

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гаранин Максим Алексеевич
Должность: Ректор
Дата подписания: 25.03.2026 17:14:41
Уникальный программный ключ:
7708e3a47e66a8ee02711b298d7c78bd1e40bf88

Приложение к рабочей программе дисциплины

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Проектирование и разработка информационных систем

(наименование дисциплины(модуля))

Направление подготовки / специальность

09.04.02 Информационные системы и технологии

(код и наименование)

Направленность (профиль)/специализация

Корпоративные информационные системы

(наименование)

Содержание

- Пояснительная записка.

Типовые контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций. • Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации.

1. Пояснительная записка

Цель промежуточной аттестации – оценивание промежуточных и окончательных результатов обучения по дисциплине, обеспечивающих достижение планируемых результатов освоения образовательной программы.

Формы промежуточной аттестации: *зачет, семестр 3, РГР.*

Экзамен, семестр 4, КР

Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции	Код индикатора достижения компетенции
ОПК-1: Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	ОПК-1.2
ОПК-2: Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	ОПК-2.1
ОПК-5: Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем	ОПК-5.1
	ОПК-5.2
ОПК-6: Способен использовать методы и средства системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий	ОПК-6.1
ОПК-8: Способен осуществлять эффективное управление разработкой программных средств и проектов	ОПК-8.1
	ОПК-8.2

Результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр 4)
ОПК-1.2: Применяет теоретические и экспериментальные исследования объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте	Обучающийся знает: особенности теоретических и экспериментальных исследований объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте	Вопросы (№1 - №5)
	Обучающийся умеет: применять теоретические и экспериментальные исследования объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте	Задание (№1 - №2)
	Обучающийся владеет: навыками применения теоретических и экспериментальных исследований объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте	

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр 4)
ОПК-2.1: Применяет основные методы представления информации и алгоритмы обработки данных в профессиональной деятельности	Обучающийся знает: основные методы представления информации и алгоритмы обработки данных в профессиональной деятельности	Вопросы (№6 - №11)

	Обучающийся умеет: применять основные методы представления информации и алгоритмы обработки данных в профессиональной деятельности	Задание (№3 - №4)
	Обучающийся владеет: навыками применения основных методов представления информации и алгоритмы обработки данных в профессиональной деятельности	

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр 4)
ОПК-5.1: Разрабатывает отдельные этапы технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей	Обучающийся знает: особенности разработки отдельных этапов технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей	Вопросы (№12 - №16)
	Обучающийся умеет: разрабатывать отдельные этапы технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей	Задание (№5 - №6)
	Обучающийся владеет: навыками разработки отдельных этапов технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей	

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр 4)
ОПК-5.2: Модернизирует программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач	Обучающийся знает: особенности модернизации программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач	Вопросы (№17 - №21)
	Обучающийся умеет: модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач	Задание (№7 - №8)
	Обучающийся владеет: навыками модернизации программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач	

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр 4)
ОПК-6.1: Применяет основные положения системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий	Обучающийся знает: основные положения системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий	Вопросы (№22 - №26)
	Обучающийся умеет: применять основные положения системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий	Задание (№9 - №10)
	Обучающийся владеет: навыками применения основных положений системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий	

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр 4)
ОПК-8.1: Применяет методологии эффективного управления разработкой программных средств и проектов	Обучающийся знает: методологии эффективного управления разработкой программных средств и проектов	Вопросы (№27 - №31)
	Обучающийся умеет: применять методологии эффективного управления разработкой программных средств и проектов	Задание (№11 - №12)
	Обучающийся владеет: навыками применения методологий эффективного управления разработкой программных средств и проектов	

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы (семестр 4)
ОПК-8.2: Планирует комплекс работ по разработке программных средств и проектов	Обучающийся знает: особенности планирования комплекса работ по разработке программных средств и проектов	Вопросы (№32 - №36)
	Обучающийся умеет: планировать комплекс работ по разработке программных средств и проектов	Задание (№13 - №14)
	Обучающийся владеет: навыками планирования комплекса работ по разработке программных средств и проектов	

Промежуточная аттестация (зачет) проводится в одной из следующих форм:

- 1) собеседование;
- 2) выполнение заданий в ЭИОС.

Промежуточная аттестация (экзамен) проводится в одной из следующих форм:

- 1) ответ на билет, состоящий из теоретических вопросов и практических заданий;
- 2) выполнение заданий в ЭИОС.

Промежуточная аттестация (курсовая работа) проводится в одной из следующих форм:

- 1) собеседование;
- 2) выполнение заданий в ЭИОС

2. Типовые¹ контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций

2.1 Типовые вопросы (тестовые задания) для оценки знаниевого образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-1.2: Применяет теоретические и экспериментальные исследования объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте	Обучающийся знает: особенности теоретических и экспериментальных исследований объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте
ОПК-2.1: Применяет основные методы представления информации и алгоритмы обработки данных в профессиональной деятельности	Обучающийся знает: основные методы представления информации и алгоритмы обработки данных в профессиональной деятельности
ОПК-5.1: Разрабатывает отдельные этапы технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей	Обучающийся знает: особенности разработки отдельных этапов технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей
ОПК-5.2: Модернизирует программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач	Обучающийся знает: особенности модернизации программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач

¹ Приводятся типовые вопросы и задания. Оценочные средства, предназначенные для проведения аттестационного мероприятия, хранятся на кафедре в достаточном для проведения оценочных процедур количестве вариантов. Оценочные средства подлежат актуализации с учетом развития науки, образования, культуры, экономики, техники, технологий и социальной сферы. Ответственность за нераспространение содержания оценочных средств среди обучающихся университета несут заведующий кафедрой и преподаватель – разработчик оценочных средств.

ОПК-6.1: Применяет основные положения системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий	Обучающийся знает: основные положения системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий
ОПК-8.1: Применяет методологии эффективного управления разработкой программных средств и проектов	Обучающийся знает: методологии эффективного управления разработкой программных средств и проектов
ОПК-8.2: Планирует комплекс работ по разработке программных средств и проектов	Обучающийся знает: особенности планирования комплекса работ по разработке программных средств и проектов

Примеры вопросов

1. Программная инженерия:

+ software engineering

- Инструменты создания программного обеспечения

- Коллектив инженеров-программистов, разрабатывающих программное обеспечение для компьютеров
+ Дисциплина, изучающая применение строгого систематического количественного подхода к разработке, эксплуатации и сопровождению программного обеспечения

- Комплекс программ, предназначенный для решения инженерных задач, связанных с большим количеством расчетов

- Инженерная индустрия применения прикладного программного обеспечения

+ Совокупность инженерных методов и средств создания программного обеспечения

- Прикладное программное обеспечение для решения офисных задач

2. Построение SADT-модели включает в себя выполнение следующих действий:

- Написание программного обеспечения для разрабатываемой системы по требованиям заказчика

+ Сбор информации об объекте, определение его границ

+ Определение цели и точки зрения модели, построение, обобщение и декомпозиция диаграмм

- Представление исследуемой системы в графическом виде

- Представление исследуемого объекта средствами системного моделирования

+ Критическая оценка, рецензирование и комментирование

- Разработка, отладка и тестирование программного обеспечения

- Использование графических пакетов для представления системы в виде модели

3. Моделирование основывается на принципах:

+ Выбор модели оказывает определяющее влияние на подход к решению проблемы и на то, как будет выглядеть это решение

- Декомпозиции системы на отдельные подзадачи

- Инкапсуляции и полиморфизма

- Децентрализации управления системой

+ Каждая модель может быть представлена с различной степенью точности; лучшие модели – те, что ближе к реальности

- Открытой трансформируемой системы

+ Нельзя ограничиваться созданием только одной модели. Наилучший подход при разработке любой нетривиальной системы – использовать совокупность нескольких моделей, почти независимых друг от друга

- Анализа и синтеза проектирования систем

4. В бизнес-процессах выделяют классы процессов:

- Решающие бизнес-процессы
- Регламентирующие бизнес-процессы
- + Основные бизнес-процессы
- Бизнес-процессы поведения системы
- Программируемые бизнес-процессы
- Экономические бизнес-процессы
- + Обеспечивающие бизнес-процессы
- + Бизнес-процессы управления

5. CASE-средства классифицируются по следующим признакам:

- + По применяемым методологиям и моделям систем и БД
- По используемому программному обеспечению
- По этапам жизненного цикла программного обеспечения
- + По степени интегрированности с СУБД
- По уровням детализации и декомпозиции проектируемой системы
- + По доступным платформам
- По используемым языкам программирования
- По степени сложности моделируемой системы

6. К малым интегрированным средствам моделирования относятся:

- ARIS Toolset
- Design/IDEF
- + ERwin
- + BPwin
- Designer/2000
- Paradigm Plus
- + Model Mart
- Rational Rose

7. К средним интегрированным средствам моделирования относятся:

- Rational Rose
- + Design/IDEF
- BPwin
- + Designer/2000
- + ARIS Toolset
- Model Mart
- Paradigm Plus
- ERwin

8. Объектно-ориентированная методология (ООМ) включает в себя составные части:

- + Объектно-ориентированный анализ
- Объектно-ориентированный подкласс

+ Объектно-ориентированное проектирование

- Объектно-ориентированная парадигма
- Объектно-ориентированная экспозиция
- Объектно-ориентированное моделирование
- + Объектно-ориентированное программирование
- Объектно-ориентированная декомпозиция

9. К основным понятиям объектно-ориентированного подхода относятся:

- Обобщение
- + Полиморфизм
- + Инкапсуляция
- Реализация
- Агрегирование
- + Наследование
- Ассоциация
- Композиция

10. Главные принципы объектного подхода:

- + Абстрагирование
- Наследование
- + Ограничение доступа или инкапсуляция
- Безграничный доступ или инкапсуляция
- + Модульность и иерархия
- Агрегирование
- Композиция
- Обобщение и специализация

11. Дополнительные принципы объектного подхода:

- Реализация
- + Типизация
- + Параллелизм
- Внедрение
- Перпендикулярность
- + Сохраняемость или устойчивость
- Несохранимость или неустойчивость
- Динамичность

12. К инструментальным средствам объектно-ориентированного анализа и проектирования относятся:

- + Rational Rose
- Model Mart
- + MS Visio
- + ARIS
- IDEF1X
- Erwin
- BPwin

- JAM

13. К инструментальным средствам представления функциональных моделей относятся:

- JAM

+ Model Mart

- MS Visio

- ARIS

- IDEF0

+ Erwin

+ BPwin

- Rational Rose

14. Методологии, поддерживаемые в BPwin:

- IDEF1X

+ IDEF0

- IDEF1

+ IDEF3

- IDEFX

- IDEF5

+ DFD

- DFD1X

15. Диаграмма IDEF0 может содержать следующие типы диаграмм:

- Диаграмму классов

+ Контекстную диаграмму, диаграмму декомпозиции

- Диаграмму компонентов

+ Диаграмму дерева узлов

- Диаграмму взаимодействий

+ Диаграмму только для экспозиции (FEO)

- Диаграмму последовательности, диаграмму кооперации

- Диаграмму узлов

16. Уровни логической модели:

- Диаграмма сущность

- Диаграмма связь

- Диаграмма пакетов

+ Диаграмма сущность-связь

- Модель данных, основанная на классах

+ Модель данных, основанная на ключах

- Полная операционная модель

+ Полная атрибутивная модель

17. Внутренние стрелки не входящие в состав диаграммы IDEF0:

+ mechanism- output

- output-input

- + mechanism- input
- output-control
- output-input feedback
- output-control feedback
- output-mechanism
- + control feedback- mechanism

18. Типы стрелок не входящие в состав диаграммы IDEF0:

- Input
- + Editor
- Control
- + Properties
- Output
- Mechanism
- Call
- + Dictionary

19. Quick Reports – создание простейших отчетов – позволяет создавать отчеты:

- Group/Totals. Табличный отчет с автоматической группировкой и сортировкой данных
- Report Header. Печатается единожды в начале отчета
- + Columnar. Простой табличный отчет
- Page Header. Печатается в верхней части каждой страницы
- + Vertical. Простой вертикальный отчет
- Group Header. Печатается в начале каждой группы
- + Blank Report. Бланк. Создается пустой бланк отчета, в который не включаются данные
- Detail. Печатается для каждой строчки набора данных

20. BPwin допускает следующие переходы с одной нотации на другую:

- IDEF3 → DFD
- DFD → IDEF0
- + IDEF0 → DFD
- DFD → DFD
- IDEF3 → IDEF0
- + IDEF0 → IDEF3
- IDEF3 → IDEF3
- + DFD → IDEF3

21. DFD описывает:

- Функции обработки стрелок (arrow)
- + Функции обработки информации (работы)
- Внешние ссылки (external references), объекты, сотрудников или отделы, которые участвуют в обработке информации
- + Документы (стрелки, arrow), объекты, сотрудников или отделы, которые участвуют в обработке информации
- Функции обработки внешних ссылок

+ Внешние ссылки (external references), таблицы для хранения документов (хранилище данных, data stor+ E)

- Функции обработки документов

- Документы (стрелки, arrow), объекты, сотрудников или отделы, которые участвуют в обработке внешних стрелок

22. ВРwin позволяет создавать на диаграмме DFD типы граничных стрелок:

+ Обычная граничная стрелка

- Специальная стрелка

- Внутренняя ссылка

+ Межстраничная ссылка и тоннельная стрелка

+ Внешняя ссылка

- Страничная ссылка и тневая стрелка

- Контрольная стрелка

- Стрелка механизм

23. Создать отчет в ВРwin возможно с помощью:

+ Встроенных шаблонов

- Программных модулей, создаваемых разработчиком на языке Visual Basic

- Создать отчет в ВРwin не возможно

+ Report Template Builder

- Отчет создается разработчиком

- Отдельно поставляемых программ

- Встроенных мастер-функций

+ RPTwin

24. В ВРwin 4.0 отчеты могут быть экспортированы в распространенные форматы:

+ Текстовый

- Символьный

+ MS Office

- Графический

+ HTML

- Internet Explorer

- Acrobat

- IBM Rational

25. Поддерживаемые в RPTwin типы операторов:

+ Текстовый оператор конкатенации (&)

- Символ

- Текст

- Дата

+ Арифметические

- Графический оператор конкатенации (&)

+ Логические

- Номер

26. Инструментальное средство ERwin позволяет:

- Редактировать и отлаживать программы
- + Проектировать на физическом и логическом уровне модели данных
- Управлять процессом конструирования ПО
- Проектировать диаграммы вариантов использования и взаимодействий
- + Проводить процессы прямого и обратного проектирования баз данных
- Управлять процессом трансляции и отладки программ
- + Выравнивать модель и содержимое системного каталога после редактирования
- Проектировать контекстные диаграммы и диаграммы декомпозиции

27. ERwin позволяет создавать модели следующих типов:

- + Модель, имеющую только логический уровень
- Модель, имеющую абстрактный уровень
- Модель, имеющую абстрактный и физический уровни
- + Модель, имеющую только физический уровень
- Модель, имеющую абстрактный и логический уровни
- + Модель, имеющую как логический уровень, так и физический уровень
- Модель, имеющую концептуальный уровень
- Модель, имеющую контекстный уровень

28. Для создания моделей ERwin используют международно признанные системы обозначений (нотации):

- IDEF0
- + IDEF1X
- IDEF3
- DFD
- + IE
- + DM
- IDEFDFD
- IDEF3

29. К основным компонентам диаграммы ERwin относятся:

- + Сущности
- Переходы
- + Атрибуты
- Классы
- Слияния
- Разветвления
- Использования
- + Связи

30. Точки зрения организации в ARIS:

- Структура внедрения и структура потоков
- + Организационная структура
- Управленческая структура
- Поведенческая структура

