МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)

УТВЕРЖДАЮ
Директор департамента образования
_____ П. Е. Троян
«___» _____ 20__ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Проектирование и архитектура программных систем

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки / специальность: **09.03.04 Программная инженерия**
Направленность (профиль) / специализация: **Индустриальная разработка программных продуктов**
Форма обучения: **очная**
Факультет: **ФСУ, Факультет систем управления**
Кафедра: **АОИ, Кафедра автоматизации обработки информации**
Курс: **3**
Семестр: **5**
Учебный план набора 2015 года

Распределение рабочего времени

№	Виды учебной деятельности	5 семестр	Всего	Единицы
1	Лекции	28	28	часов
2	Лабораторные работы	36	36	часов
3	Всего аудиторных занятий	64	64	часов
4	Самостоятельная работа	44	44	часов
5	Всего (без экзамена)	108	108	часов
6	Подготовка и сдача экзамена	36	36	часов
7	Общая трудоемкость	144	144	часов
		4.0	4.0	З.Е.

Экзамен: 5 семестр

Томск 2018

Согласована на портале № 35157

Рассмотрена и одобрена на заседании кафедры
протокол № 318 от « 3 » 7 2018 г.

ЛИСТ СОГЛАСОВАНИЯ

Рабочая программа дисциплины составлена с учетом требований федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 09.03.04 Программная инженерия, утвержденного 12.03.2015 года, рассмотрена и одобрена на заседании кафедры АОИ «___» _____ 20__ года, протокол №_____.

Разработчик:
доцент каф. АОИ _____ А. А. Голубева

Заведующий обеспечивающей каф.
АОИ _____ Ю. П. Ехлаков

Рабочая программа дисциплины согласована с факультетом и выпускающей кафедрой:

Декан ФСУ _____ П. В. Сенченко

Заведующий выпускающей каф.
АОИ _____ Ю. П. Ехлаков

Эксперты:

Доцент кафедры автоматизации обработки информации (АОИ) _____ А. А. Сидоров

Доцент кафедры автоматизации обработки информации (АОИ) _____ Н. Ю. Салмина

Согласована на портале № 35157

1. Цели и задачи дисциплины

1.1. Цели дисциплины

Целью изучения дисциплины является получение студентами знаний и приобретение практических
навыков в области проектирования и архитектуры программных систем.

В рамках дисциплины «Проектирование и архитектура программных систем» изучается процесс
проектирования программных систем, технологии разработки интегрированных моделей программных
систем с использованием инструментальных сред проектирования.

Дисциплина нацелена на формирование у студента навыков сбора, обработки и представления
исходных данных для принятия проектных решений; разработки концептуальных, информационно-логических
и функциональных моделей программных систем; объектно-ориентированного
анализа и проектирования.

1.2. Задачи дисциплины

– Задачи дисциплины:
– - ознакомление с основными архитектурными стилями и моделями программных систем;
– - ознакомление с современными методологиями проектирования программных систем;
– - изучение способов проектирования программных систем с использованием языка моделирования
– UML;
– - формирования умений и навыков выработки проектных решений;
– - формирование навыков работы в современных инструментальных средах поддержки процесса
– проектирования программных систем;
– - изучение основных способов документирования проектных решений.

2. Место дисциплины в структуре ОПОП

Дисциплина «Проектирование и архитектура программных систем» (Б1.Б.20) относится к блоку 1 (базовая часть).

Предшествующими дисциплинами, формирующими начальные знания, являются: Введение в программную инженерию, Информатика и программирование, Менеджмент, Объектно-ориентированное программирование.

Последующими дисциплинами являются: Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты, Методы и технологии программирования, Разработка интернет-приложений, Управление ИТ-сервисами и контентом.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

– ОПК-2 владением архитектурой электронных вычислительных машин и систем;
– ОПК-3 готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов;

В результате изучения дисциплины обучающийся должен:

– **знать** знать методы анализа предметной области и проектирования прикладных программных систем;
– **уметь** уметь решать задачи, возникающие на различных фазах жизненного цикла программных систем, связанных с проектирование архитектуры программных систем; разрабатывать концептуальные, информационно-логические и функциональные модели программных систем.
– **владеть** владеть навыками сбора, обработки и представления исходных данных для принятия проектных решений; навыками объектно-ориентированного анализа и проектирования.

Согласована на портале № 35157

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 4.0 зачетных единицы и представлена в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры
		5 семестр
Аудиторные занятия (всего)	64	64
Лекции	28	28
Лабораторные работы	36	36
Самостоятельная работа (всего)	44	44
Выполнение индивидуальных заданий	8	8
Проработка лекционного материала	18	18
Самостоятельное изучение тем (вопросов) теоретической части курса	12	12
Выполнение контрольных работ	6	6
Всего (без экзамена)	108	108
Подготовка и сдача экзамена	36	36
Общая трудоемкость, ч	144	144
Зачетные Единицы	4.0	4.0

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

Названия разделов дисциплины	Лек., ч	Лаб. раб., ч	Сам. раб., ч	Всего часов (без экзамена)	Формируемые компетенции
5 семестр					
1 Введение в проектирование и архитектуру программных систем	2	0	6	8	ОПК-2, ОПК-3
2 Общие концепции проектирования	2	0	8	10	ОПК-2, ОПК-3
3 Объектно-ориентированное проектирование с использование UML	2	12	4	18	ОПК-2, ОПК-3
4 Проектирование с использование структурных диаграмм	6	8	2	16	ОПК-2, ОПК-3
5 Проектирование поведения систем	6	8	2	16	ОПК-2, ОПК-3
6 Проектирование взаимодействия систем	4	0	2	6	ОПК-2, ОПК-3
7 Архитектурные стили и модели	2	0	8	10	ОПК-2, ОПК-3
8 Документирование проектных решений	2	0	10	12	ОПК-2, ОПК-3
9 Переход от проектных решений к программному коду	2	8	2	12	ОПК-2, ОПК-3
Итого за семестр	28	36	44	108	

