
**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования

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

Кафедра ЭМИС

Вагнер Д.П.

Методические указания по проведению лабораторных и самостоятельной
работе студентов по курсу

«Проектирование информационных систем»
для направления 230200 – Информационные системы

Томск 2012

Целью лабораторных работ и индивидуальных заданий для самостоятельной работы по курсу **«Проектирование информационных систем»** студентам направления 230200 – «Информационные системы» является закрепление теоретических знаний по курсу и овладение навыками концептуального, логического и физического проектирования информационных систем, а также применение CASE-средств для автоматизации разработки готовых систем.

СОДЕРЖАНИЕ

Введение.....	4
ЛАБОРАТОРНАЯ РАБОТА №1. Технологии концептуального проектирования.....	4
ЛАБОРАТОРНАЯ РАБОТА №2. Автоматизированное проектирование систем с использованием CASE-средства ErWin	9
ЛАБОРАТОРНАЯ РАБОТА №3. Проектирование и разработка экономических информационных систем.....	12
ЛАБОРАТОРНАЯ РАБОТА №4. Автоматизированное проектирование систем с использованием CASE-средства BPWin	15
Методические указания по самостоятельной работе	19
ТЕОРЕТИЧЕСКАЯ ЧАСТЬ.....	19
Форма контроля	20
Приложение А.....	22

Введение

Цель лабораторных работ — овладение навыками концептуального, логического и физического проектирования информационных систем, а также применение CASE-средств для автоматизации разработки готовых систем.

Методические указания к лабораторным работам по курсу «Проектирование информационных систем» содержат только те понятия и определения, которые необходимы для их успешного выполнения.

ЛАБОРАТОРНАЯ РАБОТА №1. Технологии концептуального проектирования

Цель работы: получение навыков работы по концептуальному проектированию информационных систем, практическое освоение основных приемов и правил методологии информационного моделирования;

Темы для предварительного изучения. Основные этапы проектирования. Современные средства проектирования и программирования баз данных, освоение способов разработки концептуальной модели и реализация модели средствами СУБД. Нормализация БД.

Задание 1.1. Произвести анализ предметной области книжного издательства. Составить концептуальную модель (сущности, атрибуты, связи) и итоговую полноатрибутную ER-диаграмму.

Введение. Основная цель системы обработки данных заключается в повышении эффективности работы компании, учреждения или организации. Система обработки данных должна:

- обеспечивать получение общих или детализированных данных по итогам работы;

- позволять легко определять тенденции изменения важнейших показателей;

- обеспечивать получение информации, критической по времени, без существенной задержки;

- выполнять точный и полный анализ данных.

Одной из популярных среди настольных СУБД является Microsoft Access. Основными преимуществами являются: популярность среди многих конечных пользователей и осуществление высокой устойчивости данных, простота в освоении, использовании непрофессиональными программистами, возможность подготавливать отчеты из баз данных различных форматов произвольной формы на основании различных данных; возможность разработки некоммерческих приложений.

Описание предметной области. База данных создаётся для информационного обслуживания редакторов, менеджеров и других сотрудников компании. БД должна содержать данные о сотрудниках компании, книгах, авторах, финансовом состоянии компании и предоставлять возможность получать разнообразные отчёты.

В соответствии с предметной областью система строится с учётом следующих особенностей:

- каждая книга издаётся в рамках контракта;
- книга может быть написана несколькими авторами;
- контракт подписывается одним менеджером и всеми авторами книги;
- каждый автор может написать несколько книг (по разным контрактам);
- порядок, в котором авторы указаны на обложке, влияет на размер гонорара;
- если сотрудник является редактором, то он может работать одновременно над несколькими книгами;

- у каждой книги может быть несколько редакторов, один из них – ответственный редактор;
- каждый заказ оформляется на одного заказчика;
- в заказе на покупку может быть перечислено несколько книг.

