цель работы;
- краткие теоретические сведения;
- описание хода выполнения работы;
- результаты выполненной работы;
- ответы на контрольные вопросы.

Краткое описание и регламент выполнения

К выполнению практических работ допускаются все студенты.

Выполняются работы на ПК с использованием программного обеспечения согласно индивидуальному варианту задания, предусмотренного в методических рекомендациях.

Критерии оценки:

Формы текущего контроля	Критерии и нормы оценки
Отчет по практической работе 1-4	15 баллов – задание выполнено в полном объеме без замечаний 10 баллов – задание выполнено в полном объеме, присутствуют замечания по выполнению задания 5 баллов – задание выполнено не в полном объеме, присутствуют несущественные замечания 2 балла – задание выполнено не в полном объеме, присутствуют замечания по выполнению задания 0 баллов – задание не выполнено

7.3. Оценочные средства для промежуточной аттестации по итогам освоения дисциплины

7.3.1. Вопросы к промежуточной аттестации

Семестр 4

№ п/п	Вопросы к экзамену
1.	Что такое «требование»
2.	Типы требований
3.	Выявление бизнес-требований и бизнес-правил
4.	Выявление ожиданий заинтересованных лиц
5.	Выявление пользовательских требований
6.	Выявление требований к пользовательскому интерфейсу
7.	Определение границ системы
8.	Выявление функциональных требований
9.	Выявление нефункциональных требований
10.	Модель прецедентов использования
11.	Модель предметной области
12.	Процесс разработки и анализа требований
13.	План управления требованиями
14.	Атрибуты требований
15.	Управление состояниями требований
16.	Приоритезация требований
17.	Спецификации требований
18.	Принципы тестирования требований
19.	Согласование и утверждение требований
20.	Управление изменениями требований
21.	План управления требованиями
22.	Общий процесс управления требованиями.
23.	Основные виды документов по управлению требованиями.
24.	Основные этапы процесса разработки требований
25.	Бизнес-требования. С чего начать?
26.	Классификации требований. Атрибуты требований. Атрибуты качества требований.
27.	Место процесса управления требованиями в жизненном цикле ПО. Управление изменениями.
28.	Пользовательские требования. Определение пользователей, выявление потребностей.
29.	Практические аспекты разработки пользовательских требований.
30.	Основные источники требований.
31.	Методы выявления и проектирования требований. Проблемы выявления требований и способы их решения.
32.	Выявление скрытых ограничений.
33.	Особенности создание системных требований.
34.	Спецификация требований. Типовые шаблоны документов. Критерии для написания текста требований.
35.	Функциональные требования.
36.	Нефункциональные требования. Типы нефункциональных требований.
37.	Атрибуты качества ПО.

№ п/п	Вопросы к экзамену
38.	Обзор систем автоматизированного управления требованиями.
39.	Современные методы выявления требований: прототипирование, совместная разработка приложений (JAD-метод), быстрая разработка приложений (RAD-метод) и др
40.	Преимущества использования юзкейсов для описания функциональных требований
41.	Назначение и содержание технического задания. 16. Функции сотрудников, участвующих в разработке документации
42.	Техники для выявления требований и потребностей. Классификация требований.
43.	Спецификация требований к ПО. Языки спецификаций. Критерии хороших требований
44.	Документирование требований с применением сценариев использования. Бизнес-правила.
45.	Процесс управления изменениями в традиционной и Agile разработке Приоритетизация требований.
46.	Анализ требований и определение спецификаций при структурном подходе
47.	Анализ требований и определение спецификаций при объектном подходе
48.	Особенности интерпретации требований
49.	Разработка требований. Подход с применением варианта использования продукта
50.	Инструментальные средства управления требованиями

7.3.2. Критерии и нормы оценки

Семестр	Форма проведения промежуточной аттестации	Критерии и нормы оценки	
4	Экзамен (по накопительному рейтингу)	«отлично»	рейтинговый балл 80-100
		«хорошо»	рейтинговый балл 65-79
		«удовлетворительно»	рейтинговый балл 41-64
		«неудовлетворительно»	рейтинговый балл 0-40