Согласована на портале № 35157

Итого	28	36	44	108	

5.2. Содержание разделов дисциплины (по лекциям)

Содержание разделов дисциплин (по лекциям) приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов дисциплин (по лекциям)

Названия разделов	Содержание разделов дисциплины (по лекциям)	Трудоемкость, ч	Формируемые компетенции
5 семестр			
1 Введение в проектирование и архитектуру программных систем	Место, цели и задачи этапа проектирования в процессе разработкипрограммных систем. Типы моделей и основные задачи, решаемыес их использованием. Основные методологии проектирования.	2	ОПК-2, ОПК-3
	Итого	2	
2 Общие концепции проектирования	Контекст проектирования. Процесс проектирования (архитектурноепроектирование, детализация архитектуры). Принципы проектирования(абстракция, связанность и соединение, декомпозиция и разбиениена модули, инкапсуляция/ сокрытие информации, разделение интерфейсаи реализации).	2	ОПК-2, ОПК-3
	Итого	2	
3 Объектно-ориентированное проектирование с использование UML	Краткая история UML, сфера применения языка, основные элементыи специфика. Принцип абстрагирования, принцип многомодельности,принцип иерархического построения моделей. Интегрированнаямодель сложной системы. Канонические диаграммы языка	2	ОПК-2, ОПК-3
	Итого	2	
4 Проектирование с использование структурных диаграмм	Структура информационно-логической модели АСОИУ. Разработкафункциональной модели. Место диаграммы прецедентов (use casediagram) в интегрированной модели системы. Цели создания диаграммы.Основные элементы и отношения между ними. Понятиепрецедента (use case) и актера (actor). Абстрактные актеры и прецеденты.Семантика отношений: ассоциации (association relationship),возникающего между актером и прецедентом; обобщения (generalizationrelationship), определенного для актеров и пре-цедентов; включения(include relationship) и расширения (extend re-lationship) междупрецедентами. Влияние установки границ систе-мы на получаемуюмодель. Сравнение диаграмм прецедентов организационных и информационныхсистем. Примеры кор-	6	ОПК-2, ОПК-3

Согласована на портале № 35157

	ректных и некорректных диаграм-м.Рассмотрение типичных ошибок моделирования. Рекомендациипо разработке диаграммы прецедентов.Место диаграммы классов (class diagram) в интегрированной моделисистемы. Цели создания диаграммы. Понятие класса. Спецификациякласса в языке UML (имя класса, имя пакета, секции). Абстрактныеклассы. Спецификация атрибутов класса: видимость,имя, кратность, тип, исходное значение. Спецификация операцийкласса: видимость, имя операции, параметры, возвращаемое значение,указание параллельности выполнения и абстрактности. Отношенияна диаграмме классов: зависимости (dependency relationship);ассоциации (association relationship); обобщения (generalization relationship);реализации (realization relationship).Для каждого типа отношений рассматриваются: семантика, назначение,графическая нотация, стереотипы, кратность, генерируемыепосредством case-средств программный код. Интерфейсы, объекты,параметризованные классы. Примеры построения диаграмм, анализтипичных ошибок, возникающих при построении диаграммы		
	Итого	6	
5 Проектирование поведения систем	Место диаграммы состояний (statechart diagram) в интегрированноймодели системы. Цели создания диаграммы. Понятие автомата.Формализм автомата: состояние и переход. Спецификация состояния.Простые и триггерные переходы. Сигнатура переходов. Подавтоматы.Последовательные и параллельные подавтоматы. Примерыпостроения диаграмм состояний. Разбор типичных ошибок и методових устранения.Место диаграммы деятельности (activity diagram) в интегрированноймодели системы. Цели создания диаграммы. Состояния деятельностии состояние действия. Переходы на диаграмме деятельности.Понятие ветвления, разделения (concurrent fork) и слияния(concurrent join). Использование диаграмм деятельности для моделированиябизнес-процессов. Использование дорожек. Взаимосвязь сдиаграммой прецедентов. Примеры построения диаграмм. Разборнаиболее типичных ошибок и методов их устранения. Общие рекомендациипо построению диа-	6	ОПК-2, ОПК-3

Согласована на портале № 35157

	граммы деятельности.		
	Итого	6	
6 Проектирование взаимодействия систем	Диаграмма последовательности, назначение, основные элементы.Объекты, сообщения. Временные ограничения. Диаграмма кооперацийНазначение, основные элементы. Понятие кооперации. Кооперацияуровня спецификаций и уровня примеров. Объекты, активныеобъекты, мультиобъекты, составные объекты на диаграммекооперации. Связи на диаграмме кооперацийсемантика, стереотипы.Сообщения: назначение, формат записи, стереотипы. Взаимосвязьс диаграммой последовательности и диаграммой классов.Примеры построения диаграмм, разбор типичных ошибок, рекомендациипо построению.	4	ОПК-2, ОПК-3
	Итого	4	
7 Архитектурные стили и модели	Классификация архитектур информационных систем (ИС). МногозвенныеИС. Специализированные подсистемы. РаспределенныеИС. Архитектуры web-приложений.Сервис–ориентированные архитектура (SOA). Функциональныеуровни информационной системы. Интеграция различных информационныхсистем.Выбор архитектуры, факторы влияющие на выбор. Архитектурныйанализ	2	ОПК-2, ОПК-3
	Итого	2	
8 Документирование проектных решений	Основные стандарты документирования программных систем. Отличияотечественных стандартов от зарубежных. Единая системапрограммной документации (ЕСПД). Технический проект на автоматизированнуюсистему (ГОСТ 34). Перечень документов, создаваемыхна стадии «Технический проект». ГОСТ 19.201-78. ЕСПД.Техническое задание. Требования к содержанию и оформлению	2	ОПК-2, ОПК-3
	Итого	2	
9 Переход от проектных решений к программному коду	Взаимосвязь этапов проектирования и разработки программныхсистем. Средства поддержки процессов проектирования программныхсистем. Возможности современных CASE-средств. КлассификацияCASE-средств. Генерация объектно-ориентированного кодана основе UML диаграмм. Возможности современных IDE средств	2	ОПК-2, ОПК-3

Согласована на портале № 35157

	Итого	2	
Итого за семестр		28	

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами

Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами представлены в таблице 5.3.

