ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ](#_Toc199081711) 4

[1 Разработка технического проекта](#_Toc199081712) 7

1.1 Постановка задачи…….………………………………………………………7

1.2 Требования к проекту.………………………………………………………...8

[2 Методологии функционального моделирования](#_Toc199081713) 11

3 Моделирование функциональных требований к БД 15

[ЗАКЛЮЧЕНИЕ](#_Toc199081712) 22

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ](#_Toc199081713) 23

ВВЕДЕНИЕ

Информация в современном мире превратилась в один из наиболее важных ресурсов, а информационные системы (ИС) стали необходимым инструментом практически во всех сферах деятельности.

Информационная система – это программный комплекс, функции которого состоят в поддержке надежного хранения информации в памяти компьютера, выполнении специфических для данного приложения преобразований информации и вычислений, предоставления для пользователя удобного и легко осваиваемого интерфейса.

Практически в любой области деятельности люди используют тот или иной вид моделей (математических, физических или компьютерных), чтобы иметь более ясное представление о том, что они делают. Разнообразие задач, решаемых с помощью ИС, привело к появлению множества разнотипных систем, отличающихся принципами построения и заложенными в них правилами обработки информации [1].

Методология проектирования информационных систем описывает процесс создания и сопровождения систем в виде жизненного цикла (ЖЦ) ИС, представляя его как некоторую последовательность стадий и выполняемых на них процессов (разработка требований, проектирование, реализация, тестирование, вход в действие).

Проектирование ИС охватывает три основные области:

* проектирование объектов данных, которые будут реализованы в базе данных;
* проектирование программ, экранных форм, отчетов, которые будут обеспечивать выполнение запросов к данным;
* учет конкретной среды или технологии, а именно: топологии сети, конфигурации аппаратных средств, используемой архитектуры (файл-сервер или клиент-сервер), параллельной обработки, распределенной обработки данных и т.п.

Проектирование информационных систем всегда начинается с определения цели проекта. В общем виде цель проекта можно определить как решение ряда взаимосвязанных задач, включающих в себя обеспечение на момент запуска системы и в течение всего времени ее эксплуатации:

* требуемой функциональности системы и уровня ее адаптивности к изменяющимся условиям функционирования;
* требуемой пропускной способности системы;
* требуемого времени реакции системы на запрос;
* безотказной работы системы;
* необходимого уровня безопасности;
* простоты эксплуатации и поддержки системы.

Сегодняшняя революция в информационных технологиях изменяет традиционные представления о ведении бизнеса. Электронная коммерция уравнивает шансы в привычной конкурентной борьбе, позволяя как крупным, так и мелким компаниям состязаться на равных в качестве, цене, ассортименте продукции. При затратах, равных стоимости рекламного объявления в местной газете, она дает торговым организациям доступ к мировому рынку, недоступному для малых компаний никакими другими средствами [2].

Сегодня у каждой фирмы может быть собственный электронный магазин, с помощью которого она предлагает свои товары всем пользователям Интернет. Ведь они могут в любой момент зайти в такой магазин и сразу же заказать нужный товар. Производить покупки в электронном магазине можно откуда угодно и когда угодно. Для этого нужен только компьютер и любое оборудование, необходимое для связи с сетью. Кроме того, такой магазин никогда не закрывается. Покупатели могут заходить в него в любое время года и суток.

Целью выполнения данной курсовой работы является разработка информационной системы «Интернет-магазин компьютеров», обеспечивающей продажу компьютеров и комплектующих посредством оформления интернет-заказов.

Для достижения поставленной цели необходимо решить задачи создания функциональной и логической схемы данной ИС, модели функционирования и диаграммы деятельности информационной системы «Интернет-магазин компьютеров».