8. Учебно-методическое и информационное обеспечение дисциплины

8.1. Обязательная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно- методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
1.	Васильев, Р. Б.	Управление развитием информационных систем : учебник / Р. Б. Васильев, Г. Н. Калянов, Г. А. Левочкина. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 507 с. — ISBN 978-5-4497-0561-7.	учебник	2020	ЭБС "IPRbooks"
2.	Поляков, Е. А.	Управление жизненным циклом информационных систем : учебное пособие / Е. А. Поляков. — Саратов : Вузовское образование, 2019. — 193 с. — ISBN 978-5-4487-0490-1.	учеб. пособие	2019	ЭБС "IPRbooks"
3.	Мякишев, Д. В.	Разработка программного обеспечения АСУ ТП на основе объектно-ориентированного подхода : методическое пособие / Д. В. Мякишев. — Москва : Инфра-Инженерия, 2019. — 128 с. — ISBN 978-5-9729-0305-4.	методическое пособие	2019	ЭБС "IPRbooks"
4.	Тагайцева, С. Г.	Разработка прикладных решений на платформе 1С: Предприятие 8 : учебное пособие / С. Г. Тагайцева, Т. В. Юрченко. — Нижний Новгород : Нижегородский государственный архитектурно-строительный университет, ЭБС АСВ, 2016. — 85 с. — ISBN 978-5-528-00146-3.	учеб. пособие	2016	ЭБС "IPRbooks"

8.2. Дополнительная литература

№ п/п	Авторы, составители	Заглавие (заголовок)	Тип (учебник, учебное пособие, учебно- методическое пособие, практикум, др.)	Год издания	Количество в научной библиотеке / Наименование ЭБС
5.	Гаряева, В. В	Решение задач с использованием пакетов прикладных программ : учебное пособие / В. В. Горяева. — Москва : Московский государственный строительный университет, Ай Пи Эр Медиа, ЭБС АСВ, 2017. — 90 с. — ISBN 978-5-7264-1788-2.	учеб. пособие	2017	ЭБС "IPRbooks"
6.	Лоскутов, В. И.	Разработка информационных систем для Windows Store / В. И. Лоскутов, И. Л. Коробова. — 2-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 179 с. — ISBN 2227-8397. —		2016	ЭБС "IPRbooks"
7.	Стешин А. И.	Информационные системы в организации [Электронный ресурс] : учебное пособие / А. И. Стешин. - 2-е изд. - Саратов : Вузовское образование, 2019. - 194 с. - ISBN 978-5-4487-0385-0.	учеб. пособие	2019	ЭБС "IPRbooks"
8.	Голицына О. Л.	Информационные системы [Электронный ресурс] : учеб. пособие / О. Л. Голицына, Н. В. Максимов, И. И. Попов. - 2-е изд. - Москва : Форум : ИНФРА-М, 2018. - 448 с. : ил. - (Высшее образование). - ISBN 978-5-91134-833-5.	учеб. пособие	2018	ЭБС "Znanium.com"

8.3. Перечень профессиональных баз данных и информационных справочных систем

– Web of Science [Электронный ресурс] : мультидисциплинарная реферативная база данных. – Philadelphia: ClarivateAnalytics, 2016– . – Режим доступа : apps.webofknowledge.com. – Загл. с экрана. – Яз. рус., англ.

Scopus [Электронный ресурс] : реферативная база данных. – Netherlands: Elsevier, 2004– . – Режим доступа : scopus.com. – Загл. С экрана. – Яз. рус., англ.

Elibrary [Электронный ресурс] : научная электронная библиотека. – Москва : НЭБ, 2000– . – Режим доступа : elibrary.ru. – Загл. с экрана. – Яз. рус., англ.

8.4. Перечень программного обеспечения

№ п/п	Наименование ПО	Реквизиты договора (дата, номер, срок действия)
	Eclipse Foundation Eclipse версия 4	неограниченный
	NetBeans Community NetBeans IDE версия8	неограниченный
	The CodeBlocks team CodeBlocks версия16	неограниченный
	Windows	Бессрочная
	OfficeStandart	Бессрочная

8.5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

№ п/п	Наименование оборудованных учебных кабинетов, лабораторий, мастерских и др. объектов для проведения практических и лабораторных занятий, помещений для самостоятельной работы обучающихся (номер аудитории)	Перечень основного оборудования
	Аудитория вебконференций. Учебная аудитория для проведения занятий лекционного типа. Учебная аудитория для проведения занятий семинарского типа. Учебная аудитория для курсового проектирования (выполнения курсовых работ). Учебная аудитория для проведения групповых и индивидуальных консультаций Учебная аудитория для проведения занятий текущего контроля и промежуточной аттестации. (УЛК-807)	Экран телевизионный, ширмы, проектор на штативе. стол преподавательский, стулья преподавательские., Транспарант-перетяжка, системный блок.