Таблица 5.3 – Разделы дисциплины и междисциплинарные связи

Наименование дисциплин	№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих и обеспечиваемых дисциплин								
	1	2	3	4	5	6	7	8	9
Предшествующие дисциплины									
1 Введение в программную инженерию	+	+	+	+	+	+	+	+	+
2 Информатика и программирование	+	+	+	+	+	+	+	+	+
3 Менеджмент								+	
4 Объектно-ориентированное программирование	+	+	+	+	+	+	+	+	+
Последующие дисциплины									
1 Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты	+	+	+	+	+	+	+	+	+
2 Методы и технологии программирования	+	+	+	+	+	+	+	+	+
3 Разработка интернет-приложений	+	+	+	+	+	+	+	+	+
4 Управление ИТ-сервисами и контентом	+	+	+	+	+	+	+	+	+

5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий представлено в таблице 5.4.

Таблица 5.4 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Компетенции	Виды занятий			Формы контроля
	Лек.	Лаб. раб.	Сам. раб.	
ОПК-2	+	+	+	Контрольная работа, Отчет по индивидуальному заданию, Экзамен, Отчет по лабораторной работе, Тест
ОПК-3	+	+	+	Контрольная работа, Отчет по индивидуальному заданию, Экзамен, Отчет по лабораторной работе, Тест

6. Интерактивные методы и формы организации обучения

Не предусмотрено РУП.

Согласована на портале № 35157

7. Лабораторные работы

Наименование лабораторных работ приведено в таблице 7.1.

Таблица 7.1 – Наименование лабораторных работ

Названия разделов	Наименование лабораторных работ	Трудоемкость, ч	Формируемые компетенции
5 семестр			
3 Объектно-ориентированное проектирование с использование UML	Создание диаграммы прецедентов	4	ОПК-2, ОПК-3
	Создание диаграмм последовательности и кооперацій	8	
	Итого	12	
4 Проектирование с использование структурных диаграмм	Создание диаграммы классов	8	ОПК-2, ОПК-3
	Итого	8	
5 Проектирование поведения систем	Создание диаграммы состояний и диаграммы деятельности системы	8	ОПК-2, ОПК-3
	Итого	8	
9 Переход от проектных решений к программному коду	Создание диаграмм компонентов и развертывания	8	ОПК-2, ОПК-3
	Итого	8	
Итого за семестр		36	

8. Практические занятия (семинары)

Не предусмотрено РУП.

9. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 9.1.

Таблица 9.1 – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
5 семестр				
1 Введение в проектирование и архитектуру программных систем	Самостоятельное изучение тем (вопросов) теоретической части курса	4	ОПК-2, ОПК-3	Тест, Экзамен
	Проработка лекционного материала	2		
	Итого	6		
2 Общие концепции проектирования	Выполнение контрольных работ	2	ОПК-2, ОПК-3	Контрольная работа, Тест, Экзамен
	Самостоятельное изучение тем (вопросов) теоретической части курса	4		
	Проработка лекционного материала	2		
	Итого	8		
3 Объектно-	Выполнение контроль-	2	ОПК-2, ОПК-3	Контрольная рабо-

Согласована на портале № 35157

ориентированное проектирование с использование UML	ных работ			та, Тест, Экзамен
	Проработка лекционного материала	2		
	Итого	4		
4 Проектирование с использование структурных диаграмм	Проработка лекционного материала	2	ОПК-2, ОПК-3	Тест, Экзамен
	Итого	2		
5 Проектирование поведения систем	Проработка лекционного материала	2	ОПК-2, ОПК-3	Тест, Экзамен
	Итого	2		
6 Проектирование взаимодействия систем	Проработка лекционного материала	2	ОПК-2, ОПК-3	Тест, Экзамен
	Итого	2		
7 Архитектурные стили и модели	Выполнение контрольных работ	2	ОПК-2, ОПК-3	Контрольная работа, Тест, Экзамен
	Самостоятельное изучение тем (вопросов) теоретической части курса	4		
	Проработка лекционного материала	2		
	Итого	8		
8 Документирование проектных решений	Проработка лекционного материала	2	ОПК-2, ОПК-3	Отчет по индивидуальному заданию, Тест, Экзамен
	Выполнение индивидуальных заданий	8		
	Итого	10		
9 Переход от проектных решений к программному коду	Проработка лекционного материала	2	ОПК-2, ОПК-3	Тест, Экзамен
	Итого	2		
Итого за семестр		44		
	Подготовка и сдача экзамена	36		Экзамен
Итого		80		

10. Курсовой проект / курсовая работа

Не предусмотрено РУП.

11. Рейтинговая система для оценки успеваемости обучающихся

11.1. Балльные оценки для элементов контроля

Таблица 11.1 – Балльные оценки для элементов контроля

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр

Согласована на портале № 35157

5 семестр				
Отчет по индивидуальному заданию	5	10	15	30
Отчет по лабораторной работе	5	10	15	30
Тест	2	3	5	10
Итого максимум за период	12	23	35	70
Экзамен				30
Нарастающим итогом	12	35	70	100

11.2. Пересчет баллов в оценки за контрольные точки

Пересчет баллов в оценки за контрольные точки представлен в таблице 11.2.