В результате анализа должны быть получены базовые сущности этой предметной области:

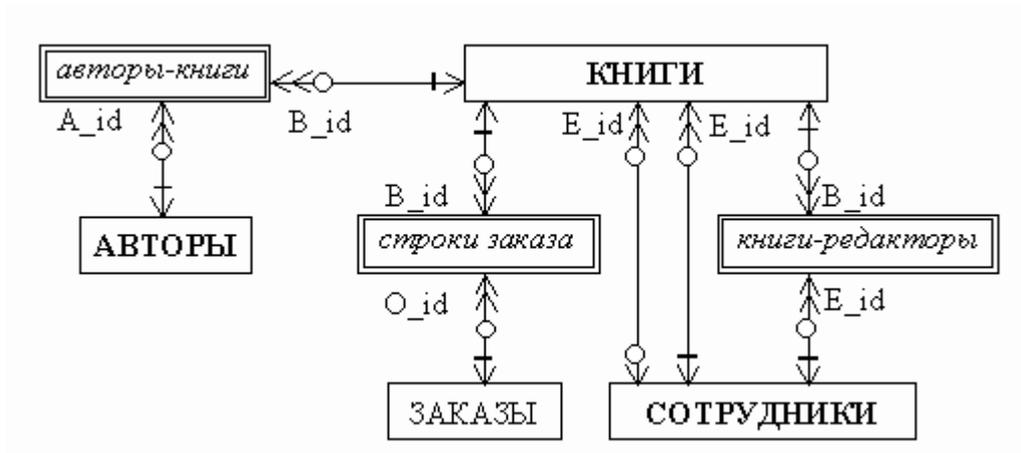
- **Сотрудники** компании. Атрибуты сотрудников – ФИО, табельный номер, пол, дата рождения, паспортные данные, ИНН, должность, оклад, домашний адрес и телефоны. Для редакторов необходимо хранить сведения о редактируемых книгах; для менеджеров – сведения о подписанных контрактах.
- **Авторы**. Атрибуты авторов – ФИО, ИНН (индивидуальный номер налогоплательщика), паспортные данные, домашний адрес, телефоны. Для авторов необходимо хранить сведения о написанных книгах.
- **Книги**. Атрибуты книги – авторы, название, тираж, дата выхода, цена одного экземпляра, общие затраты на издание, авторский гонорар.

Контракты рассматриваются как связь между авторами, книгами и менеджерами. Атрибуты контракта – номер, дата подписания и участники.

Для отражения финансового положения компании в системе нужно учитывать **заказы** на книги. Для заказа необходимо хранить номер заказа, заказчика, адрес заказчика, дату поступления заказа, дату его выполнения, список заказанных книг с указанием количества экземпляров.

Проверочная ER–диаграмма издательской компании приведена в приложении А.

Задание 1.2. Полученную модель реализовать в виде схемы БД MS Access путем сопоставления каждой сущности и каждой связи, имеющей атрибуты, отношения (таблицы БД).



При создании схемы БД обратить внимание на следующие вопросы:

1. Связь типа 1:1 – обязательная связь, например между КНИГАМИ и КОНТРАКТАМИ. Такие отношения следует объединять в одно. Дополнительный эффект от объединения этих отношений – слияние связей авторы–контракты и авторы–книги: ведь в нашем случае контракт заключается именно для написания книги. Исключение для связи типа 1:1 составляют ситуации, когда для увеличения производительности системы в отдельную таблицу выделяются редко используемые данные большого объёма.

2. Связь типа 1:n (один-ко-многим) между отношениями реализуется через внешний ключ. Ключ вводится для того отношения, к которому осуществляется множественная связь. Например, связь «редактировать» между отношениями КНИГИ и СОТРУДНИКИ принадлежит к типу n:m (многие-ко-многим). Этот тип связи реализуется через вспомогательное отношение, которое является соединением первичных ключей соответствующих отношений.

3. Бинарная связь между отношениями не может быть обязательной для обоих отношений. После объединения сущностей КНИГИ и КОНТРАКТЫ остаётся три связи, обязательные для всех участников: между авторами и книгами и между заказами и строками заказов. Такой тип связи означает, что, например, прежде чем добавить новый заказ в отношение

ЗАКАЗЫ, нужно добавить новую строку в отношении СТРОКИ ЗАКАЗА, и наоборот. Поэтому для такой связи необходимо снять с одной стороны условие обязательности. Так как все эти связи будут реализованы с помощью внешнего ключа, снимем условие обязательности связей для отношений, содержащих первичные ключи.

Особое внимание необходимо уделить вопросам **нормализации**:

1. **1НФ**. Для приведения таблиц к 1НФ требуется составить прямоугольные таблицы (один атрибут – один столбец) и разбить сложные атрибуты на простые, а многозначные атрибуты вынести в отдельные отношения.

2. **2НФ**. Неключевые атрибуты отношений должны функционально полно зависеть от первичных ключей.

3. **3НФ**. В отношении *ЗАКАЗЫ* атрибут *Адрес заказчика* не должен зависеть от атрибута *Заказчик*, поэтому адрес следует вынести в отдельное отношение *ЗАКАЗЧИКИ*. Но при этом первичным ключом нового отношения станет атрибут *Заказчик*, т.е. длинная символьная строка. Целесообразнее перенести в новое отношение атрибуты *Заказчик* и *Адрес заказчика* и ввести для него суррогатный ПК. Так как каждый заказчик может сделать несколько заказов, связь между отношениями *ЗАКАЗЧИКИ* и *ЗАКАЗЫ* будет 1:n и суррогатный ПК станет внешним ключом для отношения *ЗАКАЗЫ*. Аналогично необходимо отследить зависимости и между другими отношениями в БД

Задание 1.3. Реализовать запросы в рамках БД.

1. получение списка всех текущих проектов (книг, находящихся в печати и в продаже);
2. получение списка редакторов, работающих над книгами;
3. получение полной информации о книге (проекте);
4. получение сведений о конкретном авторе (с перечнем всех книг);

5. получение информации о продажах (по одному или по всем проектам);
6. определение общей прибыли от продаж по текущим проектам;
7. определение размера гонорара автора по конкретному проекту.

Задание 1.4. Аналогично заданиям 1.1-1.3 произвести проектирование ИС(выделить базовые сущности, связи между ними, составить ER-дигramму), разработать схему БД и реализовать около 10 типовых запросов в соответствии с вариантом выданным преподавателем.

Варианты для задания 1.4:

1. Информационная система библиотеки
2. Информационная система ВУЗа
3. Информационная система швейного производства
4. Информационная система ресторана
5. Информационная система больницы
6. Информационная система склада
7. Информационная система зоопарка
8. Информационная система аэропорта
9. Информационная система аптеки
10. Информационная система автомастерской
11. Информационная система школы
12. Информационная система фотоцентра

ЛАБОРАТОРНАЯ РАБОТА №2. Автоматизированное проектирование систем с использованием CASE-средства ErWin

Цель работы: получение навыков работы с современными методами и средствами автоматизированного проектирования информационных систем на примере программного средства ErWin.

Темы для предварительного изучения. Фирмы-поставщики CASE-средств. Программные средства поддержки жизненного цикла ПО. Оценка и выбор CASE-средств. Технология внедрения CASE-средств

Задание 2.1. Ознакомление с назначением CASE-технологии на примере ErWin, предназначенного для построения логических и физических моделей предметных областей, проведения анализа и генерации готовых БД.

Для создания моделей данных в ERwin можно использовать две нотации: IDEF1X и IE (Information Engineering). В данной работе будет использоваться нотация IDEF1X. Для внесения сущности в модель необходимо кликнуть по кнопке сущности на панели инструментов (ERwin Toolbox), затем кликнуть по тому месту на диаграмме, где Вы хотите расположить новую сущность. Кликнув правой кнопкой мыши по сущности и выбрав из всплывающего меню пункт Entity Editor... можно вызвать диалог Entity Editor, в котором определяются имя, описание и комментарии сущности.

Каждый атрибут хранит информацию об определенном свойстве сущности. Каждый экземпляр сущности должен быть уникальным. Атрибут или группа атрибутов, которые идентифицируют сущность, называется первичным ключом. Для описания атрибутов следует, кликнув правой кнопкой по сущности, выбрать в появившемся меню пункт Attribute Editor.

Для установки связи между сущностями нужно воспользоваться кнопками в палитре инструментов. На логическом уровне можно установить идентифицирующую связь один ко многим, связь многие ко многим и неидентифицирующую связь один ко многим (соответственно кнопки - слева направо в палитре инструментов). Идентифицирующая связь устанавливается между независимой (родительский конец связи) и зависимой (дочерний конец связи) сущностями. Зависимая сущность изображается

прямоугольником со скругленными углами. Экземпляр зависимой сущности определяется только через отношение к родительской сущности. При установлении идентифицирующей связи атрибуты первичного ключа родительской сущности переносятся в состав первичного ключа дочерней сущности (миграция атрибутов). В дочерней сущности они помечаются как внешний ключ - (FK). При установлении неидентифицирующей связи дочерняя сущность остается независимой, а атрибуты первичного ключа родительской сущности мигрируют в состав неключевых компонентов родительской сущности.

Задание 2.2. Разработать концептуальную модель издательства из задания 1.2. и отобразить эту модель в среде ErWin.

Задание 2.3. Сгенерировать полученную модель в реальную СУБД на примере СУБД MS Access и mysql. Изучить особенности генерации SQL-кода.

После завершения проектирования модель может быть перенесена в среду целевой СУБД-сервера. Для этого нужно выбрать в главном меню Tasks / Forward Engineer. Можно либо сгенерировать схему БД, либо скрипт на диалекте SQL, соответствующем заранее выбранному серверу. Возможна обратная задача - по существующей схеме БД сгенерировать графическую модель данных. Возможно также выравнивание схемы БД с моделью данных. Для этого следует использовать соответствующую кнопку в панели инструментов.

Задание 2.4. С помощью CASE-средства ErWin осуществить проектирование ПО из задания 1.4 в соответствии с вариантом задания, выданным преподавателем.

ЛАБОРАТОРНАЯ РАБОТА №3. Проектирование и разработка экономических информационных систем

Цель работы: изучение преимуществ и особенностей проектирования и разработки информационных систем в среде 1С.

Темы для предварительного изучения. Современные средства проектирования и программирования информационных систем и баз данных.

Задание 3.1. Ознакомление с особенностями организации функционирования информационных систем в среде 1С.

Главным технологическим понятием, которое используется в 1С:Предприятии, является **информационная база**. Каждая информационная база содержит в себе не только данные, но и все объекты и механизмы для её ввода и обработки. В среде 1С существует большое количество готовых информационных баз, исполненных в виде поставляемых конфигураций, например, 1С Бухгалтерия, 1С Склад и т.д. Конфигурации в процессе установки разворачиваются в информационные системы, готовые к использованию и далее пользователи системы осуществляют лишь заполнение данных.

В системе 1С данные могут храниться в блоках называемыми объектами конфигурации, такими как справочники, документы, константы, регистры и т.д.. Все объекты группируются по типам и собираются в дерево. Это очень удобно при проектировании, ведь достаточно взглянуть на дерево конфигурации, чтобы понять из чего она состоит.

Справочники хранят условно-постоянную информацию сгруппированную по какому либо критерию или признаку, к примеру справочник "Контрагенты" хранит информацию о фирмах, с которыми организация имеет взаимоотношения. Основное предназначение

справочников - ускоренный ввод информации в систему, а так же избежание дублирования информации или разночтений.

Предназначение регистров подразумевает хранение какой-либо информации - ресурса в разрезе времени (в "периоде") и нескольких аналитических срезов. Для примера можно рассмотреть курсы валют. Валюта это аналитический срез, дата курса это временной срез, а значение курса - это ресурс. Такой тип хранения данных обусловлен возможностью, или лучше сказать - необходимостью быстрым доступом к ключевым данным. Зная значения измерений, данные из регистра, хранящего сотни тысяч строк можно получить за мгновения, простым наложением фильтра - так называемое, адресное обращение к данным. Хранение данных в регистре производится в разрезе нескольких уникальных показателей, называемых измерениями. Хранимые в регистре данные называются ресурсами, так же имеются поля для хранения сопутствующей, не играющей важной роли или не подходящей для измерения или ресурса, информации называемые реквизитами (для примера, в реквизиты можно записать комментарий неограниченной длины, в то время как в измерение или ресурс строки неограниченной длины поместить нельзя).

Подробнее рассмотрим регистры накопления. В ресурсы регистра накопления можно выбрать только числовые значения. В реквизиты регистра накопления – данные любого типа.

Регистры накопления могут быть двух типов - остаточный и оборотный. Остаточный регистр способен хранить промежуточные итоги, которые рассчитываются в начале каждого периода. Это делается для ускорения расчетов, например, гораздо проще посчитать 100 записей, отталкиваясь от остатка на начало месяца, чем 10 000, отталкиваясь от начала ведения учета в программе. В противоположность остаточным регистрам накопления – оборотные регистры накопления, промежуточные итоги по ним не рассчитываются и не хранятся. В таких регистрах обычно хранят данные, которые важны в какой-то определенный промежуток времени, который

может сдвигаться. Основное назначение оборотных регистров - накопление управленческих данных.

Документы предназначены для отображения в хронологическом порядке каждой операции совершаемой организацией в процессе своей деятельности. Например, поступление товаров регистрируется документом "Приходная накладная", продажа товаров документом "Расходная накладная" и т.д.

У документов, как и у справочников, имеется два predeterminedных реквизита это **номер** и **дата**. Документы имеют два режима записи. «Режим записи» - документ только фиксируется в базе, и «режим проведения» - документ фиксируется в базе и формирует записи в регистрах накопления.

Задание 3.2. Анализ предметной области торговой компании, проектирование исходной конфигурации системы.

Торговая компания, осуществляет покупку товаров у поставщиков и их реализацию покупателям. У компании есть круг постоянных поставщиков и покупателей. Операции купли - продажи выполняются как в рублях, так и в иностранной валюте.

Основные операции:

1. Закупка товаров
2. Продажа товаров

Основные справочники:

1. Контрагенты
2. Номенклатура(Товары)
3. Валюта

Основные реквизиты накопления:

1. Остаток товаров на начало месяца
2. Остаток товаров на начало года

Основные документы:

1. Покупка товаров

2. Продажа товаров

Основные отчеты:

1. Отчет о движении товара

На основании анализа предметной области необходимо произвести проектирование и разработку итоговой информационной системы торговой компании, создав указанные объекты конфигурации.

Задание 3.3. Осуществить проектирование ПО в среде 1С из задания 1.4 в соответствии с вариантом задания, выданным преподавателем.

ЛАБОРАТОРНАЯ РАБОТА №4. Автоматизированное проектирование систем с использованием CASE-средства BPWin

Цель работы: получение навыков работы с современными методами и средствами автоматизированного проектирования информационных систем на примере программного средства BPWin.

Темы для предварительного изучения. Фирмы-поставщики CASE-средств. Программные средства поддержки жизненного цикла ПО. Оценка и выбор CASE-средств. Технология внедрения CASE-средств

Задание 4.1. Ознакомление с назначением CASE-технологии на примере BPWin, освоение основных инструментов BPWin.

Порядок работы:

1. Ознакомиться с принципами построения модели в BPwin.
2. Ознакомиться с основными инструментами.
3. Ознакомиться с палитрой инструментов IDEF0.

4. Научиться строить контекстную диаграмму, определять цель, точку зрения, границы модели. Освоить работу с закладками General, Purpose, Definition, Status, Numbering, Display.

VPwin - CASE-средство, помогающее проводить анализ и реорганизацию бизнес-процессов. Поддерживает методология IDEF0 (функциональная модель), IDEF3 (Work Flow Diagram), DFD (Data Flow Diagram). Функциональная модель предназначена для описания существующих бизнес-процессов на предприятии (так называемая модель AS-IS) и идеального положения вещей – того, к чему надо стремиться (модель TO-BE). Процесс построения информационной модели в VPwin состоит из следующих шагов: построение контекстной диаграммы; проводится функциональная декомпозиция; после каждого сеанса декомпозиции проводится сеанс экспертизы. В зависимости от выбранной методологии программой автоматически подбирается нужная панель инструментов VPwin Toolbox. В VPwin существует три разных панели инструментов — по числу поддерживаемых программой методологий.

Построение модели IDEF0. Контекстная диаграмма

Функциональное моделирование является технологией анализа системы в целом как набора связанных между собой действий или функций. Действия системы анализируются независимо от объектов, которые обеспечивают их исполнение. Моделировать деловой процесс можно исходя из различных перспектив и временных рамок. Процесс моделирования какой-либо системы в IDEF0 начинается с определения КОНТЕКСТА, т.е. наиболее абстрактного уровня описания системы в целом. В контекст входит определение субъекта моделирования, цели и точки зрения на модель. Под субъектом понимается сама система, при этом необходимо точно установить ГРАНИЦЫ СИСТЕМЫ, определить, что входит в систему, а что лежит за ее пределами. То есть необходимо решить, что будет

рассматриваться как компоненты системы, а что как внешнее воздействие. Другими словами, первоначально необходимо определить область (Scope) моделирования.

Наименование функции самого высокого уровня описывает систему непосредственно и, как правило, состоит из одного активного глагола в сочетании с обобщающим существительным, которое разъясняет цель деятельности с точки зрения самого общего взгляда на систему. Например «Изготовить изделие».

Формулировка цели моделирования (Purpose) позволяет сфокусировать усилия в нужном направлении. Модель не может быть построена без четко сформулированной цели. Точку зрения можно представить как взгляд человека, который видит систему в нужном для моделирования аспекте. Точка зрения должна соответствовать цели моделирования. Очевидно, что описание работы предприятия с точки зрения финансиста и технолога будет выглядеть совершенно по-разному, поэтому в течение моделирования важно оставаться на выбранной точке зрения. Для определения контекста модели в VPwin следует выбрать пункт меню Model/Model Properties. В закладке General указывается наименование и сведения об авторе модели, в закладку Purpose следует внести цель и точку зрения, а в закладку Definition – определение модели и описание области.

Для создания контекстной диаграммы необходимо сначала создать новую модель, выбрав пункт «New» в меню «File». В появившемся диалоге необходимо набрать имя модели и выбрать ее тип. Этот диалог также отображается при запуске VPwin. После создания модели можно задать ее параметры. Кроме вышеперечисленных свойств модели (Model Properties) можно задать состояние, в котором находится модель, например «в работе» или «для публикации» (закладка Status). Каждый блок может иметь различные типы связанных с ним стрелок. Стрелки обозначают людей, место,

вещи, понятия или события. Стрелки связывают границы диаграммы с блоками, а также действия (блоки) на диаграмме между собой. В диаграммах IDEF0 имеется четыре основных типа стрелок.

Вход блока представляет материал или информацию, которая должна быть использована или преобразована блоком, чтобы произвести продукцию (выпуск). Стрелки входа всегда направляются в левую сторону блока. Стрелки входа необязательны, так как не все действия могут преобразовать или изменять (заменять) что-либо. Каждый блок должен иметь, по крайней мере, одну стрелку контроля (управления). Управление всегда входит в вершину блока. Управление, как правило, представляется в виде правил, инструкций, политики компании, процедур или стандартов. Оно влияет на деятельность без фактического преобразования чего-либо. Управление может также использоваться для описания процедуры начала или окончания выполнения действия. Стрелки выхода (выпуска) — это материал или информация, произведенная блоком. Каждый блок должен иметь, по крайней мере, одну стрелку выхода (выпуска).

Задание 4.2. Согласно варианту из задания 1.4, создайте контекстную диаграмму. Определите цель, точку зрения модели самостоятельно. Опишите свойства в соответствующих закладках диалога Model Properties..

Методические указания по самостоятельной работе

1. Математические и методологические аспекты проектирования информационных систем.

2. Унифицированный язык визуального моделирования(UML). Этапы проектирования ИС с использованием UML.

3.Проектирование пользовательского интерфейса систем.
Проектирование оконного и web-интерфейсов

4. Проектирование распределенных информационных систем

5. Технологии проектирования экономических информационных систем.

6. Использование методологии RAD при проектировании систем

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Тема 1. Математические и методологические аспекты проектирования информационных систем.

Вопросы для рассмотрения

1. Модели выбора проектных решений
2. Разработка модели системы на основе сетей Петри

Тема 2. Унифицированный язык визуального моделирования(UML).
(25 часов)

Вопросы для рассмотрения

1. Особенности разработки ИС с использованием UML.
2. Этапы проектирования ИС с использованием UML.

Тема 3. Проектирование пользовательского интерфейса систем.

Вопросы для рассмотрения

1. Принципы проектирования интерфейса, ориентированного на пользователя.
2. Проектирование web-интерфейса.

Тема 4. Планирование и управление проектами с использованием MS Project.

Вопросы для рассмотрения

1. Этапы планирования проекта в MS Project
2. Основные бизнес-процессы.
3. Планирование ресурсной базы проекта.

Тема 5. Технологии проектирования экономических информационных систем.

Вопросы для рассмотрения

1. Стадии и этапы проектирования ЭИС.
2. Особенности проектирования корпоративных ЭИС.
3. Особенности разработки ЭИС в среде 1С.

Тема 6. Использование методологии RAD при проектировании систем (14 часов)

Вопросы для рассмотрения

1. Особенности методологии RAD
2. Отличия ЖЦ ИС при использовании RAD
3. Примеры использования методологии.

ФОРМА КОНТРОЛЯ

1. Сообщения в форме докладов, индивидуальный опрос.
2. Индивидуальные отчеты, опрос по принципу коллоквиумов.

Список рекомендованной литературы

1. Боровской И.Г. Технология разработки программных средств. Уч.пособие. Томск: ТУСУР, 2005. – 300 с.
2. Вендров А.М. Практикум по проектированию программного обеспечения экономических информационных систем. – М.: Финансы и статистика, 2002.
3. Грекул В.И. Проектирование информационных систем. Курс лекций : Учебное пособие для вузов / В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина. - М. : Интернет-Университет Информационных Технологий, 2005. - 298с. : ил.
4. Григорьева В. 1С: бухгалтерия и 1С: торговля : Практическое пособие / Виктория Григорьева. - СПб. : Невский Диалект, 2001. - 286 с. : ил.
5. Карпова Т.С. Базы данных: модели, разработка, реализация : учебное пособие / Т. С. Карпова. - СПб. : Питер, 2002. - 303[1] с. : ил.
6. Ковалев С.П. Формальный подход к разработке программных систем: Уч.пособие / НГУ. Новосибирск, 2004. 180с.
7. Ларман К. Применение UML и шаблонов проектирования. Пер с англ. - М.: Вильнюс, 2001.
8. Орлов С.А. Технологии разработки программного обеспечения. Разработка сложных программных систем: Учебное пособие для вузов/ Сергей Александрович. - СПб.: Питер, 2002. - 464 с.:
9. Сибилёв В.Д. Базы данных : учебное пособие. Федеральное агентство по образованию, Томский государственный университет систем управления и радиоэлектроники, Кафедра автоматизированных систем управления. - Томск : ТУСУР, 2007. - 278[1] с. : ил., табл.
10. Хорев П. Б. Технологии объектно-ориентированного программирования: Учебное пособие для вузов. - М.: Академия, 2004. – 446 с.

Приложение А

Пример ER-диаграммы издательства

