Министерство образования Республики Беларусь

Учреждение образования «Институт информационных технологий Белорусского государственного университета информатики и радиоэлектроники»

Факультет компьютерных технологий

Кафедра информационных систем и технологий

Дисциплина: Базы данных

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

на тему

“Центр детского творчества”

БГУИР КП 1-40 01 01 10  ПЗ

Студент: гр. 68107042 Иванов К. Е.

Руководитель: Николаенко В.Л.

Минск 2018

**Оглавление**

[ВВЕДЕНИЕ 3](#_Toc516147417)

[1. ТЕОРЕТИЧЕСКИЙ РАЗДЕЛ 4](#_Toc516147418)

[2. ПРОЕКТНАЯ ЧАСТЬ 8](#_Toc516147419)

[2.1 Анализ предметной области 8](#_Toc516147420)

[Список литературы 20](#_Toc516147421)

### ВВЕДЕНИЕ

Данная курсовая работа выполнена в соответствии с заданием на курсовое проектирование. Темой работы является «Центр детского творчества».

В современном мире стремительными темпами развиваются информационные технологии и те сферы человеческой деятельности, которые с ними связаны. С каждым годом всё больший и больший объём информации обрабатывается при помощи современных и продолжающих совершенствоваться персональных компьютерах.

Любая организация нуждается в своевременном доступе к информации. Ценность информации в современном мире очень высока. Роль распорядителей информации в современном мире чаще всего выполняют базы данных. Базы данных обеспечивают надежное хранение информации, структурированном виде и своевременный доступ к ней. Практически любая современная организация нуждается в базе данных, удовлетворяющей те или иные потребности по хранению, управлению и администрированию данных.

Главная цель, которая стоит перед написанием курсового проекта – это приобретение практических навыков проектирования базы данных «Центр детского творчества» с использованием программ СУБД. В ходе выполнения данного курсового проекта необходимо будет разработать информационную базу данных и программное обеспечение для работы совместно с базой, данной для центра детского творчества, которые помогут пользователю легко найти нужную информацию о сотруднике, детях, группах и событиях в любом момент времени.

## 1. ТЕОРЕТИЧЕСКИЙ РАЗДЕЛ

База данных (БД) – это совокупность специальным образом организованных данных хранимых в памяти вычислительной системы отображающих состояние объектов и их взаимосвязи в рассматриваемой предметной области.

Предметной областью называется часть реальной системы, представляющая интерес для определенного исследования. Различать полную предметную область (предприятие) и организационную единицу этой предметной области (отделы).

Основные термины.

Объект – термин, обозначающий факт, лицо, событие, предмет, о котором могут быть собраны данные (каждая таблица представляет один объект);

Реляционная БД - основной тип современных баз данных. Состоит из таблиц, между которыми могут существовать связи по ключевым значениям.

Таблица базы данных (table) – регулярная структура, которая состоит из однотипных строк (записей, records), разбитых на столбцы (поля, fields).

В теории реляционных баз данных синоним таблицы – отношение (relation), в котором строка называется кортежем, а столбец называется атрибутом.

В концептуальной модели реляционной БД аналогом таблицы является сущность (entity), с определенным набором свойств - атрибутов, способных принимать определенные значения (набор допустимых значений - домен).

Ключевой элемент таблицы (ключ, regular key) - такое ее поле (простой ключ) или строковое выражение, образованное из значений нескольких полей (составной ключ), по которому можно определить значения других полей для одной или нескольких записей таблицы. На практике для использования ключей создаются индексы - служебная информация, содержащая упорядоченные сведения о ключевых значениях. В реляционной теории и концептуальной модели понятие "ключ" применяется для атрибутов отношения или сущности.

Первичный ключ (primary key) - главный ключевой элемент, однозначно идентифицирующий строку в таблице. Могут также существовать альтернативный (candidate key) и уникальный (unique key) ключи, служащие также для идентификации строк в таблице.

