

Документ подписан простой электронной подписью
Информация о владельце: МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФИО: Силин Яков Петрович
Должность: Ректор
Дата подписания: 08.09.2023 10:50:04
Уникальный программный ключ:
24f866be2aca16484036a8c8b9c5e309931af051

Одобрена
на заседании кафедры

22.11.2022 г.
протокол № 4
И.о. зав. кафедрой Кормышев В.М.

Утверждена
Советом по учебно-методическим
вопросам и качеству образования

14 декабря 2022 г.
протокол № 4
Председатель



Карх Д.А.
(подпись)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Наименование дисциплины	Проектирование архитектуры программных систем
Направление подготовки	09.03.01 Информатика и вычислительная техника
Профиль	Программное обеспечение автоматизированных систем
Форма обучения	очная
Год набора	2023
Разработана:	
Доцент, к.э.н.	
Кислицын Е.В.	

Екатеринбург
2022 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ	3
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП	3
3. ОБЪЕМ ДИСЦИПЛИНЫ	3
4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ОПОП	3
5. ТЕМАТИЧЕСКИЙ ПЛАН	5
6. ФОРМЫ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ШКАЛЫ ОЦЕНИВАНИЯ	5
7. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ	7
8. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ	12
9. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ	12
10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ЛИЦЕНЗИОННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ, ОНЛАЙН КУРСОВ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ	13
11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ	14

ВВЕДЕНИЕ

Рабочая программа дисциплины является частью основной профессиональной образовательной программы высшего образования - программы бакалавриата, разработанной в соответствии с ФГОС ВО

ФГОС ВО	Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.01 Информатика и вычислительная техника (приказ Минобрнауки России от 19.09.2017 г. № 929)
ПС	

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

изучение современных инженерных принципов (методов) создания надежного, качественного программного обеспечения, удовлетворяющего предъявляемым к нему требованиям; формирование у студентов умений и навыков применения методов программной инженерии.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина относится к вариативной части учебного плана.

3. ОБЪЕМ ДИСЦИПЛИНЫ

Промежуточный контроль	Часов					3.е.
	Всего за семестр	Контактная работа .(по уч.зан.)			Самостоятельная работа в том числе подготовка контрольных и курсовых	
		Всего	Лекции	Лабораторные		
Семестр 7						
Экзамен, Курсовая работа	252	56	28	28	160	7

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ОПОП

В результате освоения ОПОП у выпускника должны быть сформированы компетенции, установленные в соответствии ФГОС ВО.

Профессиональные компетенции (ПК)

Шифр и наименование компетенции	Индикаторы достижения компетенций
проектный	
ПК-1 Анализ требований к программному обеспечению	ИД-1.ПК-1 Знать: Возможности существующей программно-технической архитектуры; Возможности современных и перспективных средств разработки программных продуктов, технических средств; Методологии разработки программного обеспечения и технологии программирования; Методологии и технологии проектирования и использования баз данных
	ИД-2.ПК-1 Уметь: Проводить анализ исполнения требований; Вырабатывать варианты реализации требований; Проводить оценку и обоснование рекомендуемых решений; Осуществлять коммуникации с заинтересованными сторонами

