

Федеральное государственное образовательное бюджетное учреждение высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)

Кафедра информационных технологий
Факультет информационных технологий и анализа больших данных

Документ подписан усиленной неквалифицированной электронной подписью.

Организация: «ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»

Сертификат: O7ZkDPhSHKR4aEe3iA+lqOWVzexUeuNh

Утверждено: Проректор по учебной и методической работе, Е.А. Каменева

Дата: 06.10.2025 г.

Е.С. Будаев
Методы промышленной разработки информационных систем

Рабочая программа дисциплины
для студентов, обучающихся по направлению подготовки
09.03.04 - Программная инженерия,
Образовательная программа «Разработка и внедрение информационно -
аналитических систем»
Профиль:
«Разработка и внедрение информационно - аналитических систем»

Рекомендовано
Факультет информационных технологий и анализа больших данных
(протокол № 2 от 18.11.2025 г.)

Одобрено
Кафедра информационных технологий
(протокол № 11 от 12.11.2025 г.)



Содержание

1. Наименование дисциплины	3
2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине	3
3. Место дисциплины в структуре образовательной программы	5
4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	5
5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий	6
5.1. Содержание дисциплины	6
5.2. Учебно – тематический план	9
5.3. Содержание семинаров, практических занятий	11
6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	14
6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы	14
6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю	16
7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	19
8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	5
9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	7
10. Методические указания для обучающихся по освоению дисциплины	31
11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)	33
12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	33



1. Наименование дисциплины

Методы промышленной разработки информационных систем

2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
ПКП-1	Способность выполнять разработку информационных систем с применением современных инструментальных средств	Демонстрирует знания основных понятий информационных систем	Знать: понятия информационных систем, их назначение, классификацию и состав Уметь: различать разные типы информационных систем, формировать документы в соответствии с отраслевыми стандартами
		Понимает достоинства и недостатки различных современных инструментальных средств для разработки ИС, может критически анализировать существующие решения	Знать: ключевые характеристики, принципы работы, области применения, сильные и слабые стороны современных инструментальных средств разработки информационных систем Уметь: проводить сравнительный анализ существующих инструментальных средств, оценивая их применимость для решения конкретных задач и аргументируя выбор или критику их недостатков
		Описывает процесс разработки информационных систем	Знать: основные этапы, методологии, модели и их характеристики в процессе разработки информационных систем Уметь: структурированно и последовательно описывать



Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
			процесс разработки ИС, выделяя ключевые фазы, участников и результаты каждой стадии
		Выполняет программную реализацию информационных систем с учетом требований к применяемым технологиям	Знать: требования, предъявляемые к современным технологиям разработки (таким как фреймворки, языки программирования, СУБД), и ограничения, накладываемые этими технологиями Уметь: применять выбранные технологии на практике для создания работоспособных компонентов информационной системы в полном соответствии с техническим заданием и заданными требованиями



3. Место дисциплины в структуре образовательной программы

Дисциплина «Методы промышленной разработки информационных систем» относится к «Циклу профиля».

4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

Очная форма обучения

Вид учебной работы по дисциплине	Всего (в з/е и часах)	Семестр 5 (в часах)	Семестр 6 (в часах)
Общая трудоемкость дисциплины	8/288	126	162
Контактная работа – Аудиторные занятия	118	68	50
<i>Лекции</i>	<i>50</i>	<i>34</i>	<i>16</i>
<i>Семинары, практические занятия</i>	<i>68</i>	<i>34</i>	<i>34</i>
<i>Самостоятельная работа</i>	<i>170</i>	<i>58</i>	<i>112</i>
<i>В том числе курсовая работа/ курсовой проект</i>	<i>24</i>	<i>-</i>	<i>-</i>
Вид текущего контроля	Контрольная работа	Контрольная работа	
Вид промежуточной аттестации	Зачет ; Экзамен	Зачет	Экзамен



5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Управление разработкой ПО. Гибкие методологии (Agile)

Введение в стандарт SWEBOOK. Идеи Agile. Принципы Agile. Основные роли в Скрам. События Скрам. Формирование требований в Agile. Основные практики Канбан. Роли в Канбан.

Тема 2. Управление разработкой ПО. Обзор методологий разработки крупных информационных систем

Введение в разработку крупных информационных систем. Unified Process — методология разработки ИС/ПО. Rational Unified Process (RUP) и Oracle Unified Method как разновидности унифицированного процесса. Унифицированная модель организации внедрения решений в методологии Microsoft Solutions Framework (MSF).

Тема 3. Формирование требований при разработке ПО

Требования к качеству ПО. Контекст системы. Категории требований. Методы сбора требований. Карта пользовательских историй. Типы документации. Критерии качества для требований. Способы моделирования требований. Типы конфликтов. Приоритизация требований. Управление изменениями требований.