- + Функциональная структура
- Коммуникационная структура
- + Структура данных и структура процессов
- Обобщенная структура

31. Уровни точки зрения в ARIS:

- Описание структуры
- + Описание требований
- Описание поведения
- Описание разработки
- + Описание спецификации
- + Описание внедрения
- Описание процессов
- Описание классов

32. Методы описания, используемые в ARIS:

- EPT – метод описания потоков
- + EPC - метод описания процессов
- ERM - модель сущность-связь для описания структуры объектов
- + ERM - модель сущность-связь для описания структуры данных
- EPP – метод описания пакетов
- EPC – метод описания компонентов
- + UML - унифицированный язык моделирования
- EPT – метод описания нитей

33. К основным компонентам инструментов ARIS Toolset относятся:

- Internet (интернет)
- WordPad (ввод текстовых данных)
- Media (средство для медиа описания моделей)
- + Explorer (проводник)
- Acrobat (чтение текстовых данных)
- + Designer (средство для графического описания моделей)
- Document (для ввода различных параметров и атрибутов) и выноски
- + Таблица (для ввода различных параметров и атрибутов) и мастер (Wizards)

34. ARIS Business Optimizer позволяет:

- + Определять целевые затраты и рассчитывать стоимость продукта: во что компании обходится предоставление отдельных продуктов
- Принимать решения о времени начала и окончания работы над проектом
- + Принимать решения по аутсорсингу: стоит ли поручить выполнение бизнес-процессов внешнему поставщику услуг
- Определять последовательность работ, выполняемых в ходе работы над проектом
- Определять требования к персоналу компании, которая в дальнейшем будет эксплуатировать программное обеспечение
- Рассчитывать заработную плату сотрудников компании после внедрения программного обеспечения

- Планировать требования к обслуживающему персоналу, сопровождающему программное обеспечение
 + Планировать требования к персоналу: сколько необходимо сотрудников для оптимального выполнения работ

35. «Взгляды» ARIS:

- + Процессы
- Потоки
- + Функции (с целями)
- + Данные и организация
- Процедуры
- Управление и внедрение
- Нити
- Память

36. Уровни анализа ARIS для каждого «взгляда»:

- Поведение
- + Требования
- + Спецификации
- Функции
- Процедуры
- Проверка
- + Внедрение
- Тестирование

2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат:

<u>Код и наименование индикатора достижения компетенции</u>	<u>Образовательный результат</u>
ОПК-1.2: Применяет теоретические и экспериментальные исследования объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте	Обучающийся умеет: применять теоретические и экспериментальные исследования объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте
	Обучающийся владеет: навыками применения теоретических и экспериментальных исследований объектов профессиональной деятельности, в том числе в новой или незнакомой среде и в междисциплинарном контексте
ОПК-2.1: Применяет основные методы представления информации и алгоритмы обработки данных в профессиональной деятельности	Обучающийся умеет: применять основные методы представления информации и алгоритмы обработки данных в профессиональной деятельности
	Обучающийся владеет: навыками применения основных методов представления информации и алгоритмы обработки данных в профессиональной деятельности
ОПК-5.1: Разрабатывает отдельные этапы технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей	Обучающийся умеет: разрабатывать отдельные этапы технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей
	Обучающийся владеет: навыками разработки отдельных этапов технологических процессов производства, ремонта, эксплуатации и обслуживания транспортных систем и сетей
ОПК-5.2: Модернизирует программное и аппаратное обеспечение информационных и автоматизированных систем для решения	Обучающийся умеет: модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач

профессиональных задач	Обучающийся владеет: навыками модернизации программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач
ОПК-6.1: Применяет основные положения системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий	Обучающийся умеет: применять основные положения системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий
	Обучающийся владеет: навыками применения основных положений системной инженерии в области получения, передачи, хранения, переработки и представления информации посредством информационных технологий
ОПК-8.1: Применяет методологии эффективного управления разработкой программных средств и проектов	Обучающийся умеет: применять методологии эффективного управления разработкой программных средств и проектов
	Обучающийся владеет: навыками применения методологий эффективного управления разработкой программных средств и проектов
ОПК-8.2: Планирует комплекс работ по разработке программных средств и проектов	Обучающийся умеет: планировать комплекс работ по разработке программных средств и проектов
	Обучающийся владеет: навыками планирования комплекса работ по разработке программных средств и проектов

Примеры заданий:

Задание №1

Выполнить анализ предметной области различными методами: контент-анализ, вебметрический анализ, анализ ситуаций, моделирование и др

Задание №2

Изучение устройств автоматизированного сбора информации

Задание №3

Разработка модели архитектуры информационной системы

Задание №4

Обоснование выбора средств проектирования ИС

Задание №5

Описание бизнес-процессов заданной предметной области

Задание №6

Построение модели управления качеством процесса изучения модуля «Проектирование и разработка информационных систем

Задание №7

Реинжиниринг методом реинтеграции

Задание №8

Разработка требований безопасности ИС

Задание №9

Реинжиниринг бизнес-процессов методом горизонтального и/или вертикального сжатия

Задание №10

Проектирование спецификации информационной системы индивидуальному заданию

Задание №11 Стандарты и профили в области ИС

Цель работы: изучить стандарты и профили в области ИС

Теоретические сведения:

1. Профиль ИС. При создании и развитии сложных, распределенных, тиражируемых программных и информационных систем требуется гибкое формирование и применение согласованных (гармонизированных) совокупностей базовых стандартов и нормативных документов разного уровня, выделение в них требований и рекомендаций, необходимых для реализации заданных функций ИС. Для унификации и регламентирования такие совокупности базовых стандартов должны адаптироваться и конкретизироваться применительно к определенным классам проектов, процессов функций и компонентов разрабатываемых систем. В связи с этой потребностью выделилось и сформировалось понятие профиля как основного инструмента функциональной стандартизации [https://revolution.allbest.ru/programming/00533576_0.html].

Профиль -- это совокупность нескольких (или подмножество одного) базовых стандартов с четко определенными и гармонизированными подмножествами обязательных и факультативных возможностей, предназначенная для реализации заданной функции или группы функций.

Профиль формируется исходя из функциональных характеристик объекта стандартизации. В профиле выделяются и устанавливаются допустимые возможности и значения параметров каждого базового стандарта или нормативного документа, входящего в профиль.

Профиль не должен противоречить использованным в нем базовым стандартам и нормативным документам. Он должен применять выбранные из альтернативных вариантов необязательные возможности и

значения параметров в пределах допустимых.

На базе одной совокупности базовых стандартов могут формироваться и утверждаться различные профили для разных проектов информационных систем. Ограничения базовых документов профиля и их согласованность, проведенная разработчиками профиля, должны обеспечивать качество, совместимость и корректное взаимодействие отдельных компонентов системы, соответствующих профилю, в за данной области его применения.

Базовые стандарты и профили в зависимости от проблемно-ориентированной области применения информационных систем могут использоваться как непосредственные директивные, руководящие или рекомендательные документы, а также как нормативная база, необходимая при выборе или разработке средств автоматизации технологических этапов или процессов создания, сопровождения и развития информационных систем.

Обычно рассматривают две группы профилей:

1) регламентирующие архитектуру и структуру информационной системы

2) регламентирующие процессы проектирования, разработки, применения, сопровождения и развития системы.

В зависимости от области применения профили могут иметь разные категории и соответственно разные статусы утверждения:

1. профили конкретной информационной системы, определяющие стандартизованные проектные решения в пределах данного проекта;

2. профили информационной системы, предназначенные для решения некоторого класса прикладных задач.

Профили информационных систем унифицируют и регламентируют только часть требований, характеристик, показателей качества объектов и процессов, выделенных и формализованных на базе стандартов и нормативных документов. Другая часть функциональных и технических характеристик системы определяется заказчиками и разработчиками творчески, без учета положений нормативных документов.