ПК-1 Анализ требований к программному обеспечению	ИД-3.ПК-1 Иметь практический опыт: анализа возможностей реализации требований к программному обеспечению; Оценка времени и трудоемкости реализации требований к программному обеспечению; Согласование требований к программному обеспечению с заинтересованными сторонами; Оценка и согласование сроков выполнения поставленных задач
ПК-2 Разработка технических спецификаций на программные компоненты и их взаимодействие	ИД-1.ПК-2 Знать: Языки формализации функциональных спецификаций; Методы и приемы формализации задач; Принципы и методы стоимостной оценки разработки программного обеспечения; Жизненный цикл проекта программного обеспечения; Процессы и стандарты управления проектом
	ИД-2.ПК-2 Уметь: Выбирать средства реализации требований к программному обеспечению; Вырабатывать варианты реализации программного обеспечения; Проводить оценку и обоснование рекомендуемых решений; Осуществлять коммуникации с заинтересованными сторонами; Применять навыки финансового моделирования для исследования роста софтверных и интернет компаний; Управлять сроками, стоимостью и качеством проекта по разработке программного обеспечения
	ИД-3.ПК-2 Иметь практический опыт: разработки и согласования технических спецификаций на программные компоненты и их взаимодействие с архитектором программного обеспечения; Распределение заданий между программистами в соответствии с техническими спецификациями; Осуществление контроля выполнения заданий; Осуществление обучения и наставничества; Формирование и предоставление отчетности в соответствии с установленными регламентами; Оценка и согласование сроков выполнения поставленных задач
ПК-3 Проектирование программного обеспечения	ИД-1.ПК-3 Знать: Принципы построения архитектуры программного обеспечения и виды архитектуры программного обеспечения; Типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; Методы и средства проектирования программного обеспечения; Методы и средства проектирования баз данных; Методы и средства проектирования программных интерфейсов
	ИД-2.ПК-3 Уметь: Использовать существующие типовые решения и шаблоны проектирования программного обеспечения; Применять методы и средства проектирования программного обеспечения, структур данных, баз данных, программных интерфейсов; Осуществлять коммуникации с заинтересованными сторонами
	ИД-3.ПК-3 Иметь практический опыт: разработки, изменения и согласования архитектуры программного обеспечения с системным аналитиком и архитектором программного обеспечения; Проектирование структур данных; Проектирование баз данных; Проектирование программных интерфейсов; Оценка и согласование сроков выполнения поставленных задач

5. ТЕМАТИЧЕСКИЙ ПЛАН

Тема	Часов						
	Наименование темы	Всего часов	Контактная работа (по уч.зан.)			Самост. работа	Контроль самостоятельной работы
			Лекции	Лабораторные	Практические занятия		
Семестр 7		216					
Тема 1.	Архитектура программных систем. Унифицированный процесс разработки.	25	6	4		15	
Тема 2.	Требования к программной системе.	19		4		15	
Тема 3.	Классы и объекты. Модель предметной области.	19		4		15	
Тема 4.	Анализ и проектирование. Рабочий поток проектирования.	30	6	4		20	
Тема 5.	Документирование проекта архитектуры программной системы.	26	4	2		20	
Тема 6.	Прямое и обратное проектирование.	21	4	2		15	
Тема 7.	Архитектурные стили и шаблоны.	26	4	2		20	
Тема 8.	Шаблоны проектирования GRASP. Шаблоны проектирования GoF.	26	2	4		20	
Тема 9.	Архитектурные стили сетевых и распределенных программных систем.	24	2	2		20	

6. ФОРМЫ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ШКАЛЫ ОЦЕНИВАНИЯ

Раздел/Тема	Вид оценочного средства	Описание оценочного средства	Критерии оценивания
Текущий контроль (Приложение 4)			
Темы 1-2	Тест (приложение 4)	Тест содержит 29 вопросов	10 баллов
Темы 3-4	Практическая работа (приложение 4)	Содержит практические задания	10 баллов
Темы 5-6	Практическая работа (приложение 4)	Содержит практические задания	10 баллов
Промежуточный контроль (Приложение 5)			
7 семестр (Эк)	Экзаменационный билет	Содержит 2 теоретических вопроса и 1 практический.	100 баллов
7 семестр (КР)	Курсовая работа	Перечень курсовых работ (Приложение 3), Методические рекомендации по выполнению курсовой работы по дисциплине (Приложение 7).	100 баллов

ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

Показатель оценки освоения ОПОП формируется на основе объединения текущей и промежуточной аттестации обучающегося.

Показатель рейтинга по каждой дисциплине выражается в процентах, который показывает уровень подготовки студента.

Текущая аттестация. Используется 100-балльная система оценивания. Оценка работы студента в течении семестра осуществляется преподавателем в соответствии с разработанной им системой оценки учебных достижений в процессе обучения по данной дисциплине.

В рабочих программах дисциплин и практик закреплены виды текущей аттестации, планируемые результаты контрольных мероприятий и критерии оценки учебных достижений.

В течение семестра преподавателем проводится не менее 3-х контрольных мероприятий, по оценке деятельности студента. Если посещения занятий по дисциплине включены в рейтинг, то данный показатель составляет не более 20% от максимального количества баллов по дисциплине.