Таблица 11.2 – Пересчет баллов в оценки за контрольные точки

Баллы на дату контрольной точки	Оценка
≥ 90% от максимальной суммы баллов на дату КТ	5
От 70% до 89% от максимальной суммы баллов на дату КТ	4
От 60% до 69% от максимальной суммы баллов на дату КТ	3
< 60% от максимальной суммы баллов на дату КТ	2

11.3. Пересчет суммы баллов в традиционную и международную оценку

Пересчет суммы баллов в традиционную и международную оценку представлен в таблице 11.3.

Таблица 11.3 – Пересчет суммы баллов в традиционную и международную оценку

Оценка (ГОС)	Итоговая сумма баллов, учитывает успешно сданный экзамен	Оценка (ECTS)
5 (отлично) (зачтено)	90 - 100	A (отлично)
4 (хорошо) (зачтено)	85 - 89	B (очень хорошо)
	75 - 84	C (хорошо)
	70 - 74	D (удовлетворительно)
3 (удовлетворительно) (зачтено)	65 - 69	
	60 - 64	E (посредственно)
2 (неудовлетворительно) (не зачтено)	Ниже 60 баллов	F (неудовлетворительно)

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основная литература

1. Проектирование информационных систем [Электронный ресурс]: Учебное пособие / С. Золотов - 2016. 117 с. - Режим доступа: https://edu.tusur.ru/publications/6478 (дата обращения: 13.08.2018).

12.2. Дополнительная литература

1. Леоненков А.В. Самоучитель UML / А. В. Леоненков. - 2-е изд. - СПб. : БХВ-Петербург, 2006. - 427 с. В библиотеке ТУСУРа: 20 экз. (наличие в библиотеке ТУСУР - 20 экз.)

2. Управление ИТ-сервисами и контентом [Электронный ресурс]: Учебное пособие / Д. Н. Бараксанов, Ю. П. Ехлаков - 2015. 144 с. - Режим доступа: https://edu.tusur.ru/publications/5030 (дата обращения: 13.08.2018).

Согласована на портале № 35157

12.3. Учебно-методические пособия

12.3.1. Обязательные учебно-методические пособия

1. Проектирование и архитектура программных систем [Электронный ресурс]: Методические указания к лабораторным работам и организации самостоятельной работы / А. А. Голубева - 2018. 74 с. - Режим доступа: https://edu.tusur.ru/publications/8294 (дата обращения: 13.08.2018).

12.3.2. Учебно-методические пособия для лиц с ограниченными возможностями здоровья и инвалидов

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа лиц с ограниченными возможностями здоровья и инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

12.4. Профессиональные базы данных и информационные справочные системы

1. http://www.garant.ru/

13. Материально-техническое обеспечение дисциплины и требуемое программное обеспечение

13.1. Общие требования к материально-техническому и программному обеспечению дисциплины

13.1.1. Материально-техническое и программное обеспечение для лекционных занятий

Для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используется учебная аудитория с количеством посадочных мест не менее 22-24, оборудованная доской и стандартной учебной мебелью. Имеются демонстрационное оборудование и учебно-наглядные пособия, обеспечивающие тематические иллюстрации по лекционным разделам дисциплины.

13.1.2. Материально-техническое и программное обеспечение для лабораторных работ

Лаборатория «Программная инженерия»

учебная аудитория для проведения занятий практического типа, учебная аудитория для проведения занятий лабораторного типа, помещение для курсового проектирования (выполнения курсовых работ), помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 409 ауд.

Описание имеющегося оборудования:

- Персональный компьютер Intel Core i3-6300 3.2 ГГц, ОЗУ – 8 Гб, жесткий диск – 500 Гб (10 шт.);
- Проектор Optoma Ex632.DLP;
- Экран для проектора Lumian Mas+Er;
- Магнитно-маркерная доска;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

- Google Chrome
- LibreOffice
- Microsoft Visio 2010

Согласована на портале № 35157

Лаборатория «Бизнес-информатика»

учебная аудитория для проведения занятий практического типа, учебная аудитория для проведения занятий лабораторного типа, помещение для курсового проектирования (выполнения курсовых работ), помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 407 ауд.

Описание имеющегося оборудования:

- Персональный компьютер Intel Core i5-2320 3.0 ГГц, ОЗУ – 4 Гб, жесткий диск – 500 Гб (12 шт.);
- Проектор Optoma Ex632.DLP;
- Экран для проектора Lumian Mas+Er;
- Магнитно-маркерная доска;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

– DosBox 0.74, GNU GPLv2

Лаборатория «Операционные системы и СУБД»

учебная аудитория для проведения занятий практического типа, учебная аудитория для проведения занятий лабораторного типа, помещение для курсового проектирования (выполнения курсовых работ), помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 430 ауд.

Описание имеющегося оборудования:

- Персональный компьютер Intel Core 2 Duo E6550 2.3 ГГц, ОЗУ – 2 Гб, жесткий диск – 250 Гб (12 шт.);
- Магнитно-маркерная доска;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

– Google Chrome
– LibreOffice
– Microsoft Visio 2010

Лаборатория «Информатика и программирование»

учебная аудитория для проведения занятий практического типа, учебная аудитория для проведения занятий лабораторного типа, помещение для курсового проектирования (выполнения курсовых работ), помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 428 ауд.

Описание имеющегося оборудования:

- Персональный компьютер Intel Core 2 Duo E6550 2.3 ГГц, ОЗУ – 2 Гб, жесткий диск – 250 Гб (14 шт.);
- Меловая доска;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

– Google Chrome
– LibreOffice
– Microsoft Visio 2010

Лаборатория «Муниципальная информатика»

учебная аудитория для проведения занятий практического типа, учебная аудитория для проведения занятий лабораторного типа, помещение для курсового проектирования (выполнения курсовых работ), помещение для самостоятельной работы

634034, Томская область, г. Томск, Вершинина улица, д. 74, 432б ауд.