2. Цели и принципы формирования профилей информационных систем

2.1 Основные цели применения профилей

- снижение трудоемкости, временных затрат, стоимости и улучшение других технико-экономических показателей проектов ИС;
- повышение качества разрабатываемых или применяемых покупных компонентов и ИС в целом;
- обеспечение расширяемости и масштабируемости ИС в зависимости от размера решаемых задач;
- возможность функциональной интеграции в ИС задач, ранее решавшихся отдельно;
- обеспечение переносимости прикладных программных средств (ПС) между разными аппаратно-программными платформами.

2.2 Принципы формирования профиля информационной системы

1. Использование профилей информационных систем призвано решить следующие задачи:

- снижение трудоемкости проектов;
- повышение качества компонентов информационной системы;
- обеспечение расширяемости и масштабируемости разрабатываемых систем;
- обеспечение возможности функциональной интеграции в информационную систему задач, которые раньше решались отдельно;
- обеспечение переносимости прикладного программного обеспечения.

В зависимости от того, какие из указанных задач являются наиболее приоритетными, производится выбор стандартов и документов для формирования профиля.

Актуальность использования профилей информационных систем обусловлена современным состоянием стандартизации информационных технологий, которое характеризуется следующими особенностями:

существует множество международных и национальных стандартов, которые не полностью и неравномерно удовлетворяют потребности в стандартизации объектов и процессов создания и применения сложных информационных систем;

длительные сроки разработки, согласования и утверждения международных и национальных стандартов приводят к их консерватизму и хроническому отставанию от современных информационных технологий;

функциональными стандартами поддержаны и регламентированы только самые простые объекты и рутинные, массовые процессы: телекоммуникации, программирование, документирование программ и данных. Наиболее сложные и творческие процессы создания и развития крупных распределенных информационных систем -- системный анализ и проектирование, интеграция компонентов и систем,

испытания и сертификация -- почти не поддержаны требованиями и рекомендациями стандартов из-за трудности их формализации и унификации;

совершенствование и согласование нормативных и методических документов в ряде случаев позволяют создать на их основе национальные и международные стандарты.

Подходы к формированию профилей информационных систем могут быть различными. В международной функциональной стандартизации информационных технологий принято довольно жесткое понятие профиля. Считается, что его основой могут быть только международные и национальные, утвержденные стандарты. Использование стандартов де-факто и нормативных документов фирм не допускается. При таком подходе затруднены унификация, регламентирование и параметризация множества конкретных функций и характеристик сложных объектов архитектуры и структуры современных информационных систем. Другой подход к разработке и применению профилей информационных систем состоит в использовании совокупности адаптированных и параметризованных базовых международных и национальных стандартов и открытых спецификаций, отвечающих стандартам де-факто и рекомендациям международных консорциумов.

Эталонная модель среды открытых систем (OSE/RM) определяет разделение любой информационной системы на две составляющие: приложения (прикладные программы и программные комплексы) и среду, в которой эти приложения функционируют.

Между приложениями и средой определяются стандартизованные интерфейсы -- ApplicationProgramInterface (API), которые являются необходимой частью профилей любой открытой системы.

Кроме того, в профилях могут быть определены унифицированные интерфейсы взаимодействия функциональных частей друг с другом и интерфейсы взаимодействия между компонентами среды системы. Спецификации выполняемых функций и интерфейсов взаимодействия могут быть оформлены в виде профилей компонентов системы.

Профили информационной системы с иерархической структурой могут включать в себя: стандартизованные описания функций, выполняемых данной системой; функции взаимодействия системы с внешней для нее средой; стандартизованные интерфейсы между приложениями и средой информационной системы; профили отдельных функциональных компонентов, входящих в систему. Для эффективного использования конкретного профиля необходимо:

выделить объединенные логической связью проблемно-ориентированные области функционирования, где могут применяться стандарты, общие для одной организации или группы организаций;

идентифицировать стандарты и нормативные документы, варианты их использования и параметры, которые необходимо включить в профиль;

документально зафиксировать участки конкретного профиля, где требуется создание новых стандартов или нормативных документов, и идентифицировать характеристики, которые могут оказаться важными для разработки недостающих стандартов и нормативных документов этого профиля;

формализовать профиль в соответствии с его категорией, включая стандарты, различные варианты нормативных документов и дополнительные параметры, которые непосредственно связаны с профилем;

опубликовать профиль и/или продвигать его по формальным инстанциям для дальнейшего распространения.

При использовании профилей важное значение имеет обеспечение проверки корректности их применения путем тестирования, испытаний и сертификации. Для этого требуется создание технологии контроля и тестирования в процессе применения профиля. Данная технология должна поддерживаться совокупностью методик, инструментальных средств, составом и содержанием оформляемых документов на каждом этапе выполнения проекта.

Использование профилей способствует унификации при разработке тестов, проверяющих качество и взаимодействие компонентов проектируемой информационной системы. Профили должны определяться таким образом, чтобы тестирование их реализации можно было проводить по возможности наиболее полно по стандартизованной методике. При этом возможно применение ранее разработанных методик, так как международные стандарты и профили являются основой для создания общепризнанных аттестационных тестов.

2.3 Структура профилей информационных систем

Разработка и применение профилей являются органической частью процессов проектирования, разработки и сопровождения информационных систем. Профили характеризуют каждую конкретную информационную систему на всех стадиях ее жизненного цикла, задавая согласованный набор базовых стандартов, которым должна соответствовать система и ее компоненты. Стандарты, важные с точки зрения заказчика, должны задаваться в ТЗ на проектирование системы и составлять ее первичный профиль. То, что не задано в ТЗ, первоначально остается на усмотрение разработчика системы, который, руководствуясь

требованиями ТЗ, может дополнять и развивать профили системы и впоследствии согласовывать их с заказчиком. Таким образом, профиль конкретной системы не является статичным, он развивается и конкретизируется в процессе проектирования информационной системы и оформляется в составе документации проекта системы.

В профиль конкретной системы включаются спецификации компонентов, разработанных в составе данного проекта, и спецификации использованных готовых программных и аппаратных средств, если эти средства не специфицированы соответствующими стандартами. После завершения проектирования и испытаний системы, в ходе которых проверяется ее соответствие профилю, профиль применяется как основной инструмент сопровождения системы при эксплуатации, модернизации и развитии.

2.4 Общая структура профиля информационной системы

Формирование и применение профилей конкретных информационных систем выполняется на основе использования международных и национальных стандартов, ведомственных нормативных документов, а также стандартов де-факто при условии доступности соответствующих им спецификаций.

Для обеспечения корректного применения профилей их описания должны содержать:

- определение целей использования данного профиля;
- точное перечисление функций объекта или процесса стандартизации, определяемого данным профилем;
- формализованные сценарии применения базовых стандартов и спецификаций, включенных в данный профиль;
- сводку требований к информационной системе или ее компонентам, определяющих их соответствие профилю, и требований к методам тестирования соответствия;
- нормативные ссылки на конкретный набор стандартов и других нормативных документов, составляющих профиль, с точным указанием применяемых редакций и ограничений, способных повлиять на достижение корректного взаимодействия объектов стандартизации при использовании данного профиля;
- информационные ссылки на все исходные документы.

На стадиях жизненного цикла информационной системы выбираются и затем применяются основные функциональные профили:

- профиль прикладного программного обеспечения;
- профиль среды информационной системы;
- профиль защиты информации в информационной системе;
- профиль инструментальных средств, встроенных в информационную систему.

План выполнения работы:

1. Получите тему от преподавателя.

Темы практических работ по дисциплине:

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «Продажи: сеть, опт»

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «План закупок, отгрузок (поставок)»

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «Закупки, отгрузки (поставки)»

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «Размещение заказа на производство»

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «Закупка сырья и комплектующих в соответствии с планом»

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «Приемка, отгрузка, выписка»

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «Платежи. Взаиморасчеты с кредиторами»

Спроектировать фрагмент информационной системы, реализующий бизнес процесс «Поступления. Взаиморасчеты с дебиторами»

2. Запишите в отчете:

- стандартизованные описания функций, выполняемых данной системой;
- функции взаимодействия системы с внешней для нее средой;
- стандартизованные интерфейсы между приложениями и средой информационной системы;
- профили отдельных функциональных компонентов, входящих в систему. Для эффективного использования конкретного профиля необходимо:

- выделить объединенные логической связью проблемно-ориентированные области функционирования, где могут применяться стандарты, общие для одной организации или группы организаций;

- идентифицировать стандарты и нормативные документы, варианты их использования и

параметры, которые необходимо включить в профиль;

- документально зафиксировать участки конкретного профиля, где требуется создание новых стандартов или нормативных документов, и идентифицировать характеристики, которые могут оказаться важными для разработки недостающих стандартов и нормативных документов этого профиля;
- формализовать профиль в соответствии с его категорией, включая стандарты, различные варианты нормативных документов и дополнительные параметры, которые непосредственно связаны с профилем.

Задание №12. Методологии и технологии проектирования ИС

Цель работы: изучить требования к методологии и технологии проектирования ИС, методологию RAD.

Теоретические сведения

Общие требования к методологии и технологиям

Методологии, технологии и инструментальные средства проектирования (CASE-средства) составляют основу проекта любой ИС. Методология реализуется через конкретные технологии и поддерживающие их стандарты, методики и инструментальные средства, которые обеспечивают выполнение процессов ЖЦ. Технология проектирования определяется как совокупность трех составляющих:

- пошаговой процедуры, определяющей последовательность технологических операций проектирования (рис. 1);
- критериев и правил, используемых для оценки результатов выполнения технологических операций;
- нотаций (графических и текстовых средств), используемых для описания проектируемой системы.



Рисунок - Представление технологической операции проектирования

Технологические инструкции, составляющие основное содержание технологии, должны состоять из описания последовательности технологических операций, условий, в зависимости от которых выполняется та или иная операция, и описаний самих операций.

Технология проектирования, разработки и сопровождения ИС должна удовлетворять следующим общим требованиям:

- технология должна поддерживать полный ЖЦ ПО;
- технология должна обеспечивать гарантированное достижение целей разработки ИС с заданным качеством и в установленное время;
- технология должна обеспечивать возможность выполнения крупных проектов в виде подсистем (т.е. возможность декомпозиции проекта на составные части, разрабатываемые группами исполнителей ограниченной численности с последующей интеграцией составных частей). Опыт разработки крупных ИС показывает, что для повышения эффективности работ необходимо разбить проект на отдельные слабо связанные по данным и функциям подсистемы. Реализация подсистем должна выполняться отдельными группами специалистов. При этом необходимо обеспечить координацию ведения общего проекта и исключить дублирование результатов работ каждой проектной группы, которое может возникнуть в силу наличия общих данных и функций;
- технология должна обеспечивать возможность ведения работ по проектированию отдельных подсистем небольшими группами (3-7 человек). Это обусловлено принципами управляемости коллектива и повышения производительности за счет минимизации числа внешних связей;
- технология должна обеспечивать минимальное время получения работоспособной ИС. Речь идет не о

сроках готовности всей ИС, а о сроках реализации отдельных подсистем. Реализация ИС в целом в короткие сроки может потребовать привлечения большого числа разработчиков, при этом эффект может оказаться ниже, чем при реализации в более короткие сроки отдельных подсистем меньшим числом разработчиков. Практика показывает, что даже при наличии полностью завершеного проекта, внедрение идет последовательно по отдельным подсистемам;

- технология должна предусматривать возможность управления конфигурацией проекта, ведения версий проекта и его составляющих, возможность автоматического выпуска проектной документации и синхронизацию ее версий с версиями проекта;

- технология должна обеспечивать независимость выполняемых проектных решений от средств реализации ИС (систем управления базами данных (СУБД), [операционных систем](#), языков и систем программирования);

- технология должна быть поддержана комплексом согласованных CASE-средств, обеспечивающих [автоматизацию](#) процессов, выполняемых на всех стадиях ЖЦ.

Реальное применение любой технологии проектирования, разработки и сопровождения ИС в конкретной организации и конкретном проекте невозможно без выработки ряда стандартов (правил, соглашений), которые должны соблюдаться всеми участниками проекта. К таким стандартам относятся следующие:

стандарт проектирования;

- стандарт оформления проектной документации;
- стандарт пользовательского интерфейса.

Стандарт проектирования должен устанавливать:

- набор необходимых моделей (диаграмм) на каждой стадии проектирования и степень их детализации;
- правила фиксации проектных решений на диаграммах, в том числе: правила именования объектов (включая соглашения по терминологии), набор атрибутов для всех объектов и правила их заполнения на каждой стадии, правила оформления диаграмм, включая требования к форме и размерам объектов, и т. д.;

- требования к конфигурации рабочих мест разработчиков, включая настройки операционной системы, настройки CASE-средств, общие настройки проекта и т. д.;

- механизм обеспечения совместной работы над проектом, в том числе: правила интеграции подсистем проекта, правила поддержания проекта в одинаковом для всех разработчиков состоянии (регламент обмена проектной информацией, механизм фиксации общих объектов и т.д.), правила проверки проектных решений на непротиворечивость и т. д.

Стандарт оформления проектной документации должен устанавливать:

комплектность, состав и структуру документации на каждой стадии проектирования;

- требования к ее оформлению (включая требования к содержанию разделов, подразделов, пунктов, таблиц и т.д.),

- правила подготовки, рассмотрения, согласования и утверждения документации с указанием предельных сроков для каждой стадии;

- требования к настройке издательской системы, используемой в качестве встроенного средства подготовки документации;

- требования к настройке CASE-средств для обеспечения подготовки документации в соответствии с установленными требованиями.

Стандарт интерфейса пользователя должен устанавливать:

- правила оформления экранов (шрифты и цветовая палитра), состав и расположение окон и элементов управления;

- правила использования клавиатуры и мыши;

- правила оформления текстов помощи;

- перечень стандартных сообщений;

- правила обработки реакции пользователя.

Методология RAD

Одним из возможных подходов к разработке ПО в рамках спиральной модели ЖЦ является получившая в последнее время широкое распространение методология быстрой разработки приложений RAD (Rapid Application Development). Под этим термином обычно понимается процесс разработки ПО, содержащий 3 элемента:

- небольшую команду программистов (от 2 до 10 человек);

- короткий, но тщательно проработанный производственный график (от 2 до 6 мес.);

- повторяющийся цикл, при котором разработчики, по мере того, как приложение начинает обретать форму, запрашивают и реализуют в продукте требования, полученные через взаимодействие с заказчиком.

Команда разработчиков должна представлять из себя группу профессионалов, имеющих опыт в анализе, проектировании, генерации кода и тестировании ПО с использованием CASE-средств. Члены коллектива должны также уметь трансформировать в рабочие прототипы предложения конечных пользователей.

Жизненный цикл ПО по методологии RAD состоит из четырех фаз:

- фаза анализа и планирования требований;

- фаза проектирования;

- фаза построения;
- фаза внедрения.

На фазе анализа и планирования требований пользователи системы определяют функции, которые она должна выполнять, выделяют наиболее приоритетные из них, требующие проработки в первую очередь, описывают информационные потребности. Определение требований выполняется в основном силами пользователей под руководством специалистов-разработчиков. Ограничивается масштаб проекта, определяются временные рамки для каждой из последующих фаз. Кроме того, определяется сама возможность реализации данного проекта в установленных рамках финансирования, на данных аппаратных средствах и т.п. Результатом данной фазы должны быть список и приоритетность функций будущей ИС, предварительные функциональные и информационные модели ИС.

На фазе проектирования часть пользователей принимает участие в техническом проектировании системы под руководством специалистов-разработчиков. CASE-средства используются для быстрого получения работающих прототипов приложений. Пользователи, непосредственно взаимодействуя с ними, уточняют и дополняют требования к системе, которые не были выявлены на предыдущей фазе. Более подробно рассматриваются процессы системы. Анализируется и, при необходимости, корректируется функциональная модель. Каждый процесс рассматривается детально. При необходимости для каждого элементарного процесса создается частичный прототип: экран, диалог, отчет, устраняющий неясности или неоднозначности. Определяются требования разграничения доступа к данным. На этой же фазе происходит определение набора необходимой документации.

После детального определения состава процессов оценивается количество функциональных элементов разрабатываемой системы и принимается решение о разделении ИС на подсистемы, поддающиеся реализации одной командой разработчиков за приемлемое для RAD-проектов время - порядка 60 - 90 дней. С использованием CASE-средств проект распределяется между различными командами (делится функциональная модель). Результатом данной фазы должны быть:

- общая информационная модель системы;
- функциональные модели системы в целом и подсистем, реализуемых отдельными командами разработчиков;
- точно определенные с помощью CASE-средства интерфейсы между автономно разрабатываемыми подсистемами;
- построенные прототипы экранов, отчетов, диалогов.

Все модели и прототипы должны быть получены с применением тех CASE-средств, которые будут использоваться в дальнейшем при построении системы. Данное требование вызвано тем, что в традиционном подходе при передаче информации о проекте с этапа на этап может произойти фактически неконтролируемое искажение данных. Применение единой среды хранения информации о проекте позволяет избежать этой опасности.

В отличие от традиционного подхода, при котором использовались специфические средства прототипирования, не предназначенные для построения реальных приложений, а прототипы выбрасывались после того, как выполняли задачу устранения неясностей в проекте, в подходе RAD каждый прототип развивается в часть будущей системы. Таким образом, на следующую фазу передается более полная и полезная информация.

На фазе построения выполняется непосредственно сама быстрая разработка приложения. На данной фазе разработчики производят итеративное построение реальной системы на основе полученных в предыдущей фазе моделей, а также требований нефункционального характера. Программный код частично формируется при помощи автоматических генераторов, получающих информацию непосредственно из репозитория CASE-средств. Конечные пользователи на этой фазе оценивают получаемые результаты и вносят коррективы, если в процессе разработки система перестает удовлетворять определенным ранее требованиям. Тестирование системы осуществляется непосредственно в процессе разработки. После окончания работ каждой отдельной командой разработчиков производится постепенная интеграция данной части системы с остальными, формируется полный программный код, выполняется тестирование совместной работы данной части приложения с остальными, а затем тестирование системы в целом. Завершается физическое проектирование системы:

- определяется необходимость распределения данных;
- производится анализ использования данных;
- производится физическое проектирование базы данных;
- определяются требования к аппаратным ресурсам;
- определяются способы увеличения производительности;
- завершается разработка документации проекта.

Результатом фазы является готовая система, удовлетворяющая всем согласованным требованиям.

На фазе внедрения производится обучение пользователей, организационные изменения и параллельно с внедрением новой системы осуществляется работа с существующей системой (до полного внедрения новой). Так как фаза построения достаточно непродолжительна, планирование и подготовка к внедрению должны начинаться

заранее, как правило, на этапе проектирования системы. Приведенная схема разработки ИС не является абсолютной. Возможны различные варианты, зависящие, например, от начальных условий, в которых ведется разработка: разрабатывается совершенно новая система; уже было проведено обследование предприятия и существует модель его деятельности; на предприятии уже существует некоторая ИС, которая может быть использована в качестве начального прототипа или должна быть интегрирована с разрабатываемой.

Следует, однако, отметить, что методология RAD, как и любая другая, не может претендовать на универсальность, она хороша в первую очередь для относительно небольших проектов, разрабатываемых для конкретного заказчика. Если же разрабатывается типовая система, которая не является законченным продуктом, а представляет собой комплекс типовых компонент, централизованно сопровождаемых, адаптируемых к программно-техническим платформам, СУБД, средствам телекоммуникации, организационно-экономическим особенностям объектов внедрения и интегрируемых с существующими разработками, на первый план выступают такие показатели проекта, как управляемость и качество, которые могут войти в противоречие с простотой и скоростью разработки. Для таких проектов необходимы высокий уровень планирования и жесткая дисциплина проектирования, строгое следование заранее разработанным протоколам и интерфейсам, что снижает скорость разработки.

Методология RAD неприменима для построения сложных расчетных программ, операционных систем или программ управления космическими кораблями, т.е. программ, требующих написания большого объема (сотни тысяч строк) уникального кода.

Не подходят для разработки по методологии RAD приложения, в которых отсутствует ярко выраженная интерфейсная часть, наглядно определяющая логику работы системы (например, приложения реального времени) и приложения, от которых зависит безопасность людей (например, управление самолетом или атомной электростанцией), так как итеративный подход предполагает, что первые несколько версий наверняка не будут полностью работоспособны, что в данном случае исключается.

Оценка размера приложений производится на основе так называемых функциональных элементов (экраны, сообщения, отчеты, файлы и т.п.) Подобная метрика не зависит от языка программирования, на котором ведется разработка.

В качестве итога перечислим основные принципы методологии RAD
[http://gendocs.ru/v395/%D0%BB%D0%B5%D0%BA%D1%86%D0%B8%D0%B8_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%B8%D1%81?page=2]:

- разработка приложений итерациями;
- необязательность полного завершения работ на каждом из этапов жизненного цикла;
- обязательное вовлечение пользователей в процесс разработки ИС;
- необходимое применение CASE-средств, обеспечивающих целостность проекта;
- применение средств управления конфигурацией, облегчающих внесение изменений в проект и сопровождение готовой системы;
- необходимое использование генераторов кода;
- использование прототипирования, позволяющее полнее выяснить и удовлетворить потребности конечного пользователя;
- тестирование и развитие проекта, осуществляемые одновременно с разработкой;
- ведение разработки немногочисленной хорошо управляемой командой профессионалов;
- грамотное руководство разработкой системы, четкое планирование и контроль выполнения работ.

Порядок выполнения работы

Реализуйте для поставленной задачи 4 фазы:

- фаза анализа и планирования требований;
- фаза проектирования;
- фаза построения;
- фаза внедрения.

Задание №13. Системное проектирование ИС

Теоретические сведения

Структурным анализом принято называть метод исследования системы, которое начинается с ее общего обзора и затем детализируется, приобретая иерархическую структуру с все большим числом уровней. Решение трудных проблем путем их разбиения на множество меньших независимых задач (так называемых «черных ящиков») и организация этих задач в древовидные иерархические структуры значительно повышают понимание сложных систем. В инженерии ПО (software engineering), Структурный анализ (Structured Analysis, SA) и одноименное с ним Структурное проектирование (Structured Design, SD) – это методы для анализа и преобразования бизнес-требований в спецификации и, в конечном счете, в компьютерные программы, конфигурации аппаратного обеспечения и связанные с ними ручные процедуры. Структурный анализ, СА (Structured Analysis, SA) и Структурное проектирование, СП (Structured Design, SD) являются фундаментальными инструментами системного анализа и развивались из классического системного анализа 1960-70-х годов.

Структурный подход заключается в поэтапной декомпозиции системы при сохранении целостности о ней

представления Основные принципы структурного подхода (первые два являются основными) [http://elar.urfu.ru/bitstream/10995/28812/1/978-5-91128-072-7_2014.pdf]:

1) принцип «разделяй и властвуй» – принцип решения сложных проблем путем их разбиения на множество меньших независимых задач, легких для понимания и решения;

2) принцип иерархического упорядочивания – принцип организации составных частей проблемы в иерархические древовидные структуры с добавлением новых деталей на каждом уровне.

3) принцип абстрагирования – заключается в выделении существенных аспектов системы и отвлечения от несущественных;

4) принцип формализации – заключается в необходимости строгого методического подхода к решению проблемы;

5) принцип непротиворечивости – заключается в обоснованности и согласованности элементов.

SADT (Structured Analysis and Design Technique) – это методология инженерии разработки ПО (software engineering) для описания систем в виде иерархии функций (функциональной структуры).

SADT использует два типа диаграмм: 1) модели деятельности (activity models); 2) модели данных (data models).

IDEF0 (Integration Definition for Function Modeling) – методология функционального моделирования для описания функций предприятия, предлагающая язык функционального моделирования для анализа, разработки, реинжиниринга и интеграции информационных систем бизнес процессов; или анализа инженерии разработки ПО (or software engineering analysis). Модель IDEF0 – это графическое описание (информационной) системы или предметной области (subject), которое разрабатывается с определенной целью с выбранной точки зрения. Модель IDEF0 представляет собой набор из одной или более (иерархически связанных) IDEF0-диаграмм, которые описывают функции системы или предметной области (subject area) с помощью графики, текста и глоссария.

SSADM (Structured Systems Analysis and Design Method) – системный подход к анализу и проектированию ИС. SSADM как комплект стандартов для системного анализа и разработки приложений был разработан в начале 1980-х для Центрального агентства по компьютерам и телекоммуникациям (Central Computer and Telecommunications Agency, сейчас это Office of Government Commerce) – государственного учреждения UK, заинтересованного в использовании технологии в управлении.

Порядок выполнения работы

Изобразить изученные виды диаграмм для своей информационной системы.

Задание №14. Детальное проектирование ИС

Порядок выполнения работы

1. Анализ материалов и разработка технико-экономического обоснования с техническим заданием на проектирование ИС.

2. Обоснуйте выбор проектных решений.

3. Опишите реальные компоненты ИС.

4. Оформите технический проект (ТП).

5. Обоснуйте выбор или разработку математических методов или алгоритмов программ.

6. Проведите корректировку построенных структур БД.

7. Выбор комплекса технических средств, разработка техно-рабочего проекта. Инструментальные средства и платформы реализации проекта.

Тематика курсовых работ:

1. «Разработка ИС для учета основных средств на производстве»
2. «Разработка ИС для учета денежных средств на производстве»
3. «Разработка ИС для учета и анализа финансовых результатов на производстве»
4. «Разработка ИС для управления проектами»
5. «Разработка ИС туроператора для туристического агентства»
6. «Разработка ИС для ведения расписания учебных занятий»
7. «Разработка ИС для учета товаров в мелкооптовой торговой фирме»
8. «Разработка ИС для оценщика автомобилей и дорогостоящего оборудования»
9. «Разработка ИС для учета движения продукции на оптовом складе»
10. «Разработка ИС для автоматизации документооборота на кафедре ВУЗа»
11. «Разработка ИС для учета и анализа затрат на выпуск и реализацию готовой продукции на производстве»
12. «Разработка ИС для управления запасами на производстве»
13. «Разработка ИС для учета расчетов по долгам предприятия»
14. «Разработка ИС для учета движения готовой продукции на предприятии»
15. «Разработка ИС для библиотеки ВУЗа»
16. «Разработка ИС для аудитора»
17. «Разработка ИС для учета движения продукции на аптечном складе»

Тематика расчетно-графических работ

1. Системы управления цифровыми технологиями предприятия

Цель работы: научиться исследовать предприятие с точки зрения цифровых технологий, предлагать методы усовершенствования

Порядок выполнения работы:

1. По заданию преподавателя или самостоятельно выбрать предприятие, имеющее развитую цифровую инфраструктуру.
2. Выделить направления производственной деятельности, построить организационную структуру в виде схемы (используя MS Visio или другие специализированные программные продукты)
3. Построить таблицу "функции подразделений"
4. Изобразить существующие информационные взаимосвязи между подразделениями
5. Изобразить схемы внутреннего и внешнего документооборота.
6. Определить требуемые учетные подсистемы, охватывающие несколько подразделений, каждое из которых заинтересовано в оперативности и актуальности данных.
7. Разработать рекомендации по усовершенствованию документооборота, исключающие дублирование информации.
8. Изучить текущий уровень автоматизации: определить перечень разработанных подсистем, состав автоматизированных рабочих мест и круг решаемых задач с целью определения функциональной полноты системы и автоматизацией учетных функций.
9. Разработать предложения по требуемому составу выбранных подсистем КИС (Корпоративная Информационная Система), уточнению перечня задач, подлежащих автоматизации, и расширению состава автоматизированных рабочих мест с целью получения полной оперативной информации по оперативному и управленческому учету производственной деятельности компании, обеспечивающих принятие верного управленческого решения в режиме реального времени.
10. Определить используемые программное, информационное обеспечения
11. Исследовать состояние существующего компьютерного парка с целью разработки предложений по использованию новых информационных технологий, предложений по модернизации или расширению компьютерного парка.
12. Исследовать существующие бизнес-процессы и бизнес- процедуры. Произвести сравнительный анализ технологий управления организацией, существующего документооборота с технологиями.
13. Сформировать организационно-функциональную схему автоматизации. Сформировать поэтапный календарный план внедрения

2. Управление и аудит цифровых технологий

Цель работы: научиться управлять цифровыми технологиями предприятия

Порядок выполнения работы:

1. На основе предыдущей лабораторной работы предложить ERP-систему или обосновать существующую:

- SCM-системы управления цепочками поставок
- WMS-системы управления складом
- RMS-системы управления ритейлом
- Space management systems- системы для мерчендайзинга
- MRP системы управления производством
- Service desk системы поддержки
- HR системы
- CRM-системы управления отношениями с клиентами
- Workflow-системы управления работами и документами
- Data quality системы контроля за качеством данных
- Системы управления финансами
- Системы управления разработкой и автоматизации ИТ службы

2. Выбрать систему автоматизации работы ИТ службы:

- Системы управления проектами

- Intranet-системы. Системы управления документооборотом внутри компании. Системы хранения информации внутри компании
 - Extranet системы-системы построения внешних сайтов и Интернет торговли
 - Интернет как мощный канал развития предприятия. Управление сайтами и интернет проектами холдинга
 - Базы данных. Мощные не реляционные Berkley DB, Constant DB, Mongo DB. Традиционные реляционные MySQL, MS SQL, Oracle. Сравнение. Условия применимости
3. Обосновать выбор СУБД для проекта.
- Выбор информационной системы и подрядчика
 - Выбор между собственной разработкой, заказной разработкой и покупным ПО
 - Выбор между компанией подрядчиком и фрилансерами
 - Критерии выбора тиражного ПО
 - Характеристики программных продуктов, не очевидные ИТ-специалисту
 - Полезность и потребительская ценность
 - Можно ли оценить отдельно полезность программного обеспечения
 - Совокупная стоимость владения
 - Когда полезны тендеры
 - Когда нужны подрядчики
 - Технология проведения тендеров
 - Тендерная документация
 - Конкурсное предложение
 - Опросники и предпроектное обследование
 - Интересы менеджера по продажам. Его стратегия поведения
 - Интересы руководителя тендера. Его стратегия поведения
 - Интересы менеджера проекта со стороны подрядчика
 - Методы манипуляции на тендерах
 - Договора с подрядчиками и поставщиками. Уловки при заключении договоров

2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

Вопросы к зачету

1. Что представляет собой модель в нотации IDEFO?
2. Что обозначают работы в IDEFO?
3. Назовите порядок наименования работ?
4. Какое количество работ должно присутствовать на одной диаграмме?
5. Что называется порядком доминирования?
6. Как располагаются работы по принципу доминирования?
7. Каково назначение сторон прямоугольников работ на диаграммах?
8. Перечислите типы стрелок.
9. Назовите виды взаимосвязей.
10. Что называется граничными стрелками?
11. Объясните принцип именования разветвляющихся и сливающихся стрелок.
12. Какие методологии поддерживаются BPWin?
13. Перечислите основные элементы главного окна BPWin.
14. Опишите процесс создания новой модели в BPWin.
15. Как провести связь между работами?
16. Как задать имя работы.
17. Опишите процесс декомпозиции работы.
18. Как добавить работу на диаграмму?
19. Как разрешить туннелированные стрелки?
20. Может ли модель BPWin содержать диаграммы нескольких методологий?
21. Что описывает диаграмма DFD?
22. Какая нотация используется в BPWin для построения диаграмм DFD?
23. Что описывает диаграмма IDEF3?
24. Перечислите составные части диаграммы DFD.
25. В чем состоит назначение процесса?
26. Что называется внешней сущностью?