Промежуточная аттестация. Используется 5-балльная система оценивания. Оценка работы студента по окончанию дисциплины (части дисциплины) осуществляется преподавателем в соответствии с разработанной им системой оценки достижений студента в процессе обучения по данной дисциплине. Промежуточная аттестация также проводится по окончанию формирования компетенций.

Порядок перевода рейтинга, предусмотренных системой оценивания, по дисциплине, в пятибалльную систему.

Высокий уровень – 100% - 70% - отлично, хорошо.

Средний уровень – 69% - 50% - удовлетворительно.

Показатель оценки	По 5-балльной системе	Характеристика показателя
100% - 85%	отлично	обладают теоретическими знаниями в полном объеме, понимают, самостоятельно умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов на высоком уровне
84% - 70%	хорошо	обладают теоретическими знаниями в полном объеме, понимают, самостоятельно умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов. Могут быть допущены недочеты, исправленные студентом самостоятельно в процессе работы (ответа и т.д.)
69% - 50%	удовлетворительно	обладают общими теоретическими знаниями, умеют применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов на среднем уровне. Допускаются ошибки, которые студент затрудняется исправить самостоятельно.
49 % и менее	неудовлетворительно	обладают не полным объемом общих теоретическими знаниями, не умеют самостоятельно применять, исследовать, идентифицировать, анализировать, систематизировать, распределять по категориям, рассчитать показатели, классифицировать, разрабатывать модели, алгоритмизировать, управлять, организовать, планировать процессы исследования, осуществлять оценку результатов. Не сформированы умения и навыки для решения профессиональных задач
100% - 50%	зачтено	характеристика показателя соответствует «отлично», «хорошо», «удовлетворительно»
49 % и менее	не зачтено	характеристика показателя соответствует «неудовлетворительно»

7. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

7.1. Содержание лекций

Тема 1. Архитектура программных систем. Унифицированный процесс разработки. Вводная информация. Понятие архитектуры программной информационной системы (ИС). Роль архитектуры в контексте процесса разработки программного обеспечения. Проект и архитектура. Цель проектирования архитектуры ИС. Парадигмы проектирования: структурная, объектно-ориентированная, функциональная, логическая. Объектно-ориентированный анализ и проектирование (ООАиП). Унифицированный процесс (УП) разработки программного обеспечения (ПО). Методология IBM Rational Unified Process (IBM RUP) как пример унифицированного процесса.

Тема 4. Анализ и проектирование. Рабочий поток проектирования. Обзор дисциплины «Анализ и проектирование». Входные и выходные артефакты (рабочие материалы) дисциплины. Роли и задачи дисциплины. Назначение и задачи этапов дисциплины. Используемые диаграммы языка UML в задачах проектирования. Роль архитектора в проекте и выполняемые им задачи. Ключевые абстракции и их идентификация. Диаграммы UML для визуализации ключевых абстракций. Архитектурные механизмы, их назначение. Архитектурные шаблоны и их использование. Понятие слоя и его представление в языке UML. Структура «реализация прецедента» (use case realization) и ее представление в визуальной модели.

Тема 5. Документирование проекта архитектуры программной системы. Документирование проектируемой архитектуры ИС. Способы описания и документирования архитектуры ИС. Архитектурный документ как артефакт проекта. Представление архитектуры N+1 (4+1). Документ программной архитектуры (Software Architecture Document, SAD). Архитектурные представления. Архитектурные факторы. Архитектурные решения. Применение и оформление диаграмм UML в документации проекта. Государственный стандарт ГОСТ Р 57100-2016. Использование этого стандарта.

Тема 6. Прямое и обратное проектирование. Лучшие практики разработки ИС. Назначение лучших практик. Связь практик с визуальным моделированием. Инструментальная поддержка практик. Понятие компонента архитектуры ИС. Интерфейс компонента. Версионный контроль основных артефактов проекта. Понятие конфигурации. CASE-средства визуального моделирования. Прямое и обратное проектирование.

Тема 7. Архитектурные стили и шаблоны. Понятия архитектурного стиля, шаблона, фреймворка. Отличия архитектурного шаблона от шаблона проектирования. Популярные стили и шаблоны архитектуры ИС. Шаблон «Слой». Трёхзвенная (многозвенная) архитектура. Шаблоны «модель-представление-контроллер» (MVC), «сенсор-вычислитель-контроллер» (SCC). Стили «программа и подпрограммы», «виртуальная машина», «клиент-сервер», «каналы-и-фильтры», «классная доска», «издатель-подписчики», «событийная архитектура», «пиринговая архитектура», «конечный автомат». Плюсы и минусы различных архитектурных стилей. Соответствие стиля задаче.