1. РАЗРАБОТКА ТЕХНИЧЕСКОГО ПРОЕКТА
   1. Постановка задачи

Магазин компьютеров предлагает возможность приобретения своих товаров через Интернет. Клиент может выбрать компьютер на web-странице магазина. Компьютеры подразделяются на серверы, настольные, портативные и карманные. Заказчик может выбрать стандартную конфигурацию из списка и детально ознакомиться с ней на отдельной web-странице. Если стандартная конфигурация ему не подходит, он может построить требуемую ему конфигурацию в диалоговом режиме. Компоненты конфигурации (такие, как оперативная память, процессор, жесткий диск и т. п.) представляются как список для выбора из доступных альтернатив. Для каждой новой конфигурации система может подсчитать цену. Чтобы оформить заказ, клиент должен заполнить электронную форму с адресами для доставки товара и отправки счета-фактуры, а также деталями, касающимися оплаты. Оплата компьютеров осуществляется наличными курьеру, осуществляющему доставку, или банковским переводом на счет интернет-магазина. После ввода заказа система отправляет клиенту по электронной почте сообщение с подтверждением получения заказа вместе с относящимися к нему деталями (стоимость, номер счета, банковские реквизиты для безналичной оплаты и т. п.). Пока клиент ожидает прибытия компьютера, он может проверить состояние заказа (поставлен в очередь / собран / отправлен). Работник магазина проверяет, поступила ли оплата (в случае безналичного расчета) и делает соответствующую пометку при поступлении денег. Если деньги не поступают в течение 5 банковских дней, заказ аннулируется. После оплаты или в случае оплаты наличными работник печатает счет-фактуру и отправляет ее на склад вместе с требованием заказанной конфигурации. Заказ помечается как поставленный в очередь. Собранный компьютер вместе со счетом-фактурой и накладной передается со склада в отдел доставки, при этом заказ помечается как собранный. Компьютер поставляется клиенту (статус заказа – отправлен). Если заказ оплачивается наличными, курьер по возращении передает деньги в кассу, заказ помечается как оплаченный. По окончании работы с заказом, он помечается в системе как выполненный. Заказы хранятся в системе в течение 15 месяцев с момента создания для составления годовых и квартальных отчетов, после чего автоматически удаляются.

* 1. Требования к проекту

Требования к функциональным характеристикам

Система должна выполнять следующие функции:

* Формирование заказа;
* Подбор компьютеров в зависимости от требований клиента;
* Подсчёт стоимости выбранного товара;
* Доставка товара клиенту;
* Предоставление отчёта о сформированном заказе;
* Предоставление возможности просмотра состояния заказа.

Исходные данные:

* Предлагаемый товар;
* Адрес для доставки;
* Метод оплаты.

Результаты:

* Отчёт о сформированном заказе;
* Состояние заказа;
* Счёт-фактура;
* Требование заказанной конфигурации;
* Накладная;
* Доставленный заказ.

Требования к надежности

Для обеспечения надежности информационной системы «Интернет-магазин компьютеров» необходимо:

* Проверка на заполнение всех полей формы заказа;
* Проверка на корректность вводимых данных (адрес электронной почты, наличие цифр в Фамилии и имени и т.д.)

Требования к техническим средствам

Система может работать как на IBM совместимых компьютерах, так и на ноутбуках, нетбуках, сотовых телефонах с выходом в интернет.

Минимальная конфигурация:

Наличие выхода в интернет.

Требования к информационной и программной совместимости

Информационная система «Интернет-магазин компьютеров» может работать под управлением любого семейства операционных систем.

Требования к программной документации

Разрабатываемая система должна включать справочную информацию о работе системы и подсказки пользователю.

В состав сопровождающей документации должны входить:

* Пояснительная записка.
* Руководство пользователя.

1. Методологии функционального моделирования

Разработка базы данных невозможна без ее тщательного проектирования: слишком велико влияние этого шага на последующие этапы жизненного цикла информационной системы, в основе которой лежит создаваемая база данных.

Для целей проектирования информационной системы могут быть использованы следующие виды моделей:

* методология функционального моделирования работ SADT (Structured Analysis and Design Technique);
* диаграммы потоков данных DFD (Data Flow Diagrams);
* методология объектного проектирования на языке UML (UML-диаграммы).