В реляционной теории первичный ключ - минимальный набор атрибутов, однозначно идентифицирующий кортеж в отношении.

В концептуальной модели первичный ключ - минимальный набор атрибутов сущности, однозначно идентифицирующий экземпляр сущности.

Связь (relation) - функциональная зависимость между объектами. В реляционных базах данных между таблицами устанавливаются связи по ключам, один из которых в главной (parent, родительской) таблице - первичный, второй - внешний ключ - во внешней (child, дочерней) таблице, как правило, первичным не является и образует связь "один ко многим" (1:N). В случае первичного внешнего ключа связь между таблицами имеет тип "один к одному" (1:1). Информация о связях сохраняется в базе данных.

Внешний ключ (foreign key) - ключевой элемент подчиненной (внешней , дочерней)

Для работы с БД используются системы управления базами данных (СУБД).

СУБД – комплекс языковых и программных средств, предназначенных для создания, ведения и совместного использования БД несколькими пользователями. СУБД позволяет: создавать БД; вставлять, обновлять, удалять и извлекать информацию из БД; предоставляет контролируемый доступ к базе данных.

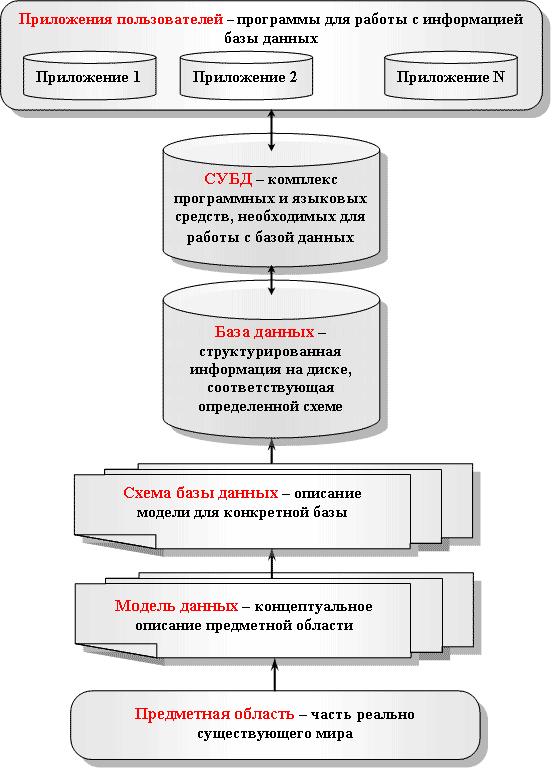


Рисунок 1. Взаимосвязь основных терминов в области проектирования баз данных и работы с ними

Процесс проектирования БД представляет собой процесс переходов от неформального словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели. Конечной целью проектирования является построение конкретной БД. Очевидно, что процесс проектирования сложен и поэтому имеет смысл разделить его на логически завершенные части – этапы.

Можно выделить пять основных этапов проектирования БД:

1. Сбор сведений и системный анализ предметной области - это первый и важнейший этап при проектировании БД. В нем необходимо провести подробное словесное описание объектов предметной области и реальных связей, присутствующих между реальными объектами.

2. Инфологическое проектирование - частично формализованное описание объектов предметной области в терминах некоторой семантической модели.

3. Выбор СУБД осуществляется на основе различных требований к БД и, соответственно, возможностей СУБД, а также в зависимости от имеющегося опыта разработчиков.

4. Даталогическое проектирование – это описание БД в терминах принятой даталогической модели данных. В реляционных БД даталогическое или логическое проектирование приводит к разработке схемы БД, т.е. совокупности схем отношений, которые адекватно моделируют объекты предметной области и семантические связи между объектами. Основой анализа корректности схемы являются функциональные зависимости между атрибутами БД.

5. Физическое проектирование заключается в увязке логической структуры БД и физической среды хранения с целью наиболее эффективного размещения данных, т.е. отображение логической структуры БД в структуру хранения.