Описание имеющегося оборудования:

Согласована на портале № 35157

- Персональный компьютер Intel Core i5-2320 3.0 ГГц, ОЗУ – 4 Гб, жесткий диск – 500 Гб (12 шт.);
- Магнитно-маркерная доска;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:
- Google Chrome
- LibreOffice
- Microsoft Visio 2010

Лаборатория «Распределенные вычислительные системы»
учебная аудитория для проведения занятий практического типа, учебная аудитория для проведения занятий лабораторного типа, помещение для курсового проектирования (выполнения курсовых работ), помещение для самостоятельной работы
634034, Томская область, г. Томск, Вершинина улица, д. 74, 432а ауд.
Описание имеющегося оборудования:
- Персональный компьютер Intel Core i5-3330 3.0 ГГц, ОЗУ – 4 Гб, жесткий диск – 500 Гб (12 шт.);
- Меловая доска;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:
- Google Chrome
- LibreOffice
- Microsoft Visio 2010

13.1.3. Материально-техническое и программное обеспечение для самостоятельной работы

Для самостоятельной работы используются учебные аудитории (компьютерные классы), расположенные по адресам:
- 634050, Томская область, г. Томск, Ленина проспект, д. 40, 233 ауд.;
- 634045, Томская область, г. Томск, ул. Красноармейская, д. 146, 201 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Состав оборудования:
- учебная мебель;
- компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 5 шт.;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения:
- Microsoft Windows;
- OpenOffice;
- Kaspersky Endpoint Security 10 для Windows;
- 7-Zip;
- Google Chrome.

13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины лицами с ограниченными возможностями здоровья и инвалидами осуществляется с использованием средств обучения общего и специального назначения.

При занятиях с обучающимися **с нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/переда-

Согласована на портале № 35157

чи учебной информации в доступных формах, мобильной системы преподавания для обучающихся с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой занимаются обучающиеся с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При занятиях с обучающимися **с нарушениями зрениями** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для комфортного просмотра.

При занятиях с обучающимися **с нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах, мобильной системы обучения для людей с инвалидностью.

14. Оценочные материалы и методические рекомендации по организации изучения дисциплины

14.1. Содержание оценочных материалов и методические рекомендации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы в составе:

14.1.1. Тестовые задания

ОПК-2: владением архитектурой электронных вычислительных машин и систем:

1) Какой принципы не относится к моделированию (проектированию) архитектуры электронных вычислительных машин и систем с использованием UML?

принцип абстрагирования

принцип многомодельности

принцип иерархического построения

принцип одномодельности

2) Какая модель при работе с архитектурой электронных вычислительных машин и систем не может входить в состав интегрированной модели сложной системы?

концептуальная

физическая

статистическая

динамическая

3) Что нельзя отнести к основным преимуществам N-уровневого архитектурного стиля при работе с архитектурой электронных вычислительных машин и систем?

удобство поддержки

отсутствие механизмов масштабируемости

доступность

гибкость

4) В контексте работы с архитектурным стилем, чем (как) характеризуется клиент/сервер (тип представления компонентов и коннекторов)?

дизайн приложения разлагается на функциональные или логические компоненты с возможностью повторного использования, предоставляющие тщательно проработанные интерфейсы связи

система разделяется на два приложения, где клиент выполняет запросы к серверу. Во многих случаях в роли сервера выступает база данных, а логика приложения представлена процедурами хранения

объектно-ориентированный архитектурный стиль, ориентированный на моделирование сферы деловой активности и определяющий бизнес-объекты на основании сущностей этой сферы

функциональные области приложения разделяются на многослойные группы (уровни)

5) В контексте работы с архитектурным стилем, чем (как) может характеризоваться компонентная архитектура (тип модульного представления)?

дизайн приложения разлагается на функциональные или логические компоненты с возможностью повторного использования, предоставляющие тщательно проработанные интерфейсы связи

система разделяется на два приложения, где клиент выполняет запросы к серверу. Во мно-

Согласована на портале № 35157

гих случаях в роли сервера выступает база данных, а логика приложения представлена процедурами хранения

объектно-ориентированный архитектурный стиль, ориентированный на моделирование сферы деловой активности и определяющий бизнес-объекты на основании сущностей этой сферы

функциональные области приложения разделяются на многослойные группы (уровни)

6) В контексте работы с архитектурным стилем, чем (как) может характеризоваться дизайн на основе предметной области (тип модульного представления)?

дизайн приложения разлагается на функциональные или логические компоненты с возможностью повторного использования, предоставляющие тщательно проработанные интерфейсы связи

система разделяется на два приложения, где клиент выполняет запросы к серверу. Во многих случаях в роли сервера выступает база данных, а логика приложения представлена процедурами хранения

объектно-ориентированный архитектурный стиль, ориентированный на моделирование сферы деловой активности и определяющий бизнес-объекты на основании сущностей этой сферы

функциональные области приложения разделяются на многослойные группы (уровни)

7) В контексте работы с архитектурным стилем, чем (как) может характеризоваться многослойная архитектура (тип модульного представления)?

дизайн приложения разлагается на функциональные или логические компоненты с возможностью повторного использования, предоставляющие тщательно проработанные интерфейсы связи

система разделяется на два приложения, где клиент выполняет запросы к серверу. Во многих случаях в роли сервера выступает база данных, а логика приложения представлена процедурами хранения

объектно-ориентированный архитектурный стиль, ориентированный на моделирование сферы деловой активности и определяющий бизнес-объекты на основании сущностей этой сферы

функциональные области приложения разделяются на многослойные группы (уровни)

8) В контексте работы с архитектурным стилем, чем (как) может характеризоваться шина сообщений (тип представления компонентов и коннекторов)?