Методология SADT (Structured Analisys and Design Technique - технология структурного анализа и проектирования) разработана Дугласом Т. Россом и является одной из самых известных и широко используемых методик проектирования. Новое название методики, принятое в качестве стандарта, -IDEF0 (Icam DEFinition) является частью программы ICAM (Integrated Computer -Aided Manufacturing - интегрированная компьютеризация производства) [3].

Процесс моделирования в SADT включает сбор информации об исследуемой области, документирование полученной информации, представление ее в виде модели и уточнение модели. Кроме того, этот процесс подсказывает вполне определенный путь выполнения согласованной и достоверной структурной декомпозиции, что является ключевым моментом в квалифицированном анализе системы.

В IDEF0 система представляется как совокупность взаимодействующих работ (или функций). Связи между работами определяют технологический процесс или структуру взаимосвязи внутри организации. Модель SADT представляет собой серию диаграмм, разбивающих сложный объект на составные части.

Основными понятиями методологии функционального моделирования работ являются:

*Работы (activity)* - поименованные процессы, функции или задачи, которые происходят в течение определенного времени и имеют распознаваемые результаты. На диаграмме работы изображаются прямоугольниками.

*Вход (Input)* - материал или информация, которые используются работой для получения результата (стрелка, входящая в левую грань).

*Управление (Control)* - правила, стратегии, стандарты, которыми руководствуется работа (стрелка, входящая в верхнюю грань). В отличие от входной информации управление не подлежит изменению.

*Выход (Output)* - материал или информация, которые производятся работой (стрелка, исходящая из правой грани). Каждая работа должна иметь хотя бы одну стрелку выхода, так как работа без результата не имеет смысла и не должна моделироваться.

*Механизм (Mechanism)* - ресурсы, которые выполняют работу (персонал, станки, устройства - стрелка, входящая в нижнюю грань).

*Вызов (Call)* представляет собой взаимодействие одной модели работ с другой (стрелка, исходящая из нижней грани).

Различают в IDEF0 пять типов связей работ.

*Связь по входу (input-output)* имеет место, когда выход вышестоящей работы направляется на вход следующей работы.

*Связь по управлению (output-control)* обозначает ситуацию, когда выход вышестоящей работы направляется на управление следующей работы. Связь показывает доминирование вышестоящей работы.

*Обратная связь по входу (output-input feedback)* имеет место, когда выход нижестоящей работы направляется на вход вышестоящей. Используется для описания циклов.

*Обратная связь по управлению (output-control feedback)* обозначает ситуацию, когда выход нижестоящей работы направляется на управление вышестоящей. Является показателем эффективности бизнес-процесса.

*Связь выход-механизм (output-mechanism)* имеет место, когда выход одной работы направляется на механизм другой и показывает, что работа подготавливает ресурсы для проведения другой работы.

На рисунках 2.1, 2.2 представлены первый и второй уровни диаграммы работы ИС «Интернет-магазин компьютеров», описывающие принципы функционирования системы.

USED AT:

AUTHOR:

DATE:

REV:

PROJECT: Интернет магазин компьютеров

18.05.2010

18.05.2010

NOTES: 1 2 3 4 5 6 7 8 9 10

WORKING

DRAFT

RECOMMENDED

PUBLICATION

READER

DATE

CONTEXT:

TOP

NODE:

TITLE:

NUMBER:

Информационная система "Интернет-магазин

компьютеров"

A-0

0

0р.