## 2. ПРОЕКТНАЯ ЧАСТЬ

### 2.1 Анализ предметной области

На сегодняшний день существует большое количество программ, разработанных специально для детских развивающих центров, детских творческих клубов, центров раннего развития и предприятий подобной направленности. Они все очень схожи между собой, и имеют различные преимущества относительно друг друга. Для примера возьмём две программы:

«Хеликс: Детский Центр» - компьютерная программа, разработанная специально для детских развивающих центров, которая поможет автоматизировать все рутинные процессы по учету и управлению бизнесом.

"Хеликс: Детский Центр" разработан на платформе "1С:Предприятие 8.3". Из плюсов программы можно выделить то что она максимально простая и удобная в работе, а также в ней реализованы множество функциональных возможностей. Из минусов можно отметить, что «Хеликс: Детский Центр» является платной программой, для её установки требуется выезд специалиста, и гарантия на обслуживание составляет всего 3 месяца.

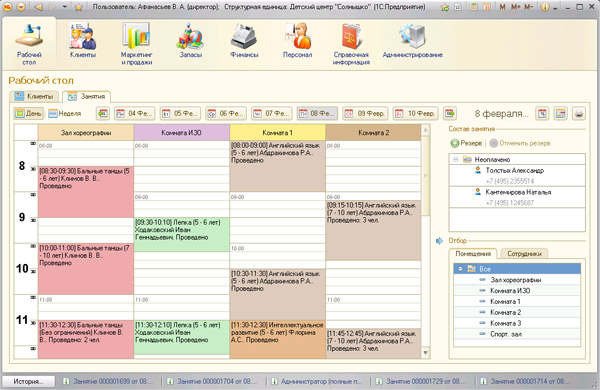


Рисунок 2.1.1 – Хеликс: Детский Центр»

«CRM-система для автоматизации детских центров Отмечалка» - это система автоматизации, предоставляющая полный набор инструментов, которые помогут раскрыть потенциал детского центра, повысить его прибыльность и конкурентоспособность.

Из плюсов это программы можно отметить то что в ней есть бесплатный 14 дневный период, постоянное обновление программы и высокая надёжность хранения данных.

В минусы этой программы также входят можно записать необходимость платить за её использование, а также необходимость подключения к сети интернет.

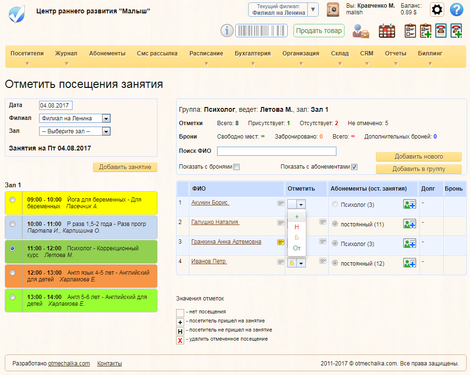


Рисунок 2.1.2 – «CRM-система Отмечалка»

2.2 Инфологическое проектирование

Основными задачами этапа инфологического проектирования являются определение предметной области системы и формирование взгляда на неё с позиций сообщества будущих пользователей БД, т.е. информационно-логической модели.

Инфологическая модель представляет собой описание структуры и динамики, характера информационных потребностей пользователей в терминах, понятных пользователю и не зависимых от реализации БД. Это описание выражается в терминах не отдельных объектов и связей между ними, а их типов, связанных с ними ограничений целостности и тех процессов, которые приводят к переходу из одного состояния в другое.

Основными подходами к созданию инфологической модели предметной области являются:

1. Функциональный подход к проектированию БД ("от задач").

2. Предметный подход к проектированию БД ("от предметной области").

3. Метод "сущность-связь" (entity-relation, ER-method).