27. Что описывают хранилища?
28. Объясните механизм дополнения диаграммы IDEF0 диаграммой DFD.
29. Перечислите составные элементы диаграмм IDEF3.
30. Что показывают связи в диаграммах IDEF3?
31. Перечислите типы стрелок в диаграммах IDEF3.
32. Что называется перекрестком?
33. Назовите типы перекрестков.
34. Что называется объектом-ссылкой?
35. Какие бывают типы объектов-ссылок?
36. Как добавить объект-ссылку?
37. Назовите уровни методологии IDEF1X.
38. Из каких моделей состоит логический уровень?
39. Из каких моделей состоит физический уровень?
40. Что включает в себя диаграмма сущность-связь?
41. Что включает в себя модель данных, основанная на ключах?
42. Какую информацию содержит трансформационная модель?
43. Сформулируйте требования, в которых необходимо убедиться перед началом проектирования БД.
44. Что называется моделью СУБД?
45. Перечислите преимущества от использования CASE-средства ERWin.
46. В чем смысл варианта использования?
47. Каково назначение диаграмм вариантов использования?
48. Назовите основные свойства вариантов использования.
49. Назовите основные компоненты диаграмм вариантов использования.
50. Каково назначение диаграмм классов?
51. Для чего используется диаграмма классов на стадии анализа?
52. Для чего используется диаграмма классов на стадии проектирования?
53. Назовите основные компоненты диаграмм классов.
54. Назовите основные типы статических связей между классами.
55. Что представляет собой ассоциация?
56. В чем смысл множественности ассоциаций?
57. В чем отличие атрибутов от ассоциаций?
58. Каково назначение диаграмм взаимодействия?
59. Как относятся между собой диаграммы вариантов использования и диаграммы взаимодействия?
60. Каково назначение диаграмм состояния?
61. Как отображаются действия и деятельности на диаграммах состояния?
62. Что такое условный переход и как он описывается на диаграмме?
63. В чем отличие диаграмм пакетов от диаграмм классов?
64. В чем смысл зависимости между элементами диаграммы пакетов?
65. Какие виды диаграмм используются для генерации кода?
66. Какие компоненты исходного кода генерирует Rational Rose?
67. Как в исходном коде отражаются атрибуты и операции класса?
68. Виды моделей жизненного цикла ИС?
69. Как построить график проекта?
70. Как определить критический путь на графике проекта?

Вопросы к экзамену

1. Управление взаимодействием с внешними участниками проекта.
2. Виды ИС предприятий, поддерживающие различные производственные циклы.
3. Виды ИС, поддерживающие процесс принятия решений.
4. Функциональные возможности, структура и особенности различных ИС.
5. Этапы управления проектом ИС.
6. Инициирование проекта ИС.
7. Организация проектного коллектива.
8. Этапы управления проектом ИС.
9. Определение структуры проекта ИС.
10. Планирование как основа управления проектом ИС.
11. Планирование как основа управления проектом ИС.
12. Организация выполнения проекта ИС.

13. Контроль подготовки и выполнения проекта ИС
14. Координация подготовки и выполнения проекта ИС.
15. UML –моделирование и и управления требованиями.
16. Системы контроля и управления доступом как компонент ИС систем.
17. Понятие жизненного цикла ИС. Существующие модели жизненного цикла ИС.
18. Информационные инструменты, обеспечивающие управление проектами.
19. Стратегия предприятия и основа целенаправленного управления проектами создания ИС.
20. Проектные отклонения. Риски, проблемы, изменения.
21. Методология функционального моделирования IDEF0.
22. Критерии сравнительной оценки CASE-средств.
23. Разработка и визуальное моделирование с помощью языка UML.
24. Средство для проектирования и разработки ПО с использованием моделей.
25. Средства разработки и создания информационных систем масштабов предприятия.
26. Характеристика существующих CASE-средств.
27. Технология внедрения CASE-средств.
28. Функционально-ориентированный и объектно-ориентированный подходы в проектировании.

3. Методические материалы, определяющие процедуру и критерии оценивания сформированности компетенций при проведении промежуточной аттестации

Критерии формирования оценок по ответам на вопросы, выполнению тестовых заданий

- оценка **«отлично»** выставляется обучающемуся, если количество правильных ответов на вопросы составляет 100 – 90% от общего объема заданных вопросов;
- оценка **«хорошо»** выставляется обучающемуся, если количество правильных ответов на вопросы – 89 – 76% от общего объема заданных вопросов;
- оценка **«удовлетворительно»** выставляется обучающемуся, если количество правильных ответов на тестовые вопросы – 75–60 % от общего объема заданных вопросов;
- оценка **«неудовлетворительно»** выставляется обучающемуся, если количество правильных ответов – менее 60% от общего объема заданных вопросов.

Критерии формирования оценок по результатам выполнения заданий

«Отлично/зачтено» – ставится за работу, выполненную полностью без ошибок и недочетов.

«Хорошо/зачтено» – ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов.

«Удовлетворительно/зачтено» – ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и двух недочетов.

«Неудовлетворительно/не зачтено» – ставится за работу, если число ошибок и недочетов превысило норму для оценки «удовлетворительно» или правильно выполнено менее 2/3 всей работы.

Виды ошибок:

- *грубые ошибки: незнание основных понятий, правил, норм; незнание приемов решения задач; ошибки, показывающие неправильное понимание условия предложенного задания.*

- *негрубые ошибки: неточности формулировок, определений; нерациональный выбор хода решения.*

- *недочеты: нерациональные приемы выполнения задания; отдельные погрешности в формулировке выводов; небрежное выполнение задания.*

Критерии формирования оценок по написанию и защите курсовой работы

«Отлично» (5 баллов) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями, в которой отражены все необходимые результаты проведенного анализа, сделаны обобщающие выводы и предложены рекомендации в соответствии с тематикой курсовой работы, а также грамотно и исчерпывающе ответившие на все встречные вопросы преподавателя.

«Хорошо» (4 балла) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями, в которой отражены все необходимые результаты проведенного анализа, сделаны обобщающие выводы и предложены рекомендации в соответствии с тематикой курсовой работы. При этом при ответах на вопросы преподавателя обучающийся допустил не более двух ошибок.

«Удовлетворительно» (3 балла) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями. При этом при ответах на вопросы преподавателя обучающийся допустил более трёх ошибок.

«Неудовлетворительно» (0 баллов) – ставится за курсовую работу, если число ошибок и недочетов превысило удовлетворительный уровень компетенции

Критерии формирования оценок по зачету

«Зачтено» - обучающийся демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки, освоил вопросы практического применения полученных знаний, не допустил фактических ошибок при ответе, достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности.

«Не зачтено» - выставляется в том случае, когда обучающийся демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У экзаменуемого слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки, допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки.

Критерии формирования оценок по экзамену

«Отлично» (5 баллов) – обучающийся демонстрирует знание всех разделов изучаемой дисциплины: содержание базовых понятий и фундаментальных проблем; умение излагать программный материал с демонстрацией конкретных примеров. Свободное владение материалом должно характеризоваться логической ясностью и четким видением путей применения полученных знаний в практической деятельности, умением связать материал с другими отраслями знания.

«Хорошо» (4 балла) – обучающийся демонстрирует знания всех разделов изучаемой дисциплины: содержание базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки, освоил вопросы практического применения полученных знаний, не допустил фактических ошибок при ответе, достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности. Таким образом данная оценка выставляется за правильный, но недостаточно полный ответ.

«Удовлетворительно» (3 балла) – обучающийся демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. Однако знание основных проблем курса не подкрепляется конкретными практическими примерами, не полностью раскрыта сущность вопросов, ответ недостаточно логичен и не всегда последователен, допущены ошибки и неточности.

«Неудовлетворительно» (0 баллов) – выставляется в том случае, когда обучающийся демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У экзаменуемого слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки, допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки.