Тема 4. Управление разработкой ПО. Проектное управление

Что такое проект. Методология PMBOK. Группы процессов. Области знаний. Ограничения проекта. Жизненный цикл проекта. Устав проекта. Содержание проекта. План управления проектом. Риски проекта. Обзор методологии PRINCE2. Принципы. Ограничения. Процессы управления. Компоненты методологии.

Тема 5. Масштабирование управления разработкой ПО

Масштабирование проектного управления по PMBOK. Портфель и программы проектов. Типы проектных офисов. Различные организационные структуры. Масштабирование гибких методологий на примере Scaled Agile Framework. События. Артефакты. Роли. Критичные факторы успеха.

Тема 6. Тестирование ПО



Цели тестирования. Принципы тестирования. Контекст тестирования. Уровни тестирования. Типы тестирования. Тестирование изменений. Статическое тестирование. Категории методов проектирования тестов. Стратегия тестирования. Дефекты. Инструменты тестирования. Автоматизация тестирования.

Тема 7. Сопровождение ПО

ITIL. Модель управления услугами. Цепочка создания ценности. Принципы. Управление инцидентами. Управление проблемами. Управление изменениями. SLA. Управление конфигурациями. Управление релизами.

Тема 8. Методология DevOps и ее расширения

Цели. Что такое DevOps. Сочетание с Agile. Культура взаимодействия. Базовые инструменты. Непрерывная интеграция. Запросы на слияние. Сборка. Модульное тестирование. TDD. Статический анализ кода. Среды разработки и тестирования. Динамические среды. Непрерывная поставка. Непрерывное развертывание. Управление репозиторием. Управление конфигурациями. Тесты развертывания. DevSecOps. Основные точки встраивания. Непрерывные измерения. Процесс измерений. Метрики. Контейнеризация. Оркестрация контейнеров. Паттерны и антипаттерны образования команд.

Тема 9. Основные идеи современной программной инженерии

Рассматриваются современные идеи программной инженерии, применение элементов научного подхода, идея о том, что две основные проблемы, которые пытается решить программная инженерия, это проблема познания в эволюционном процессе создания программных систем и проблема управления сложностью. Рассматриваются самые базовые подходы применимые с вариациями на различных стадиях инженерного процесса (анализ, проектирование, конструирование, поставка, эксплуатация).

Тема 10. Управление качеством производства ПО

Основные аспекты управления качеством: качество результата, качество системы, качество процессов. Основные стандарты, регламентирующие качество ПО, модели качества ПО, процессы оценки качества, модели зрелости организаций.

Тема 11. Бизнес-архитектура и техническая архитектура

Бизнес-архитектура, как основа для программной архитектуры решения. Переход от требований к драйверам и архитектурно значимым артефактам. Создание



программной архитектуры на основе бизнес-архитектуры. Архитектурные стили и шаблоны.

Тема 12. Современные варианты распределенных приложений

Более подробно рассматриваются (относительно) новые варианты построения приложений - микросервисная архитектура и дается представление о подходе Cloud Native Apps (в трендах Гартнер рассматривается как топ-3 технология на 2023-2024 годы).

Тема 13. Атрибуты качества и сценарии атрибутов качества

Рассматривается понятие атрибутов качества, атрибутов, наличие которых в предполагаемой (разрабатываемой) системе должна обеспечить создаваемая архитектура системы. Сценарии атрибутов качества как способы подтверждения свойств системы. Практические вопросы применения атрибутов качества. Тактики достижения показателей качества.

Тема 14. Паттерны проектирования

Паттерны проектирования, как подход к повторному использованию удачных решений. Принципы применения паттернов. Классификация паттернов. Каталог паттернов.

Тема 15. Мониторинг

Мониторинг, как средство обеспечения качества функционирующей системы. Типы мониторинга, основные подходы и инструменты реализации.

Тема 16. Рефакторинг. Метрики программного кода

Рефакторинг, как подход к улучшению существующего кода. Принципы рефакторинга. Каталог методов рефакторинга. Типовые ошибки. Способы и методы оценки параметров исходного кода. Основные группы метрик, применимость их в различных сценариях.



5.2. Учебно – тематический план

Очная форма обучения

п/ п	Наименование тем (разделов) дисциплины	Трудоемкость в часах				
		Всего	Контактная работа - Аудиторная работа			Самостоятельная работа
			Общая, в т.ч.:	Лекции	Семинары, практические занятия	
1	Управление разработкой ПО. Гибкие методологии (Agile)	16	6	2	4	10
2	Управление разработкой ПО. Обзор методологий разработки крупных информационных систем	16	6	2	4	10
3	Формирование требований при разработке ПО	16	6	2	4	10
4	Управление разработкой ПО. Проектное управление	16	6	2	4	10
5	Масштабирование управления разработкой ПО	16	6	2	4	10
6	Тестирование ПО	20	8	2	6	12
7	Сопровождение ПО	14	6	2	4	8
8	Методология DevOps и ее расширения	16	8	4	4	8
9	Основные идеи современной программной	22	8	4	4	14