Мы будем использовать метод "сущность-связь" как наиболее распространённый. Приведём основные термины, которыми мы будем пользоваться:

Сущность - это объект, о котором в системе будут накапливаться данные. Для сущности указывается название и тип (сильная или слабая). Сильные сущности существуют сами по себе, а существование слабых сущностей зависит от существования сильных.

Атрибут - свойство сущности. Различают:

1) Идентифицирующие и описательные атрибуты. Идентифицирующие

позволяют отличить один экземпляр сущности от другого. Описательные атрибуты заключают в себе интересующие нас свойства сущности.

2) Составные и простые атрибуты. Простой атрибут имеет неделимое значение. Составной атрибут является комбинацией нескольких элементов, возможно, принадлежащих разным типам данных (ФИО, адрес и др.).

3) Однозначные и многозначные атрибуты (могут иметь соответственно одно или много значений для каждого экземпляра сущности). Например, дата рождения - это однозначный атрибут, а номер телефона - многозначный.

4) Основные и производные атрибуты. Значение основного атрибута не зависит от других атрибутов; значение производного атрибута вычисляется на основе значений других атрибутов. Например, возраст вычисляется на основе даты рождения и текущей даты.

5) Обязательные и необязательные (первые должны быть указаны при размещении данных в БД, вторые могут не указываться). Для каждого атрибута необходимо определить название, указать тип данных и описать ограничения целостности - множество значений, которые может принимать данный атрибут.

Связь - это осмысленная ассоциация между сущностями. Для связи указывается название, тип (факультативная или обязательная), кардинальность (1:1, 1:n или m:n) и степень (унарная, бинарная, тернарная или n-арная). Инфологическая модель предметной области «Ломбард» представлена на рис. 2.2.1.

Рисунок 2.2.1 - Инфологическая (концептуальная) модель предметной области

2.3 Обоснование выбора программных средств

При разработке программного средства на компьютере была установлена операционная система Microsoft Windows 10 Pro. Данная операционная система проста в своем использовании и проста как среда для разработки программных проектов.

В ОС Windows 10 программы разрабатывается без всевозможных проблем и полностью совместима с программной средой C++.

Операционная система (ОС) – комплекс программ, которые осуществляют контроль и управление всеми устройствами компьютера, осуществляют загрузку и запуск пользовательских программ, организуют взаимодействие узлов компьютера и предоставляют пользователю возможность управлять вычислительным процессом.

Анализ развития и использования операционных систем и сред показывает, что как отдельные пользователи персональных компьютеров (ПК), так и пользователи компьютерных сетей предпочитают работу на Windows.

Операционная среда Windows подготовлена таким образом, что обращение к конкретным типам аппаратуры она взяла на себя, то есть среда полностью независима от аппаратуры. Она обладает свойством аппаратной и программной совместимости, в ней реализована возможность работы со всеми прикладными программами DOS (текстовыми редакторами, электронными таблицами и так далее). Операционная система использует возможности компьютера и периферийных устройств в полном объеме. Таким образом, если на вашем компьютере устанавливается новое устройство (скажем, новый тип принтера), то все программы, которые выполняются под управлением Windows, получают к нему доступ.

Windows позволяет запускать одновременно большое количество программ, что дает возможность осуществить более полную загрузку всех компонентов системы, обеспечивая механизм быстрого переключения с одной задачи на другую. В отличие от DOS система Windows не загружает программу в память, а «подкачивает» ее фрагменты по мере необходимости. Это позволяет менее расточительно использовать память. Кроме того, заложенный в Windows механизм предоставляет возможность пользователю распоряжаться всей расширенной памятью компьютера, в то время как в MS DOS это возможно лишь через виртуальный диск или мало распространенные специальные библиотеки, механизм функционирования которых скрыт от пользователя. Но все это лишь средства временного и локального решения вопросов.

Наличие в Windows стандартного интерфейса с пользователем позволяет, изучив работу одной программы, без особых усилий осваивать все последующие. В Windows унифицирован интерфейс всех программ и правила работы с ними.