Информационная система

"Интернет-магазин компьютеров"

Стандартная конфигурация

Компоненты конфигурации

Данные клиента

Вид оплаты

Электронное письмо

Состояние заказа

Доставленный ПК

Цены

Вид наценки

Комплектующие

Рисунок 2.1 – Первый уровень диаграммы работы ИС «Интернет-магазин компьютеров»

Рисунок 2.2 – Второй уровень диаграммы работы ИС «Интернет-магазин компьютеров»

1. Моделирование функциональных требований к БД

Для построения модели проектируемой ИС воспользуемся моделью потоков данных. Диаграммы потоков данных (Data Flow Diagrams - DFD) используются для описания движения документов и обработки информации как дополнение к IDEF0. В отличие от IDEF0, где система рассматривается как взаимосвязанные работы, стрелки в DFD показывают лишь то, как объекты (включая данные) движутся от одной работы к другой. DFD отражает функциональные зависимости значений, вычисляемых в системе, включая входные значения, выходные значения и внутренние хранилища данных. DFD - это граф, на котором показано движение значений данных от их источников через преобразующие их процессы к их потребителям в других объектах.

DFD содержит процессы, которые преобразуют данные, потоки данных, которые переносят данные, активные объекты, которые производят и потребляют данные, и хранилища данных, которые пассивно хранят данные [4].

Диаграмма потоков данных содержит:

* процессы, которые преобразуют данные;
* потоки данных, переносящие данные;
* активные объекты, которые производят и потребляют данные;
* хранилища данных, которые пассивно хранят данные.

На рисунке 3.1 изображена диаграмма потоков данных ИС «Интернет-магазин компьютеров».



Рисунок 3.1 – Диаграмма потоков данных ИС «Интернет-магазин компьютеров»

Основой логической модели статического представления моделируемой системы является диаграмма классов.

*Классы* – это базовые элементы любой объектно – ориентированной системы. Они представляют собой описание совокупностей однородных объектов с присущими им свойствами – атрибутами, операциями, отношениями и семантикой.

В рамках модели каждому классу присваивается уникальное имя, отличающее его от других классов. Если используется составное имя (в начале имени добавляется имя пакета, куда входит класс), то имя класса должно быть уникальным в пакете.

Сущность – это субъект, место, вещь, событие или понятие, содержа­щие информацию. Точнее, сущность – это набор объектов, называемых экземплярами. Каждый экземпляр сущности обладает набором характеристик.

Логические взаимосвязи представляют собой связи между сущностями. Они определяются глаголами, показывающими, как одна сущность отно­сится к другой [5].

Некоторые примеры взаимосвязей:

• команда включает много игроков;

• самолет перевозит много пассажиров;

• продавец продает много продуктов.

Во всех этих случаях взаимосвязи отражают взаимодействие между двумя сущностями, называемое «один – ко – многим». Это означает, что один экземпляр первой сущности взаимодействует с несколькими экземплярами другой сущности. Взаимосвязи отображаются линиями, соединяющими две сущности с точкой на одном конце и глаголом, располагаемым над линией.

Кроме взаимосвязи «один – ко – многим» существует еще один тип – это «многие – ко – многим». Этот тип связи описывает ситуацию, при которой эк­земпляры сущностей могут взаимодействовать с несколькими экземпля­рами других сущностей. Связь «многие – ко – многим» используют на перво­начальных стадиях проектирования. Этот тип взаимосвязи отображается сплошной линией с точками на обоих концах.

Классы представляются в виде таблиц. В каждой таблице определен первичный ключ – поле, однозначно идентифицирующее запись и облегчающее установление связи между таблицами. При создании сущности необходимо выделить группу атрибутов, которые потенциально могут стать первичным ключом (потенциальные ключи), затем произвести отбор атрибутов для включения в состав первичного ключа, следуя следующим рекомендациям:

* Первичный ключ должен быть подобран таким образом, чтобы по значениям атрибутов, в него включенных, можно было точно идентифицировать экземпляр сущности.
* Никакой из атрибутов первичного ключа не должен иметь нулевое значение.
* Значения атрибутов первичного ключа не должны меняться. Если значение изменилось, значит, это уже другой экземпляр сущности. [6]

При выборе первичного ключа можно внести в сущность дополнительный атрибут и сделать его ключом. Так, для определения первичного ключа часто используют уникальные номера, которые могут автоматически генерироваться системой при добавлении экземпляра сущности в БД. Применение уникальных номеров облегчает процесс индексации и поиска в БД.

В процессе построения логической модели системы первым этапом явля­ется определение сущностей и атрибутов.

Таблица 3.1 – Атрибуты сущности «Клиент»

|  |  |
| --- | --- |
| Атрибут | Описание |
| ID клиента | Уникальный номер для идентификации клиента |
| Ф.И.О. | Фамилия, имя и отчество клиента |
| Адрес | Адрес проживания, куда будет доставлен заказ |
| E-mail | Адреса электронной почты |
| ID заказа | Уникальный номер сделанного данным клиентом заказа |

Так как клиент формирует заказ из предложенного товара, определим сущность «Товар» или «Прайс-лист» (таблица 3.2).

Таблица 3.2 - Атрибуты сущности «Товар»

|  |  |
| --- | --- |
| Атрибут | Описание |
| ID товара | Уникальный номер для идентификации товара |
| Наименование | Полное название товара |
| Цена | Стоимость данного товара |
| Вид конфигурации | Стандартная или Комплектующие; если Стандартная:  Серверы/Настольные/Портативные/Карманные |
| Описание | Подробное описание и характеристики |

Определим сущность «Заказ» (таблица 3.3).

Таблица 3.3 - Атрибуты сущности «Заказ»

|  |  |
| --- | --- |
| Атрибут | Описание |
| ID заказа | Уникальный номер для идентификации заказа |
| Сумма заказа | Подсчитанная стоимость заказа |
| Оплачен | Да или Нет |
| ID товара | Уникальные номера товаров, составляющих данный заказ |
| Состояние заказа | Поставлен в очередь/Собран/Отправлен |
| Номер счёта | Для оплаты данного заказа |
| Вид оплаты | Курьеру/ Переводом на счёт |

Составляется ERD-диаграмма, определяя типы атрибутов и проставляя связи между сущностями (рисунок 3.2). Связь «Клиент» - «Заказ» - «один-к-одному», а «Заказ» - «Товар» - «один-ко-многим».

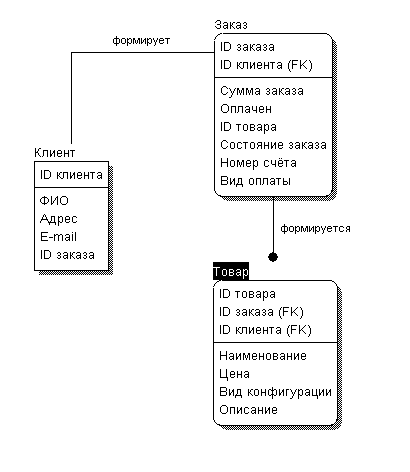


Рисунок 3.2 – ERD-диаграмма ИС «Интернет-магазин компьютеров»

Следующим этапом при построении логической модели является опре­деление ключевых атрибутов и типов атрибутов (таблица 3.4).

Таблица3. 4 - Типы атрибутов

|  |  |  |
| --- | --- | --- |
| Атрибут | Тип | |
| ID клиента | Number | |
| Ф.И.О. | String | |
| Адрес | String | |
| E-mail | String | |
| ID заказа | Number |
| ID товара | Number |
| Наименование | String |
| Цена | Number |
| Вид конфигурации | String |
| Описание | String |
| Сумма заказа | Number |
| Оплачен | Number |
| Состояние заказа | String |
| Номер счёта | Number |
| Вид оплаты | String |

Выберем для каждой сущности ключевые атрибуты, однозначно опре­деляющие сущность. Для сущности «Клиент» это будет ID клиента, для сущности «Товар» - ID товара, для сущности «Заказ» - ID заказа и номер счёта.

ЗАКЛЮЧЕНИЕ

Использование информационных систем делает любое производство более конкурентоспособным за счет повышения его управляемости и адаптируемости. Подобная автоматизация позволяет повысить эффективность управления за счет обеспечения руководителей и специалистов максимально полной, оперативной и достоверной информацией на основе единого банка данных.

В ходе выполнения данной курсовой работы была разработана информационная система «Интернет-магазин компьютеров», которая позволяет покупать компьютеры и комплектующие в сети Интернет.

В данной курсовой работе были построены диаграммы работы информационной системы «Интернет-магазин компьютеров» в целом и отдельных сценариев работ, отражающих логику и взаимоотношение подсистем, диаграмма потоков данных, описывающая движение информации в системе.

Следующим этапом было создание логической схемы, используя диаграммы классов с соответствующими атрибутами и взаимодействиями между классами.

Результаты проектирования являются основой для разработки информационной системы «Интернет-магазин компьютеров», которая может быть использована любой торговой компанией, имеющей выход в Интернет, а также любым клиентом, пожелавшим приобрести компьютер или его комплектующие.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Боггс, У. UML и Rational Rose: Пер. с англ. / У. Боггс, М. Боггс. - М.: ЛОРИ, 2000.
2. Смирнова, Г. Н., Проектирование экономических информационных систем: Учебник./ Г. Н. Смирнова, А. А. Сорокин, Ю. Ф. Тельнов – М.: Финансы и статистика, 2003. – 512 с.
3. Буч, Г. Язык UML. Руководство пользователя: Пер. с англ. / Г. Буч, Д.
4. Вендров, A. M., Проектирование программного обеспечения экономических информационных систем: Учебник./ А. М. Вендров – М.: Финансы и статистика, 2006. – 544 с.
5. Вендров, A. M., Практикум по проектированию программного обеспечения экономических информационных систем./ А. М. Вендров – М.: Финансы и статистика, 2006. – 191 с.
6. Смирнов, Н. В., Методические рекомендации по дисц. "Проектирование информационных систем"./ Н. В. Смирнов – СПб: БГТУ "Военмех", 2008. – 217 с.
7. Балдин, К. В., Информационные системы в экономике./ К. В. Балдин, В. Б. Уткин – М.: Издательский центр Академия, 2005 – 288 с.
8. Гайдамакин, Н. А., Автоматизированные информационные системы, базы и банки данных: Вводный курс./ Н. А. Гайдамакин, – М.: Гелиос АРВ, 2002. – 368с.
9. Абросимов, А. Г., Теория экономических информационных систем./ А. Г. Абросимов, М. А. Бородинова – Самара: Самарск. Гос. Экон.академ., 2001. – 170 с.
10. Маняшек, Г., Анализ требований и проектирование систем с использованием UML./ Г. Маняшек, А. Лешек – М.: Издательский дом «Вильямс», 2000. – 570 с.
11. Бутко, В. Р., CASE - технологии моделирования и проектирования АИС - Учебн. пособие./ В. Р. Бутко, В. П. Дерябкин – Самара: Самарск. Гос. Экон. академ., 2001. – 105 с.
12. Мюллер, Р., Базы данных и UML. Проектирование./ Р. Мюллер – М.: Издательский дом «Вильямс», 2000. – 590 с.
13. Глушаков, С. В., Базы данных./ С. В. Глушаков, Д. В. Ломотько – СПб.: Питер, 2000. – 387 с.
14. Грекул, В. И., Проектирование информационных систем. Курс лекций. Учебное пособий./ В. И. Грекул, Г. Н. Денищенко, Н. Л. Коровкина – М.: Интернет – Ун – т Информ. технологий, 2005. – 304 с.
15. Избачков, Ю. С., Информационные системы: Учебник для вузов. 2-е изд./ Ю. С. Избачков, В. Н. Петров – СПб.: Питер, 2006. – 656 с

Маклаков, С. В., Создание информационных систем с AllFusion Modeling Suite./ С. В. Маклаков – М.: Диалог-МИФИ, 2003. – 432 с.