п/ п	Наименование тем (разделов) дисциплины	Трудоемкость в часах				
		Всего	Контактная работа - Аудиторная работа			Самостоятельная работа
			Общая, в т.ч.:	Лекции	Семинары, практические занятия	
	инженерии					
10	Управление качеством производства ПО	20	8	4	4	12
11	Бизнес-архитектура и техническая архитектура	18	8	4	4	10
12	Современные варианты распределенных приложений	20	8	4	4	12
13	Атрибуты качества и сценарии атрибутов качества	22	8	4	4	14
14	Паттерны проектирования	22	10	4	6	12
15	Мониторинг	18	8	4	4	10
16	Рефакторинг. Метрики программного кода	16	8	4	4	8
В целом по дисциплине		288	118	50	68	170



5.3. Содержание семинаров, практических занятий

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Управление разработкой ПО. Гибкие методологии (Agile)	Формирование команд (принципы, методика). Инструменты коллаборации в командах. Практики взаимодействия.	Устный опрос, проверка практических заданий.
Управление разработкой ПО. Обзор методологий разработки крупных информационных систем	Методологии разработки крупных информационных систем. Плюсы и минусы данных методологий.	Устный опрос, проверка практических заданий.
Формирование требований при разработке ПО	Сбор требований различными способами. Формирование итоговых требований.	Устный опрос, проверка практических заданий.
Управление разработкой ПО. Проектное управление	Типы проектных офисов. Основы организации работ.	Устный опрос, проверка практических заданий.
Масштабирование управления разработкой ПО	Походы к масштабированию – портфели проектов (проектный подход) и SAFE (гибкий подход).	Устный опрос, проверка практических заданий.
Тестирование ПО	Формирование сценариев ручного тестирования на основе User Stories. Формирование списка дефектов по результатам тестирования.	Устный опрос, проверка практических заданий.



Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Сопровождение ПО	Составление SLA.	Устный опрос, проверка практических заданий.
Методология DevOps и ее расширения	Настройка конвейеров поставки (ci/cd pipelines). Используемые инструменты, типовые стадии для построения конвейера.	Устный опрос, проверка практических заданий.
Основные идеи современной программной инженерии	Отличительные особенности программной инженерии от производственной. Современные подходы к управлению сложностью и к работе с меняющимися требованиями.	Устный опрос, проверка практических заданий.
Управление качеством производства ПО	Работа с регламентными документами, проработка стратегии обеспечения качества программного продукта.	Устный опрос, проверка практических заданий.
Бизнес-архитектура и техническая архитектура	Работа с требованиями для выработки проектных решений. Выявление и формализация нефункциональных требований, определяющих качество системы. Планирование взаимодействия с заинтересованными лицами на стадии реализации проекта.	Устный опрос, проверка практических заданий.



Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Современные варианты распределенных приложений	Контейнеризация приложений как универсальный подход для развертывания приложений. Docker и Kubernetes как примеры промышленной реализации.	Устный опрос, проверка практических заданий.
Атрибуты качества и сценарии атрибутов качества	Атрибуты качества как средство формализации нефункциональных требований. Принципы формирования, стратегии обеспечения.	Устный опрос, проверка практических заданий.
Паттерны проектирования	Паттерны как решение для стандартных проблем разработки ПО. Практика применения.	Устный опрос, проверка практических заданий.
Мониторинг	Инструменты мониторинга. Обеспечение качества на стадии эксплуатации программного комплекса.	Устный опрос, проверка практических заданий.
Рефакторинг. Метрики программного кода	Рефакторинг приложений как инструмент повышения качества. Инструменты и принципы. Работа с метриками для обеспечения стабильного качества проекта.	Устный опрос, проверка практических заданий.



6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Управление разработкой ПО. Гибкие методологии (Agile)	Различные методики гибкой разработки – XP, Kanban, Lean, Scrum, MSF for Agile Software Development.	Изучение материалов (статьи, стандарты). Изучение литературы.
Управление разработкой ПО. Обзор методологий разработки крупных информационных систем	Методологии разработки крупных ИС: ARIS (Architecture of Integrated Information Systems), ASAP (Accelerated SAP) компании SAP AG, MDSS (Microsoft Dynamics Sure Steps) от Microsoft и AIM (Application Implementation Method) как часть концепции Oracle Unified Method.	Изучение материалов (статьи). Изучение литературы.
Формирование требований при разработке ПО	Изучение нормативных документов (ГОСТы).	Изучение документации.
Управление разработкой ПО. Проектное управление	Изучение нормативных документов по проектному управлению (PMBOOK).	Изучение литературы.
Масштабирование управления разработкой ПО	Подходы к масштабированию при проектном и гибком подходах.	Изучение литературы.
Тестирование ПО	Подход TDD (Test Driven Development).	Изучение литературы.
Сопровождение ПО	Передача в поддержку, работа с изменениями и реорганизация команды проекта.	Изучение литературы.
Методология DevOps и ее расширения	Изучение практик DevOps.	Изучение литературы.



Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Основные идеи современной программной инженерии	Современные подходы к управлению сложностью и неопределенности требований.	Изучение литературы.
Управление качеством производства ПО	Модели зрелости – CMM и CMMI.	Изучение литературы.
Бизнес-архитектура и техническая архитектура	Основные элементы бизнес-архитектуры применительно к программным продуктам.	Изучение литературы.
Современные варианты распределенных приложений	Современные подходы к организации распределенных приложений.	Изучение материалов (статьи).
Атрибуты качества и сценарии атрибутов качества	Стандартные подходы к формированию сценариев атрибутов качества. Стратегии обеспечения качества.	Изучение литературы.
Паттерны проектирования	Работа с паттернами проектирования.	Изучение литературы.
Мониторинг	Open-source инструменты мониторинга Zabbix, Elasticsearch, Prometheus.	Изучение литературы.
Рефакторинг. Метрики программного кода	Рефакторинг программного кода. Внедрение инструментов сбора метрик в конвейеры сборки и поставки программного кода.	Изучение материалов (статьи).



6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

Примерные задания для контрольной работы

1. Написать 15 требований к веб-сайту заказа билетов в кино. Разделить на функциональные и качественные, дополнительно разделить в соответствии с моделью Кано. Требования должны удовлетворять критериям качества.
2. Составить ER-диаграмму по веб-сайту заказа билетов. Не менее 6 сущностей, общий состав атрибутов не менее 40, связи много ко много не допускаются, минимум один пример связи один к одному.
3. Написать 15 пользовательских историй к веб-сайту заказа билетов в кино с указанием критериев приемки и разделением минимум на две роли.
4. Написать 6 сценариев тестирования по веб-сайту заказа билетов с общим количеством шагов не менее 40. Обязательно должны быть использованы методы анализа граничных значений и таблицы альтернатив (сложно).
5. Написать базовые положения соглашения об уровне обслуживания веб-сайта заказа билетов со стороны внешнего подрядчика. Выделить три уровня приоритета, критерии определения приоритета и временные характеристики.

Критерии балльной оценки различных форм текущего контроля успеваемости содержатся в соответствующих методических рекомендациях кафедры информационных технологий Факультета информационных технологий и анализа больших данных.

Дополнительная информация

Примеры тестовых заданий

1. Выберите из нижеперечисленных требований те, которые не удовлетворяют критериям качества требований:

- a. Хранение данных о моделях автомобилей, их комплектации и ценах
- b. Возможность сравнения автомобилей между собой по заданным параметрам
- c. Уникальный дизайн приложения
- d. Минимальное использование приложением ресурсов телефона.
- e. Возможность авторизации в приложении по логину/паролю

2. Выберите из нижеперечисленных требований те, которые не удовлетворяют критериям качества требований:

- a. Наличие рецептов различных коктейлей в приложении
- b. Отсутствие рекламы в платной версии приложения
- c. Дружелюбный дизайн приложения
- d. Обеспечение надежности хранения информации
- e. Возможность авторизации в приложении по логину/паролю



3. Вам необходимо собрать требования по автоматизации операционного участка предприятия, выполняющего рутинную работу. Документация отсутствует, а работники не горят желанием рассказывать про свою работу, так как понимают, что автоматизация приведет к сокращениям. Какой способ сбора требований подходит больше всего:

- a. Интервью
- b. Анкетирование
- c. Мозговой штурм
- d. Работа в «поле»
- e. Диаграммы использования

4. Ваша компания делает стандартные сайты-визитки для небольших компаний. Какой способ сбора требований подходит больше всего:

- a. Интервью
- b. Анкетирование
- c. Мозговой штурм
- d. Работа в «поле»
- e. Диаграммы использования

5. Вы собираетесь в команде делать пилот нового приложения, аналоги которого практически отсутствуют. Выберите наиболее подходящие способы сбора требований для вашего приложения:

- a. Интервью
- b. Анкетирование
- c. Мозговой штурм
- d. Работа в «поле»
- e. Диаграммы использования

6. Вы собираетесь переделывать текущее приложение и с вами готовы работать продвинутые пользователи, объясняя текущую работу приложения и все особенности его работы. Выберите наиболее подходящий способ сбора требований для нового приложения:

- a. Интервью
- b. Анкетирование
- c. Мозговой штурм
- d. Работа в «поле»
- e. Диаграммы использования

7. Вам дано задание составить тестовые сценарии с обязательной проверкой граничных значений. Калькулятор рассчитывает тариф отдельно для людей старше 60 лет. Какие значения вы будете проверять для проверки границ:

- a. 60, 61, 62



- b. 59, 60
- c. 58, 59, 60, 61, 62
- d. 58, 59, 60
- e. 59, 60, 61

8. Вам дано задание внедрить статический анализатор кода. В какое место pipeline вы поставите этот анализатор:

- a. После автотестов UI
- b. После компиляции приложения
- c. Перед компиляцией приложения
- d. Перед выкладкой в промышленную среду
- e. Автоматическая проверка при написании кода

9. Вам дано задание внедрить динамический анализатор кода. В какое место pipeline вы поставите этот анализатор:

- a. После автотестов UI
- b. После компиляции приложения
- c. Перед компиляцией приложения
- d. Перед выкладкой в промышленную среду
- e. Автоматическая проверка при написании кода

10. Ваша команда собирается перейти на полностью дистанционную работу, выберите типы приложений, которые обязательно понадобятся вашей команде:

- a. Мессенджер
- b. Видео-связь
- c. Задачник
- d. Мониторинг работы
- e. База знаний



7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Перечень компетенций с указанием индикаторов их достижения в процессе освоения образовательной программы содержится в разделе 2. «Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».

Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний

Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
ПКП-1. Способность выполнять разработку информационных систем с применением современных инструментальных средств	Демонстрирует знания основных понятий информационных систем	Знать: понятия информационных систем, их назначение, классификацию и состав Уметь: различать разные типы информационных систем, формировать документы в соответствии с отраслевыми стандартами	Создать описание информационной системы предприятия (выбрать тип, архитектуру, содержание и описание подсистем, модулей).



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
	Понимает достоинства и недостатки различных современных инструментальных средств для разработки ИС, может критически анализировать существующие решения	Знать: ключевые характеристики, принципы работы, области применения, сильные и слабые стороны современных инструментальных средств разработки информационных систем Уметь: проводить сравнительный анализ существующих инструментальных средств, оценивая их применимость для решения конкретных задач и аргументируя выбор или критику их недостатков	Сделать обзор, выбор, определить критерии сравнения, провести сравнение и анализ информационных систем по конкретной предметной области для заданного предприятия.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
	Описывает процесс разработки информационных систем	Знать: основные этапы, методологии, модели и их характеристики в процессе разработки информационных систем Уметь: структурированно и последовательно описывать процесс разработки ИС, выделяя ключевые фазы, участников и результаты каждой стадии	Для разработки информационной системы выбрать подходящую/подходящие методологии, модели, определить основных стейкхолдеров, распределить примерный объем работы исполнителей работ, описать состав и структуру необходимых артефактов после каждой стадии разработки ИС.
	Выполняет программную реализацию информационных систем с учетом требований к применяемым технологиям	Знать: требования, предъявляемые к современным технологиям разработки (таким как фреймворки, языки программирования, СУБД), и огра-	На основе технического задания представить программную реализацию подсистемы/системы с использованием современных технологий разработки, с последующим тестированием и документированием.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		ничения, накладываемые этими технологиями Уметь: применять выбранные технологии на практике для создания работоспособных компонентов информационной системы в полном соответствии с техническим заданием и заданными требованиями	



Примеры практико-ориентированных заданий

1. Написать 15 требований к (ИС в конкретной предметной области). Разделить на функциональные и качественные, дополнительно разделить в соответствии с моделью Кано. Требования должны удовлетворять критериям качества.
2. Составить ER-диаграмму по (БД для ИС в конкретной предметной области). Не менее 6 сущностей, общий состав атрибутов не менее 40, связи много ко много не допускаются, минимум один пример связи один к одному.
3. Написать 15 пользовательских историй к (ИС в конкретной предметной области) с указанием критериев приемки и разделением минимум на две роли.
4. Написать 6 сценариев тестирования к (ИС в конкретной предметной области) с общим количеством шагов не менее 40. Обязательно должны быть использованы методы анализа граничных значений и таблицы альтернатив.
5. Написать базовые положения соглашения об уровне обслуживания (ИС в конкретной предметной области) со стороны внешнего подрядчика. Выделить три уровня приоритета, критерии определения приоритета и временные характеристики.

Примеры вопросов для подготовки к промежуточной аттестации

Примерные вопросы для подготовки к зачету

1. Вам необходимо собрать требования по автоматизации операционного участка предприятия, выполняющего рутинную работу. Документация отсутствует, а работники не горят желанием рассказывать про свою работу, так как понимают, что автоматизация приведет к сокращениям. Какой способ сбора требований подходит больше всего.
2. Ваша компания делает стандартные сайты-визитки для небольших компаний. Какой способ сбора требований подходит больше всего.
3. Вы собираетесь в команде делать пилот нового приложения, аналоги которого практически отсутствуют. Выберите наиболее подходящие способы сбора требований для вашего приложения.
4. Вы собираетесь переделывать текущее приложение на новый технологический стек и с вами готовы работать продвинутые пользователи, объясняя текущую работу приложения и все особенности его работы. Выберите наиболее подходящий способ сбора требований для нового приложения.
5. Вам необходимо внедрить новую систему мониторинга приложений. Выберите фреймворк управления.
6. Вы собираетесь разработать новое мобильное приложение, состоящее внутри из различных модулей. Выберите фреймворк управления разработкой.



7. Вам поручено управление по разворачиванию новой инфраструктуры по виртуализации. Выберите фреймворк по управлению.
8. Вам поручено управлению по внедрению целого ряда процессов ITIL – управление инцидентами, управлением проблемами, управление запросами на обслуживание, управление запросами на изменение и управление конфигурациями. Выберите фреймворк по управлению.
9. Вам поручено управление разработкой набора микросервисов. Выберите фреймворк по управлению разработкой.
10. Вам поставили задачу внедрить в разработке софта DevOps pipeline. С каких базовых типов инструментов вы начнете?
11. Вы внедряете DevOps pipeline в разработке софта. К вам пришла информационная безопасность и потребовала предоставить им возможности контроля, чтобы сделать разработку безопасной. Какие варианты вы предложите информационной безопасности?
12. Вы внедряете DevOps pipeline в разработке софта. К вам пришли разработчики и тестировщиками с тем, что они тратят кучу времени для поиска причин дефектов в работе внешнего сайта из-за меняющихся и различных настроек операционной системы и веб-сервера на тестовых виртуалках. Что вы им предложите?
13. Назовите основные практики DevOps и чем они концептуально друг от друга отличаются.
14. Система состоит из 5 сильно взаимосвязанных модулей, по запросу заказчика были внесены изменения в один из модулей. Какие типы тестирования обязательно применить перед выкладкой изменения на промышленную среду.
15. Безопасность постоянно ругается с разработчиками из-за регулярного использования в коде паролей в открытом виде. Какой вид тестирования может помочь безопасности избавиться от такого грубейшего нарушения в принципе?
16. К вам предъявили требования организовать тестирование на как можно более ранних этапах жизненного цикла разработки. Какой способ тестирования обеспечивает его на самом раннем этапе?
17. Ваша компания занимается разработкой стандартных сайтов-визиток по заданным шаблонам. Вам поставили задачу организовать процесс тестирования. Какой тип тестирования лучше всего подойдет с учетом требований по высокой скорости обратной связи?

Примерные вопросы для подготовки к экзамену



1. Жизненный цикл ПО. Понятие жизненного цикла ПО, этапы, их описание в терминологии различных стандартов (примеры).
2. Процессы создания ПО. Управление процессами, 5 групп процессов, их характерные особенности.
3. Проект. Управление проектами. Понятие проектного офиса, типы проектных офисов. Программы и портфели.
4. Требования к ПО. Виды требований. Техники сбора и описания.
5. Проектное управление. Ограничения и жизненный цикл проекта
6. Проектное управление. Методологии. PMBOOK и PRINCE2
7. Гибкие принципы управление. Agile – идеи и принципы.
8. Гибкие технологии. Scrum
9. Гибкие технологии. Kanban
10. Масштабирование гибкой разработки. Проблемы и решения. SAFe как пример масштабируемого фреймворка
11. Гибкое управление. Особенности организации работы дистанционно работающих команд
12. Поток поставки ценности. DevOps – понятия и основные принципы
13. DevOps. Компоненты и процессы. Понятие безопасной разработки
14. Документация. Типы, назначение. Принципы создания, группы заинтересованных лиц, методики формирования пакетов документации.
15. Тестирование ПО. Цели и принципы
16. Тестирование. Уровни и типы тестирования.
17. Тестирование. Методы черного и белого ящика. Статическое и динамическое тестирование
18. Тестирование. Методы и инструменты. Классификация дефектов
19. Сопровождение и поддержка ПО. ITSM и ITIL общие понятия и принципы
20. Сопровождение и поддержка. ITIL. Базовые процессы
21. Управление изменениями. ITIL и процесс управления изменениями
22. Процесс разработки. Архитектура решения. Назначение, принципы выбора. Анализ и оценка
23. Процесс разработки. Архитектура. Документирование и распространение сведений. Группы заинтересованных лиц.
24. Атрибуты качества систем и их связь с функциональными требованиями. Сценарии атрибутов качества
25. Атрибуты качества и сценарии - подробное описание на примере одного из типа сценариев (Модифицируемость, Ремонтопригодность, Производительность, Безопасность...)



26. Тактики обеспечения атрибутов качества (на примере одного из типов Модифицируемость, Ремонтопригодность, Производительность, Безопасность...)
27. Документирование программных систем. Заинтересованные лица, представления и нотации (с примерами).
28. Документирование программных систем. Неформальные нотации. Плюсы и минусы, области применения
29. Документирование программных систем. Частично формальные и формальные нотации. Плюсы и минусы, области применения
30. Нотации. Рассмотреть пример одной из нотаций на выбор. Основные типы диаграмм, области применения
31. Стили и шаблоны проектирования. Понятие архитектурного стиля и архитектурного шаблона
32. Архитектурные принципы проектирования. Основные принципы и подробно на примере 1 – 2 выбранных
33. Архитектурные стили. Уровни применения, типичные представители (на примере выбранного - монолит, SOA, Микросервисы...)
34. Архитектурные стили на уровне компонентов. Слоистая и модульная архитектура
35. Паттерны проектирования. Общие понятия, назначение, плюсы и минусы.
36. Паттерны проектирования. Порождающие паттерны (на выбранном примере)
37. Паттерны проектирования. Структурные паттерны (на выбранном примере)
38. Паттерны проектирования. Поведенческие паттерны (на выбранном примере)
39. Управление качеством. Основные понятия, процессы и практики
40. Управление качеством. Стандарты и модели
41. Управление качеством. Эталонная модель
42. Методологии управления качеством. Классификация
43. Рефакторинг. Назначение и принципы
44. Рефакторинг. Техники рефакторинга (на примере одной из техник)
45. Метрики программного кода. Основные принципы и подходы, инструменты, практика применения.

Пример экзаменационного билета

Экзаменационный билет №

1 вопрос **(30 баллов)**: Написать 15 требований к веб-сайту заказа билетов в кино. Разделить на функциональные и качественные, дополнительно разделить согласно модели Кано. Требования должны удовлетворять критериям качества.



2 вопрос **(15 баллов)**: Опишите отличия между проектным управлением pmbook и работой по agile (scrum)

3 вопрос **(15 баллов)**: Паттерны проектирования. Порождающие паттерны (на выбранном примере). Описать компоненты паттерна, их взаимодействие, привести собственный пример. Желательно проиллюстрировать схемой вызова.

Примерные темы курсовой работы /курсового проекта

1. Описание проекта разработки программного продукта в соответствии с ГОСТ 34.
2. Разработка стратегии тестирования программного продукта.
3. Разработка пакета документации для передачи в сопровождение.
4. Составление набора проектной документации по PMBOK для разработки ПО.
5. SWEBOOK: свод знаний в области программной инженерии.
6. Методы и стандарты документирования ПО.
7. Методы и стандарты качества разработки ПО .
8. Методы и стандарты проектирования ПО.
9. Методы и стандарты тестирования ПО.
10. Методы и стандарты сопровождения ПО.
11. Методы и стандарты управления конфигурацией ПО.
12. Методы и стандарты верификации и валидации ПО.
13. Методы и стандарты управления требованиями к ПО.
14. Сравнительный анализ стандартов ISO/IEC 12207-95 и ISO/IEC 12207-2008.
15. Характеристика стандартов ГОСТ 34.
16. Стандарт ISO/IEC 15504 (SPICE) оценки процессов разработки и поддержки ПО.
17. Agile-методология XP.
18. Agile-методология Scrum.
19. Agile-методология SAFE.
20. SEI CMMI: модель зрелости процесса разработки.
21. Реинжиниринг ПО.
22. Модификация ПО.
23. Проектирование с повторным использованием программных компонент.
24. Проектные паттерны.
25. Гибкое тестирование.
26. Сравнительный анализ моделей жизненного цикла ПО.
27. Сравнительный анализ моделей оценки трудозатрат на разработку ПО.
28. Управление проектами по разработке программного обеспечения.



29. ITIL/ITSM: методология управления и организации ИТ-услуг.

Тема работы также может быть предложена обучающимся и, после утверждения преподавателем, принята к исполнению.

Каждый год темы могут пересматриваться: изменяться, добавляться, уточняться в связи с появляющимися новыми тенденциями. Список конкретных тем для выбора приведен в приложении к данному руководству



8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Ратушняк, Г.Я. Технологии разработки и проектирования информационных систем. Часть 1 : Учебное пособие / Г.Я. Ратушняк, А.Л. Золкин Электрон. дан. Москва : Русайнс, 2024 201 с. Режим доступа: book.ru Internet access <https://book.ru/book/952868> ISBN 978-5-466-05304-3. [БИК ID: RU\bookru\bibl\952868]
2. Зараменских, Евгений Петрович Разработка информационных систем : учебник и практикум для вузов / Е. П. Зараменских. 2-е изд. Электрон. дан. Москва : Юрайт, 2025 78 с (Высшее образование) URL: <https://urait.ru/bcode/571333> (дата обращения: 01.09.2025). Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользователей <https://urait.ru/bcode/571333> ISBN 978-5-534-21420-8 : 469.00. [БИК ID: RU2fURAIT2f571333]

Дополнительная литература:

1. Жматов, Д. В. Технологии разработки программных приложений, основанные на использовании Git, Docker, Gradle: Практикум [Электронный ресурс] / Жматов Д. В.,Петрова А. А. Москва : РТУ МИРЭА, 2025 70 с. Книга из коллекции РТУ МИРЭА - Информатика <https://e.lanbook.com/book/504851> ISBN 978-5-7339-2592-9. [БИК ID: RU-LAN-BOOK-504851]
2. Грекул, Владимир Иванович Проектирование информационных систем : учебник и практикум для вузов / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. 2-е изд., пер. и доп Электрон. дан. Москва : Юрайт, 2025 404 с (Высшее образование) URL: <https://urait.ru/bcode/560976> (дата обращения: 01.09.2025). Режим доступа: Электронно-библиотечная система Юрайт, для авториз. пользователей <https://urait.ru/bcode/560976> ISBN 978-5-534-19505-7 : 1569.00. [БИК ID: RU2fURAIT2f560976]
3. Брусникин, Г. Н. Разработка UML-моделей при проектировании информационных систем [Электронный ресурс] : учебное пособие / Брусникин Г. Н.,Соколова Н. Ю. Москва : МИЭТ, 2023 52 с. Книга из коллекции МИЭТ - Информатика <https://e.lanbook.com/book/461570> ISBN 978-5-7256-1016-1. [БИК ID: RU-LAN-BOOK-461570]

Нормативные правовые акты:



-

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Электронная библиотека Финансового университета (ЭБ) <http://elib.fa.ru/>
(<http://library.fa.ru/files/elibfa.pdf>)
2. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
3. Электронно-библиотечная система "Университетская библиотека ОНЛАЙН"
<http://biblioclub.ru/>
4. "Деловая онлайн библиотека" издательства "Альпина Паблишер"
<http://lib.alpinadigital.ru/en/library>
5. Электронно-библиотечная система издательства "Лань" <https://e.lanbook.com/>
6. Образовательная платформа "ЮРАЙТ" <https://urait.ru/>
7. Научная электронная библиотека eLibrary.ru <http://elibrary.ru>
8. Информационно-образовательный портал Финуниверситета: <https://org.fa.ru>



10. Методические указания для обучающихся по освоению дисциплины

Основные этапы работы студента по дисциплине **"Методы промышленной разработки информационных систем"**:

1. Предварительная ориентировка в подлежащем изучению учебном материале по программе.
2. Ознакомление с рекомендованной учебной литературой.
3. Слушание и конспектирование лекций, а также выполнение других видов учебной работы.
4. Планирование самостоятельной работы.
5. Обобщение и систематизация информации, почерпнутой из лекций и прочитанной литературы.
6. Выполнение контрольной работы.

Рекомендации по работе с учебным материалом:

1. Осознавайте наличный уровень полученных вами знаний.
2. В ситуации непонимания нужно выявить тот первичный уровень и факторы непонимания, которые стали препятствием понимания последующего.
3. Задавайте сами себе вопросы и пытайтесь ответить на них.

Рекомендации по работе на лекции и с лекционным материалом:

1. Основная задача на лекции – осмысление излагаемого в ней материала. Для этого необходимо слушать лекцию с самого начала, не упуская общих, ориентирующих в материале рассуждений и установок лектора.
2. Ведение записей на лекции важно и полезно для лучшего осмысливания материала, для сохранения информации, с целью ее дальнейшего использования.
3. Для облегчения записи рекомендуется применять сокращения повторяющихся терминов или хорошо известных понятий.

Рекомендации по работе с литературой:

1. Если возникли затруднения при разыскивании материала, по какому-либо конкретному вопросу, следует обратиться к предметному указателю, напечатанному, как правило, в конце каждого литературного источника.
2. Предметный указатель – это алфавитный список основных научных понятий (терминов), содержание которых раскрыто в книге, рядом с термином стоят числа, обозначающие номера страниц, на которых изложен материал, относящийся к данному понятию.

Рекомендации по выполнению контрольной работы:



1. Перед выполнением контрольной работы студент должен изучить соответствующие разделы учебной литературы.
2. Контрольную работу студент должен выполнять самостоятельно, используя те навыки и умения, которые получил на лекциях и практических занятиях.
3. При затруднениях, возникших при выполнении контрольной работы, студент может получить консультацию преподавателя.

Рекомендации по выполнению курсовой работы:

1. Перед выполнением курсовой работы студент должен изучить соответствующие разделы учебной литературы.
2. Курсовую работу студент должен выполнять самостоятельно, используя те навыки и умения, которые получил на лекциях и практических занятиях.
3. При затруднениях, возникших при выполнении курсовой работы, студент может получить консультацию преподавателя.



11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

Комплект лицензионного программного обеспечения:

1. Kaiten
2. Figma
3. Kaspersky
4. Пакет офисных программ
5. Комплект свободно распространяемого программного обеспечения

Современные профессиональные базы данных и информационные справочные системы

1. Информационно-правовая система «Гарант»
2. Информационно-правовая система «Консультант Плюс»
3. Электронная энциклопедия: <http://ru.wikipedia.org/wiki/Wiki>

Сертифицированные программные и аппаратные средства защиты информации

1. не предусмотрены

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

1. **Учебная аудитория** для проведения учебных занятий, предусмотренных программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная оборудованием и техническими средствами обучения: мебель аудиторная (столы, стулья, доска аудиторная), набор демонстрационного оборудования (проектор, экран)
2. **Помещение для самостоятельной работы** обучающихся оснащенное компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа к электронной информационно-образовательной среде Университета.
3. **Компьютерный класс** для проведения учебных занятий, предусмотренных программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенный оборудованием и техническими средствами обучения: мебель аудиторная



(столы, стулья, доска аудиторная), персональные компьютеры, набор демонстрационного оборудования (проектор, экран)