Данный курсовой проект разрабатывался в системе программирования Rad Studio XE10, располагающий широкими возможностями, а также снабжённый необходимым набором драйверов для доступа к самым известным форматам данных, удобными и развитыми средствами для доступа к информации и естественно коллекцией визуальных компонентов для построения отображаемых на экране окон, что так необходимо для создания удобного интерфейса.

C++ – компилируемый, статически типизированный язык программирования общего назначения.

Поддерживает такие парадигмы программирования как процедурное программирование, объектно-ориентированное программирование, обобщённое программирование, обеспечивает модульность, раздельную компиляцию, обработку исключений, абстракцию данных, объявление типов (классов) объектов, виртуальные функции. Стандартная библиотека включает, в том числе, общеупотребительные контейнеры и алгоритмы. C++ сочетает свойства как высокоуровневых, так и низкоуровневых языков. В сравнении с его предшественником – языком C, – наибольшее внимание уделено поддержке объектно-ориентированного и обобщённого программирования.

C++ широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем, высокопроизводительных серверов, а также развлекательных приложений (игр). Существует множество реализаций языка C++, как бесплатных, так и коммерческих и для различных платформ. Например, на платформе x86 это GCC, Visual C++, Intel C++ Compiler, Embarcadero (Borland) C++ Builder и другие. C++ оказал огромное влияние на другие языки программирования, в первую очередь на Java и C#.

Синтаксис C++ унаследован от языка C. Одним из принципов разработки было сохранение совместимости с C. Тем не менее, C++ не является в строгом смысле надмножеством C; множество программ, которые могут одинаково успешно транслироваться как компиляторами C, так и компиляторами C++, довольно велико, но не включает все возможные программы на C.

SQLite - компактная встраиваемая реляционная база данных. «Встраиваемая» означает, что SQLite не использует парадигму клиент-сервер, то есть движок SQLite не является отдельно работающим процессом, с которым взаимодействует программа, а предоставляет библиотеку, с которой программа компонуется и движок становится составной частью программы.

Сама библиотека SQLite написана на C,но существует большое количество привязок к другим языкам программирования, в том числе PHP.

2.4 Даталогическое проектирование

Даталогическое проектирование - создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных. Для реляционной модели данных даталогическая модель — набор схем отношений, обычно с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

Преобразование концептуальной модели в логическую модель, как правило, осуществляется по формальным правилам. Этот этап может быть в значительной степени автоматизирован.

На этапе логического проектирования учитывается специфика конкретной модели данных, но может не учитываться специфика конкретной СУБД.

На этапе логического проектирования разрабатывается логическая (концептуальная) структура БД. Для реляционной модели существуют формальные правила, которые позволяют преобразовать инфологическую модель в виде ER-диаграммы в логическую схему базы данных. Кроме получения схемы БД в целом на этом этапе выполняют создание схем отношений и их нормализацию.

Шесть сущностей находятся в третьей нормальной форме, т. е. все атрибуты сущностей содержат атомарные значения (значения в домене не являются ни списками, ни множествами простых или сложных значений); каждый неключевой атрибут полностью зависит от первичного ключа; никакой неключевой атрибут не зависит от другого неключевого атрибута. Логическая структура предметной области «Ломбард» представлена на рис. 2.4.1.

Рисунок 2.4.1 - Логическая структура базы данных

2.5 Физическое проектирование

Этап физического проектирования заключается в определении схемы хранения, т.е. физической структуры БД. Схема хранения зависит от той физической структуры, которую поддерживает выбранная СУБД. Физическая структура БД, с одной стороны, должна адекватно отражать логическую структуру БД, а с другой стороны, должна обеспечивать эффективное размещение данных и быстрый доступ к ним. Результаты этого этапа документируются в форме схемы хранения на языке определения данных (DDL, Data Definition Language) выбранной СУБД. Принятые на этом этапе решения оказывают огромное влияние на производительность системы.