архитектурный стиль, предписывающий использование программной системы, которая может принимать и отправлять сообщения по одному или более каналам связи, так что приложения получают возможность взаимодействовать, не располагая конкретными сведениями друг о друге

функциональность выделяется в отдельные сегменты, во многом аналогично многослойному стилю, но в данном случае сегменты физически располагаются на разных компьютерах

парадигма проектирования, основанная на распределении ответственности приложения или системы между отдельными многократно используемыми и самостоятельными объектами, содержащими данные и поведение

описывает приложения, предоставляющие и потребляющие функциональность в виде сервисов с помощью контрактов и сообщений

9) В контексте работы с архитектурным стилем, чем (как) может характеризоваться N-уровневая / 3-уровневая архитектура (тип представления компонентов и коннекторов)?

архитектурный стиль, предписывающий использование программной системы, которая может принимать и отправлять сообщения по одному или более каналам связи, так что приложения получают возможность взаимодействовать, не располагая конкретными сведениями друг о друге

функциональность выделяется в отдельные сегменты, во многом аналогично многослойному стилю, но в данном случае сегменты физически располагаются на разных компьютерах

парадигма проектирования, основанная на распределении ответственности приложения или системы между отдельными многократно используемыми и самостоятельными объектами, содержащими данные и поведение

описывает приложения, предоставляющие и потребляющие функциональность в виде сервисов с помощью контрактов и сообщений

10) В контексте работы с архитектурным стилем, чем (как) может характеризоваться объектно-ориентированная архитектура (тип модульного представления)?

архитектурный стиль, предписывающий использование программной системы, которая мо-

Согласована на портале № 35157

жет принимать и отправлять сообщения по одному или более каналам связи, так что приложения получают возможность взаимодействовать, не располагая конкретными сведениями друг о друге

функциональность выделяется в отдельные сегменты, во многом аналогично многослойному стилю, но в данном случае сегменты физически располагаются на разных компьютерах

парадигма проектирования, основанная на распределении ответственности приложения или системы между отдельными многократно используемыми и самостоятельными объектами, содержащими данные и поведение

описывает приложения, предоставляющие и потребляющие функциональность в виде сервисов с помощью контрактов и сообщений

11) В контексте работы с архитектурным стилем, чем (как) может характеризоваться сервисно-ориентированная

архитектура (SOA) (тип представления компонентов и коннекторов)?

архитектурный стиль, предписывающий использование программной системы, которая может принимать и отправлять сообщения по одному или более каналам связи, так что приложения получают возможность взаимодействовать, не располагая конкретными сведениями друг о друге

функциональность выделяется в отдельные сегменты, во многом аналогично многослойному стилю, но в данном случае сегменты физически располагаются на разных компьютерах

парадигма проектирования, основанная на распределении ответственности приложения или системы между отдельными многократно используемыми и самостоятельными объектами, содержащими данные и поведение

описывает приложения, предоставляющие и потребляющие функциональность в виде сервисов с помощью контрактов и сообщений

12) В архитектуре программного средства выделяют заинтересованные стороны. В чем выражается роль системного архитектора в рамках жизненного цикла программного обеспечения?

отвечает за разработку архитектуры и сопутствующей документации

отвечает за описание модели предметной области и бизнес-процессов

выступает заказчиком разработки, зачастую представляет конечного пользователя продукта

принимает участие в проектировании хранилища данных

13) В архитектуре программного средства выделяют заинтересованные стороны. В чем выражается роль системного бизнес-аналитика в рамках жизненного цикла программного обеспечения?

отвечает за разработку архитектуры и сопутствующей документации

отвечает за описание модели предметной области и бизнес-процессов

выступает заказчиком разработки, зачастую представляет конечного пользователя продукта

принимает участие в проектировании хранилища данных

14) В архитектуре программного средства выделяют заинтересованные стороны. В чем выражается роль (суть) потребителя программного продукта в рамках жизненного цикла программного обеспечения?

отвечает за разработку архитектуры и сопутствующей документации

отвечает за описание модели предметной области и бизнес-процессов

выступает заказчиком разработки, зачастую представляет конечного пользователя продукта

принимает участие в проектировании хранилища данных

15) В архитектуре программного средства выделяют заинтересованные стороны. В чем выражается роль (суть) администратора системы управления базами данных (СУБД) в рамках жизненного цикла программного обеспечения?

отвечает за разработку архитектуры и сопутствующей документации

отвечает за описание модели предметной области и бизнес-процессов

выступает заказчиком разработки, зачастую представляет конечного пользователя продукта

принимает участие в проектировании хранилища данных

16) В архитектуре программного средства выделяют заинтересованные стороны. Какая роль у специалиста по внедрению системы в рамках жизненного цикла программного обеспечения?

ответственный за установку готового программного продукта

ответственный за проектирование программного продукта на основе документации по архитектуре

Согласована на портале № 35157

ответственный за поддержку программного продукта в течение всего жизненного цикла

ответственный за поддержку аппаратного и программного обеспечения в компьютерной сети. Принимает участие во внедрении и настройке системы

17) В решении вопросов архитектуры электронных вычислительных машин и систем большое внимание уделяется формированию и анализу требований. Что подразумевают под собой требования?

спецификация того, что должно быть реализовано. В них описано поведение системы, свойства системы или ее атрибуты. Они могут служить ограничениями в процессе разработки системы

набор сервисов системы

набор ограничений к системе

характеристики системы

18) Основой процесса проектирования является определение требований к системе. Что из перечисленного не относится к основным группам требований, выделяемых в рамках процесса проектирования?

бизнес-требования

пользовательские требования

функциональные требования

требования к эргономике

19) Основой процесса проектирования является определение бизнес-правил (корпоративных политик). Что из перечисленного не относится к основным группам бизнес-правил?

факты

функции

ограничения

выводы

архитектура Фон Неймана

20) В каком типе архитектур сетевая нагрузка распределена между поставщиками услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами?

клиент-серверная архитектура

массивно-параллельная архитектура

симметричная архитектура

ОПК-3: готовностью применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов

1) Моделирование (проектирование) программного обеспечения имеет богатую историю во всех инженерных дисциплинах. Какой постулат не относится к его принципам?

выбор модели оказывает определяющее влияние на подход к реше¬нию проблемы и на то, как будет выглядеть это решение

каждая модель может быть воплощена с разной степенью абстракции

лучшие модели - те, что ближе к реальности

для полноценного видения системы возможно ограничиваться созданием только одной модели

2) Какое положение не относится к цели применения UML во время проектирования и конструирования программных продуктов?

моделировать системы целиком, от концепции до исполняемого артефакта, с помощью объектно-ориентированных методов

решить проблему масштабируемости, которая присуща сложным системам, предназначенным для выполнения ответственных задач

создать такой язык моделирования, который может использоваться не толь¬ко людьми, но и компьютерами

создать такой подход к моделированию, который позволит созданием одной модели подробно описать всю систему

3) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. К какому представлению системы она относится?

концептуальному представлению системы

логическому представлению системы

Согласована на портале № 35157

физическому представлению системы
представлению процесса функционирования

4) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. Что можно описать с помощью диаграммы прецедентов?

узлы системы
объекты системы
структуру системы
назначение системы

5) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. Какое количество отношений может быть установлено на диаграмме прецедентов?

1
2
3
4

6) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. Какие типы отношений нельзя построить на диаграмме прецедентов?

отношение ассоциации
отношение расширения
отношение включения
отношение дополнения

7) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. Какое отношение, направленное от прецедента А к прецеденту В, будет означать, что свойства экземпляра прецедента В могут быть дополнены, благодаря наличию свойств у расширяющего прецедента А?

расширения
включения
ассоциации
обобщения

8) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. Какие элементы нельзя отразить на диаграмме прецедентов?

актеры
прецеденты
классы
отношения между прецедентами

9) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. Какое отношение, направленное от прецедента А к прецеденту В, указывает, что каждый экземпляр прецедента А включает в себя функциональные свойства прецедента В?

ассоциации
обобщения
включения
расширения

10) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма прецедентов. Какое отношение, направленное от актера А к актеру В, призвано отразить тот факт, что каждый экземпляр актера А является одновременно экземпляром актера В и обладает всеми его свойствами?

ассоциации
обобщения
включения
расширения

11) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма деятельности. Какие элементы не могут присутствовать на диаграмме?

переходы

Согласована на портале № 35157

ветвления
деятельности
актеры

12) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма состояний. Какое свойство характеризует диаграмму состояний, представляющую собой автомат, у которого есть некоторое количество начальных/конечных состояний?

одно начальное, одно конечное состояние
одно начальное, несколько конечных состояний
несколько начальных, одно конечное состояние
несколько начальных, несколько конечных состояний

13) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма классов. Какое отношения между классами нельзя установить?

отношение зависимости
отношение расширения
отношение ассоциации
отношение реализации

14) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма классов. Какие элементы не могут фигурировать в диаграмме классов?

классы
пакеты
сервисы
отношения

15) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма классов. Что не может иметь (быть установлено) класс?

имя
атрибуты
методы
актеры

16) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма последовательности. Что не относится к её элементам?

линия жизни
фокус управления
объект
класс

17) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма последовательности. Какое количество разновидностей сообщений она содержит?

1
2
3
4

18) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма последовательности, что является её элементом?

узлы
компоненты
интерфейсы
классы

19) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма деятельности. К какому виду (типу) представления системы относится диаграмма деятельности?

концептуальному представлению системы
логическому представлению системы
физическому представлению системы
представлению процесса функционирования

Согласована на портале № 35157

20) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма компонентов. К какому виду (типу) представления системы относится диаграмма компонентов?

концептуальному представлению системы

логическому представлению системы

физическому представлению системы

представлению процесса функционирования

21) Одной из основных диаграмм, используемых в процессе проектирования систем, является диаграмма развертывания. К какому виду (типу) представления системы относится диаграмма развертывания?

концептуальному представлению системы

логическому представлению системы

физическому представлению системы

представлению процесса функционирования

14.1.2. Экзаменационные вопросы

1. Место, цели и задачи этапа проектирования в процессе разработки программных систем. Типы моделей и основные задачи, решаемые с их использованием. Основные методологии проектирования.

2. Процесс проектирования и принципы проектирования.

3. UML: принцип абстрагирования, принцип многомодельности, принцип иерархического построения моделей.

4. Место диаграммы прецедентов (use case diagram) в интегрированной модели системы. Цели создания диаграммы. Основные элементы и отношения между ними. Понятие прецедента (use case) и актера (actor). Абстрактные актеры и прецеденты. Семантика отношений: ассоциации (association relationship), возникающего между актером и прецедентом; обобщения (generalization relationship), определенного для актеров и пре-цедентов; включения (include relationship) и расширения (extend re-lationship) между прецедентами. Влияние установки границ системы на получаемую модель. Сравнение диаграмм прецедентов организационных и информационных систем. Примеры корректных и некорректных диаграмм. Рассмотрение типичных ошибок моделирования. Рекомендации по разработке диаграммы прецедентов.

5. Место диаграммы классов (class diagram) в интегрированной модели системы. Цели создания диаграммы. Понятие класса. Спецификация класса в языке UML (имя класса, имя пакета, секции). Абстрактные классы. Спецификация атрибутов класса: видимость, имя, кратность, тип, исходное значение. Спецификация операций класса: видимость, имя операции, параметры, возвращаемое значение, указание параллельности выполнения и абстрактности. Отношения на диаграмме классов: зависимости (dependency relationship); ассоциации (association relationship); обобщения (generalization relationship); реализации (realization relationship). Для каждого типа отношений рассматриваются: семантика, назначение, графическая нотация, стереотипы, кратность. Интерфейсы, объекты, параметризованные классы. Примеры построения диаграмм, анализ типичных ошибок, возникающих при построении диаграммы.

6. Место диаграммы состояний (statechart diagram) в интегрированной модели системы. Цели создания диаграммы. Понятие автомата. Формализм автомата: состояние и переход. Спецификация состояния. Простые и триггерные переходы. Сигнатура переходов. Подавтоматы. Последовательные и параллельные подавтоматы. Примеры построения диаграмм состояний.

7. Место диаграммы деятельности (activity diagram) в интегрированной модели системы. Цели создания диаграммы. Состояния деятельности и состояние действия. Переходы на диаграмме деятельности. Понятие ветвления, разделения (concurrent fork) и слияния (concurrent join). Использование диаграмм деятельности для моделирования бизнес-процессов. Использование дорожек. Взаимосвязь с диаграммой прецедентов. Примеры построения диаграмм.

8. Диаграмма последовательности, назначение, основные элементы. Объекты, сообщения. Временные ограничения. Взаимосвязь с диаграммой последовательности и диаграммой классов. Примеры построения диаграммы.

9. Диаграмма кооперaций. Назначение, основные элементы. Понятие кооперации. Кооперация уровня спецификаций и уровня примеров. Объекты, активные объекты, мультиобъекты, со-

Согласована на портале № 35157

ставные объекты на диаграмме кооперации. Связи на диаграмме кооперациий: семантика, стереотипы. Сообщения: назначение, формат записи, стереотипы. Примеры построения диаграмм.

10. Диаграмма компонентов, назначение, основные элементы. Понятие компонента. Связи на диаграмме компонентов. Примеры построения диаграмм компонентов.

11. Диаграмма развертывания, назначение, основные элементы. Примеры построения диаграмм.

14.1.3. Темы контрольных работ

1. Принципы проектирования программных систем. Принцип иден-тичности.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

2. Принципы проектирования программных систем. Принцип тех-нологичности.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

3. Принципы проектирования программных систем. Принцип не-прерывности, поэтапности, преемственности разработки и развития.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

4. Принципы проектирования программных систем. Принцип адап-тивноси.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

5. Принципы проектирования программных систем. Модульный принцип построения программных и технических средств.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

6. Принципы проектирования программных систем. Технологиче-ская интеграция.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

7. Принципы проектирования программных систем. Полная норма-лизация процессов и их мониторинг.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

8. Принципы проектирования программных систем. Регламентация.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

9. Принципы проектирования программных систем. Экономиче-ская целесообразность.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

10. Принципы проектирования программных систем. Типизация или максимальное использование готовых решений и средств.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

11. Принципы проектирования программных систем. Стандартиза-ция проектных решений.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

12. Принципы проектирования программных систем. Принцип кор-поративности.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

13. Принципы проектирования программных систем. Ориентация на первых лиц объекта автоматизации.

В рамках выполнения контрольного задания необходимо описать принцип. Привести пример.

14. Методологии проектирования программных систем. Scrum.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения методологии. Привести примеры.

Согласована на портале № 35157

15. Методологии проектирования программных систем. Kanban.

В рамках выполнения контрольного задания необходимо описать достоинства и недостатки применения методологии. Привести примеры.

16. Методологии проектирования программных систем. Dynamic system development method.

В рамках выполнения контрольного задания необходимо описать достоинства и недостатки применения методологии. Привести примеры.

17. Методологии проектирования программных систем. Microsoft solutions framework.

В рамках выполнения контрольного задания необходимо описать достоинства и недостатки применения методологии. Привести примеры.

18. Методологии проектирования программных систем. Rational unified process.

В рамках выполнения контрольного задания необходимо описать достоинства и недостатки применения методологии. Привести примеры.

19. Архитектурные стили и модели. Клиент-серверная архитектура.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

20. Архитектурные стили и модели. Компонентная архитектура.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

21. Архитектурные стили и модели. Проблемно-ориентированное проектирование архитектура.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

22. Архитектурные стили и модели. Многослойная архитектура.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

23. Архитектурные стили и модели. Архитектура на основе канала сообщений.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

24. Архитектурные стили и модели. Трехуровневая архитектура.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

25. Архитектурные стили и модели. Объектно-ориентированная ар-хитектура.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

26. Архитектурные стили и модели. Сервисно-ориентированная ар-хитектура.

В рамках выполнения контрольного задания необходимо описать досто-инства и недостатки применения архитектуры. Привести примеры.

14.1.4. Темы индивидуальных заданий

Разработка технического проекта программной системы

14.1.5. Темы лабораторных работ

Создание диаграммы прецедентов

Создание диаграммы классов

Создание диаграммы состояний и диаграммы деятельности системы

Создание диаграмм последовательности и кооперацией

Создание диаграмм компонентов и развертывания

14.2. Требования к оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусмотрены дополнительные оценочные материалы, перечень которых указан в таблице 14.

Таблица 14 – Дополнительные материалы оценивания для лиц с ограниченными возможностями здоровья и инвалидов

Категории	Виды дополнительных оценочных	Формы контроля и оценки

Согласована на портале № 35157

обучающихся	материалов	результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами исходя из состояния обучающегося на момент проверки

14.3. Методические рекомендации по оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной форме;
- в печатной форме с увеличенным шрифтом;
- в форме электронного документа;
- методом чтения ассистентом задания вслух;
- предоставление задания с использованием сурдоперевода.

Лицам с ограниченными возможностями здоровья и инвалидам увеличивается время на подготовку ответов на контрольные вопросы. Для таких обучающихся предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге;
- набор ответов на компьютере;
- набор ответов с использованием услуг ассистента;
- представление ответов устно.

Процедура оценивания результатов обучения лиц с ограниченными возможностями здоровья и инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

При необходимости для лиц с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 35157