Тема 8. Шаблоны проектирования GRASP. Шаблоны проектирования GoF. Шаблоны проектирования GRASP. Шаблоны: Creator, Information Expert, Low Coupling, Controller, High Cohesion, Polymorphism, Pure Fabrication, Indirection, Protected Variations. Шаблоны проектирования GoF. Шаблоны: Adapter, Factory, Singleton, Strategy, Composite, Facade, Observer и другие. Связь между шаблонами проектирования и архитектурными стилями. Применение шаблонов проектирования. Принципы проектирования: SRP, OCP, LSP, ISP, DIP. Адаптация и поддержка архитектуры ИС. Тестирование и рефакторинг при проектировании архитектуры. Итеративность и инкрементальность при проектировании архитектуры ИС. Элементы гибких процессов проектирования.

Тема 9. Архитектурные стили сетевых и распределенных программных систем. Проектирование архитектуры сетевых и распределенных программных систем. Архитектурный стиль «пиринговая архитектура» (peer-to-peer). Архитектурный стиль Representational state transfer (REST). Связь архитектурного стиля REST и устройства всемирной сети Интернет. Архитектурный шаблон «Удаленный вызов процедур» (RPC). Веб-сервисы. Стил «Сервис-ориентированная архитектура приложений» (SOA). Проектирование архитектуры мобильных и встроенный приложений.

7.2 Содержание практических занятий и лабораторных работ

Тема 1. Архитектура программных систем. Унифицированный процесс разработки.

Основные принципы визуального моделирования. Сложность ПО и архитектурные представления. Статические и динамические диаграммы языка UML. Репозиторий модели CASE-средства.

Тема 2. Требования к программной системе.

Визуальное представление на UML модели прецедентов (сценариев использования). Словарь проекта. Концепция ИС. Модель прецедентов.

Тема 3. Классы и объекты. Модель предметной области.

Используемые UML-диаграммы, шаблоны при их построении. Понятие состояния объекта и его визуальное представление. Поиск экторов в модели предметной области.

Тема 4. Анализ и проектирование. Рабочий поток проектирования.

Выявление классов-участников. Применение прецедентов для проектирования объектной динамики. Стереотипы «граничный» («boundary»), «управляющий» («control»), «сущность» («entity») классов-участников реализации. Шаблоны для идентификации классов. Диаграммы последовательности и коммуникации UML. Правила использования стереотипов «boundary», «control», «entity» при проектировании объектной динамики. Диаграмма классов-участников (VOPC). Правила идентификации операций (ответственностей) классов. Правила идентификации отношений между классами – ассоциаций и зависимостей.

Понятие заинтересованных лиц. Детальность проекта. Архитектурные шаблоны и их использование в архитектурном проекте. Использование модели предметной области при решении задачи проектирования. Пакеты в UML. Связи между пакетами UML и их использование в представлении архитектуры. Задача архитектора «Идентификация проектных элементов». Определение классов, подлежащих декомпозиции. Упаковка концептуальных классов по пакетам проекта. Зависимости между пакетами. Архитектурный шаблон «Слой» и его использование при проектировании ИС. Задача архитектора «Идентификация проектных механизмов». Понятие проектного шаблона. Представление шаблонов в CASE-средстве. Типы архитектурных механизмов и их представление в визуальной модели.

Задача проектировщика «Проектирование прецедентов». Реализация прецедента на этапе проектирования. Применение архитектурных механизмов. Представление компонентов в проекте.

Использование интерфейсов на диаграммах последовательности UML. Задача проектировщика «Проектирование подсистем». Отличие пакетов и подсистем. Структурный класс UML и его использование. Операции интерфейса и их реализация в проекте подсистемы. Диаграмма последовательности UML. Динамическое и статическое представление дизайна подсистемы.

Использование проектных механизмов. Зависимости классов и пакетов.

Проектные классы. Ответственности и операции классов. Выбор атрибутов классов. Атрибуты и понятие состояния объекта. Шаблоны выявления состояний. Специальные состояния – начальное и конечное. Супер состояние и его использование. Диаграммы состояний UML. Условия на диаграмме состояний. Правила перехода из состояния в состояние и из запись на диаграмме. Диаграммы деятельности языка UML. Их использование в рамках унифицированного процесса. Семантика деятельности. События, исключения. Конечные автоматы. Конечные автоматы в рамках унифицированного процесса. Конечные автоматы и язык UML. Состояния и переходы. Другие языки моделирования поведения: сети Петри, модели BPMN.

Тема 5. Документирование проекта архитектуры программной системы.

Нефункциональные требования. Эффективность, сложность, масштабируемость, расширяемость, адаптируемость, безопасность, зависимости системы. Учёт нефункциональных требований в проекте архитектуры ИС. Трассируемость требований в архитектуре при проектировании.

Объектный язык ограничений OCL. Возможности OCL и область применения. Синтаксис и типы выражений. OCL в диаграммах UML.

Тема 6. Прямое и обратное проектирование.

Различия рисования и визуального моделирования. Репозиторий CASE-средства и синхронизация его содержимого на UML диаграммах. Понятие каркасного кода при прямом проектировании. Сравнение возможностей CASE-средств визуального моделирования.

Тема 7. Архитектурные стили и шаблоны.

Описание и оценка архитектурного стиля. Архитектурные фреймворки: 4+1, TOGAF, RM-ODP, SOMF.

<p>Тема 8. Шаблоны проектирования GRASP. Шаблоны проектирования GoF.</p> <p>Реализация и развертывание ИС. Рабочий поток реализации унифицированного процесса. Учёт особенностей реализации и развертывания при проектировании архитектуры ИС. Связь программного и аппаратного обеспечения. Влияние особенностей аппаратного обеспечения на программную архитектуру. Артефакты реализации. Диаграммы развертывания языка UML. Применение диаграмм развертывания для проектирования процесса развертывания и адаптации ИС.</p>
<p>Тема 9. Архитектурные стили сетевых и распределенных программных систем.</p> <p>Защищенность как ключевое нефункциональное требование к ИС. Проектирование защищенных ИС. Принципы и шаблоны проектирования архитектуры для обеспечения защищенности программной системы. Контроль доступа. Проектирование и разработка программной системы — командная работа. Роли и распределение задач в команде. Значение организации процесса при проектировании. Ещё раз о лучших практиках разработки и проектирования: гибкость, итеративность, инкрементальность процесса. История создания и развития языка UML. Группа Object Management Group (OMG). Заключительные замечания.</p>

7.3. Содержание самостоятельной работы

<p>Тема 1. Архитектура программных систем. Унифицированный процесс разработки. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>
<p>Тема 2. Требования к программной системе. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>
<p>Тема 3. Классы и объекты. Модель предметной области. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>
<p>Тема 4. Анализ и проектирование. Рабочий поток проектирования. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>
<p>Тема 5. Документирование проекта архитектуры программной системы. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>
<p>Тема 6. Прямое и обратное проектирование. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>
<p>Тема 7. Архитектурные стили и шаблоны. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>
<p>Тема 8. Шаблоны проектирования GRASP. Шаблоны проектирования GoF. Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.</p>

Тема 9. Архитектурные стили сетевых и распределенных программных систем.
Изучение основной и дополнительной литературы по теме, интернет-источников. Разбор лабораторного практикума. Разработка программных решений. Подготовка к текущему контролю.

7.3.1. Примерные вопросы для самостоятельной подготовки к зачету/экзамену
Приложение 1.

7.3.2. Практические задания по дисциплине для самостоятельной подготовки к зачету/экзамену
Приложение 2.

7.3.3. Перечень курсовых работ
Приложение 3.

7.4. Электронное портфолио обучающегося
Размещается курсовая работа.

7.5. Методические рекомендации по выполнению контрольной работы
Не предусмотрено.

7.6 Методические рекомендации по выполнению курсовой работы
Приложение 7.

8. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

По заявлению студента

В целях доступности освоения программы для лиц с ограниченными возможностями здоровья при необходимости кафедра обеспечивает следующие условия:

- особый порядок освоения дисциплины, с учетом состояния их здоровья;
- электронные образовательные ресурсы по дисциплине в формах, адаптированных к ограничениям их здоровья;
- изучение дисциплины по индивидуальному учебному плану (вне зависимости от формы обучения);
- электронное обучение и дистанционные образовательные технологии, которые предусматривают возможности приема-передачи информации в доступных для них формах.
- доступ (удаленный доступ), к современным профессиональным базам данных и информационным справочным системам, состав которых определен РПД.

9. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Сайт библиотеки УрГЭУ
<http://lib.usue.ru/>

Основная литература:

1. Антипов В.А., Бубнов А.А., Пылькин А.Н., Столчнев В.К. Введение в программную инженерию [Электронный ресурс]: Учебник. - Москва: ООО "КУРС", 2019. - 336 с. – Режим доступа: <https://znanium.com/catalog/product/1035160>

2. Гагарина Л.Г., Кокорева Е. В., Сидорова-Виснадул Б.Д. Технология разработки программного обеспечения [Электронный ресурс]: Учебное пособие. - Москва: Издательский Дом "ФОРУМ", 2022. - 400 с. – Режим доступа: <https://znanium.com/catalog/product/1699927>

Дополнительная литература:

1. Виноградова Е. Ю. Паттерны программирования [Электронный ресурс]: учебное пособие. - Екатеринбург: [Издательство УрГЭУ], 2017. - 27 – Режим доступа: <http://lib.usue.ru/resource/limit/ump/18/p490388.pdf>

2. Плещев В. В., Шишков Е. И. Основы программирования на языках C++ и C# с примерами и упражнениями [Электронный ресурс]: учебное пособие. - Екатеринбург: [Издательство УрГЭУ], 2018. - 286 – Режим доступа: <http://lib.usue.ru/resource/limit/ump/18/p490708.pdf>

3. Федорова Г.Н. Разработка, внедрение и адаптация программного обеспечения отраслевой направленности [Электронный ресурс]: Учебное пособие. - Москва: ООО "КУРС", 2022. - 336 – Режим доступа: <https://znanium.com/catalog/product/1858587>

10. ПЕРЕЧЕНЬ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, ВКЛЮЧАЯ ПЕРЕЧЕНЬ ЛИЦЕНЗИОННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И ИНФОРМАЦИОННЫХ СПРАВОЧНЫХ СИСТЕМ, ОНЛАЙН КУРСОВ, ИСПОЛЬЗУЕМЫХ ПРИ ОСУЩЕСТВЛЕНИИ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Перечень лицензионного программного обеспечения:

Microsoft Visual Studio Community. Лицензия для образовательных учреждений. Срок действия лицензии - без ограничения срока.

MySQL Community Server. Стандартная общественная лицензия GNU (GPL). Срок действия лицензии - без ограничения срока.

Система контроля версий Git. Лицензия GNU GPL v2 and GNU LGPL v2.1. Срок действия лицензии - без ограничения срока.

Microsoft Windows 10 .Договор № 52/223-ПО/2020 от 13.04.2020, Акт № Tr000523459 от 14.10.2020. Срок действия лицензии 30.09.2023.

Microsoft Office 2016. Договор № 52/223-ПО/2020 от 13.04.2020, Акт № Tr000523459 от 14.10.2020 Срок действия лицензии 30.09.2023.

Перечень информационных справочных систем, ресурсов информационно-телекоммуникационной сети «Интернет»:

Справочно-правовая система Гарант. Договор № 58419 от 22 декабря 2015. Срок действия лицензии - без ограничения срока

Справочно-правовая система Консультант+. Срок действия лицензии до 31.12.2023

11. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Реализация учебной дисциплины осуществляется с использованием материально-технической базы УрГЭУ, обеспечивающей проведение всех видов учебных занятий и научно-исследовательской и самостоятельной работы обучающихся:

Специальные помещения представляют собой учебные аудитории для проведения всех видов занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду УрГЭУ.

Все помещения укомплектованы специализированной мебелью и оснащены мультимедийным оборудованием спецоборудованием (информационно-телекоммуникационным, иным компьютерным), доступом к информационно-поисковым, справочно-правовым системам, электронным библиотечным системам, базам данных действующего законодательства, иным информационным ресурсам служащими для представления учебной информации большой аудитории.

Для проведения занятий лекционного типа презентации и другие учебно-наглядные пособия, обеспечивающие тематические иллюстрации.