Одной из важнейших составляющих проекта базы данных является разработка средств защиты БД. Защита данных имеет два аспекта: защита от сбоев и защита от несанкционированного доступа. Для защиты от сбоев на этапе физического проектирования разрабатывается стратегия резервного копирования.

Для защиты от несанкционированного доступа каждому пользователю доступ к данным предоставляется только в соответствии с его правами доступа, набор которых также является составной частью проекта БД.

Каждое реляционное отношение соответствует одной сущности и в него вносятся все атрибуты этой сущности. Для каждого отношения определяются первичный ключ и внешние ключи (в соответствии со схемой БД). В том случае, если базовое отношение не имеет потенциальных ключей, вводится суррогатный первичный ключ, который не несёт смысловой нагрузки и служит только для идентификации записей.

Отношения приведены в табл. 1-6. Для каждого отношения указаны атрибуты с их внутренним названием, типом и длиной.

Таблица 1

«Client»

Атрибуты Тип Ключ Описание

id Integer primary Код клиента

fname Varchar(50) Имя

sname Varchar(50) Фамилия

name Varchar(50) Отчество

pnum Varchar(50) Серия и номер паспорта

bdate Varchar(50) Дата рождения

loc Varchar(50) Адрес

phone Varchar(50) Номер телефона

Таблица 2

«Contr»

Атрибуты Тип Ключ Описание

Id Integer primary Код договора

datei Varchar(50) Дата принятия

subj Varchar(50) Название предмета

indet Varchar(50) Модель предмета

fname Varchar(50) Имя

sname Varchar(50) Фамилия

name Varchar(50) Отчество

pnum Varchar(50) Номер паспорта

bdate Varchar(50) Дата рождения

loc Varchar(50) Адрес

phone Varchar(50) Номер телефона

pricei Integer Цена принятия

priceo Integer Цена выкупа

dateo Varchar(50) Дата выкупа

cid Integer Код клиента

eid Integer Код сотрудника

sid Integer Код предмета

Таблица 3

«Employee»

Атрибуты Тип Ключ Описание

id Integer primary Код работника

fname Varchar(50) Имя

sname Varchar(50) Фамилия

name Varchar(50) Отчество

bdate Varchar(50) Дата рождения

position Varchar(50) Должность

Таблица 4

«Idet»

Атрибуты Тип Ключ Описание

id Integer primary Код модели

sid Integer Код предмета

name Varchar(50) Название модели

Таблица 5

«Subject»

Атрибуты Тип Ключ Описание

id Integer primary Код предмета

name Varchar(50) Имя предмета

Таблица 6

«Subjfsale»

Атрибуты Тип Ключ Описание

id Integer primary Код предмета

subj Varchar(50) Имя предмета

idet Varchar(50) Модель

price Integer Цена

sid Integer Код модели

ЗАКЛЮЧЕНИЕ

Результатом выполнения курсовой работы стало разработанное приложение баз данных, позволяющее автоматизировать операции учета и процессы составления отчетных документов в ломбарде. Разработанное приложение отвечает всем требованиям предметной области, таблицы созданной базы данных отвечают требованиям нормализации, что позволяет обеспечить целостность и непротиворечивость информации. В данной работе разработана база данных, реализующая базу данных материально-ответственного лица организации.

В процессе выполнения курсовой работы были закреплены знания, полученные при изучении дисциплины. Были изучены такие пункты:

 анализ предметной области;

 построение концептуальной модели базы данных;

 организация базы данных;

 разработка прикладной программы;

 наполнение и сопровождение базы данных;

В процессе организации БД проведен до необходимого уровня абстракций анализ предметной области, построена реляционная модель БД, произведена нормализация реляционной БД. Реализация проекта была выполнена на современных программных платформах. В качестве технологии доступа к данным была использована объектно-реляционная модель, которая позволяет просто и лаконично осуществлять запросы к базе данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРИЛОЖЕНИЯ