**Министерство образования и науки Российской Федерации**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**Московский государственный университет технологии и управления имени К.Г. Разумовского (Первый казачий университет)**

**Университетский колледж информационных технологий**

Специальность: 09.02.03 «Программирование в компьютерных системах»

**КУРСОВОЙ ПРОЕКТ**

По модулю ПМ.02 «Разработка и администрирование баз данных»

МДК.02.02 «Технология разработки и защиты баз данных»

на тему «Генератор экзаменационных билетов»

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**УКИТ 09.02.03.2016\_303.015ПЗ**

Группа \_\_\_*П-303\_\_\_*

Студент \_\_\_\_\_\_\_\_\_\_\_ Ильин П.М.

(*личная подпись*)

Руководители проекта \_\_\_\_\_\_\_\_\_\_\_ Ларионова Е.А.

(*личная подпись*)

**МОСКВА 2016**

Содержание

**ВВЕДЕНИЕ**

Курсовой проект посвящен разработке прикладной программы, которая позволит резервировать места в гостиницах.

Приложение состоит из нескольких частей. Части, отвечающей за заполнение базы данных информацией о гостиницах и комнатах в них, и части, отвечающей за расселение постояльцев, посредством выбора подходящей гостиницы, комнаты и заполнения анкеты.

Целью курсового проекта является приобретение навыков разработки локальных и удалённых баз данных и приложений для работы с ними с использованием объектно-ориентированного и визуального программирования.

Задачи курсового проекта:

* Анализ предметной области;
* Проектирование базы данных средствами ErwinModeler и MSSQLServer 2008 R2
* Разработка приложения средствами Lazarus (Delphi);
* Защита базы данных и приложения должна осуществляется логином и паролем администратора или пользователя.

Возможности приложения:

* Добавление, изменение данных о гостиницах;
* Добавление, изменение данных о комнатах в гостиницах;
* Резервирование мест комнат в гостиницах.

1. **ОБЩАЯ ЧАСТЬ**

**1.1. Анализ предметной области.**

Приложение позволяет пользователю быстро и просто предоставить человеку информацию о свободных местах в гостиницах и так же легко зарезервировать для него комнату на определённый срок.

Приложение состоит из нескольких частей. Части, отвечающей за заполнение базы данных информацией о гостиницах и комнатах в них, и части, отвечающей за расселение постояльцев, посредством выбора подходящей гостиницы, комнаты и заполнения анкеты.

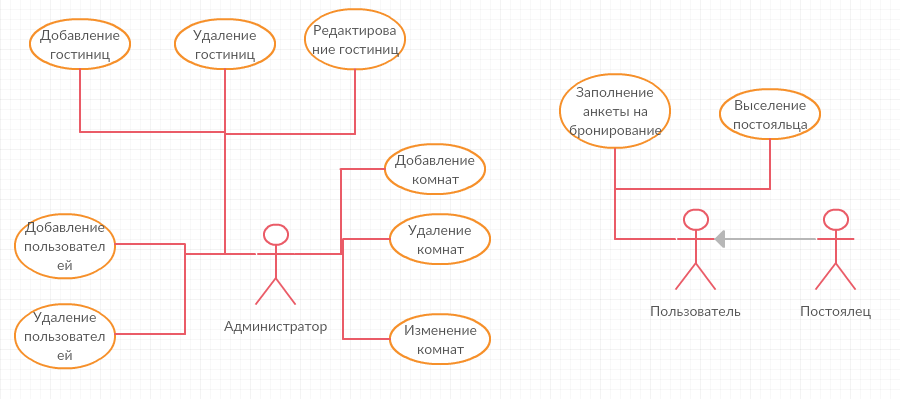
Выделим базовые сущности этой предметной области:

**Гостиницы** – Атрибуты – Номер гостиницы, название, адрес.

**Комнаты** – Атрибуты – Номер гостиницы, номер комнаты, статус, класс.

**Постояльцы** – Атрибуты – Паспорт, фамилия, имя, отчество, номер телефона, длительность резервирования.

**Журнал** – Атрибуты – Паспорт, номер гостиницы, номер комнаты, дата заселения, дата выселения.

Далее следует ER - диаграмма приложения "Система резервирования мест в гостиницах"

**1.2. Жизненный цикл базы данных.**

Жизненный цикл БД- это процесс проектирования, реализации и управления БД. Жизненный цикл БД состоит из 7-ми этапов:

1) Планирование разработки

2) Проверка осуществимости

3) Определение требований

4) Концептуальное проектирование

5) Логическое проектирование

6) Физическое проектирование

7) Оценка работы и поддержка базы данных

1. Предварительное планирование базы данных – важный этап в процессе перехода от разрозненных данных к интегрированным. На этом этапе собирается информация об используемых и находящихся в процессе разработки прикладных программ и файлах, связанных с ними. Она помогает установить связи между текущими приложениями и то, как используется их информация. Кроме того, позволяет определить будущее требования в БД. Информация документируется в виде обобщенной концептуальной модели данных.

2. Проверка осуществимости предполагает подготовку отчетов по трем вопросам:

1) Есть ли технология – необходимое оборудование и программное обеспечение - для реализации запланированной БД (технологическая осуществимость)

2) Имеется ли персонал, средства и эксперты для успешного осуществления плана создания БД (операционная осуществимость)

3) Окупится ли запланированная БД (экономическая эффективность)

3. Определение требований. На этом этапе определяются:

* Цели базы данных
* Информационные потребности различных структурных подразделений и их руководителей
* Требование к оборудованию
* Требование к программному обеспечению

4. Концептуальное проектирование. На этом этапе создаются подробные модели пользовательских преставлений данных предметной области. Затем они интегрируются в концептуальную модель, которая фиксирует все элементы корпоративных данных, подлежащих загрузке в базу данных. Эту модель еще называют концептуальной схемой базы данных.

5. Логическое проектирование. На этом этапе осуществляется выбор типа модели данных. Концептуальная модель отображается в логическую модель, основанную на структурах, характерных для выбора модели.

6. Физическое проектирование. На этом этапе логическая модель расширяется характеристиками, необходимыми для определения способов физического хранения базы данных, типа устройств для хранения, методов доступа к данным базы, требуемого объема памяти, правил сопровождения базы данных и др.

7. Оценка и поддержка базы данных. Оценка включает опрос пользователей на предмет выяснения, какие их информационные потребности остались неучтенными. При необходимости в спроектированную БД вносятся изменения. Пользователи обучаются работе с БД. По мере расширения и изменения потребностей бизнеса поддержка БД обеспечивается путем внесения изменений, добавления новых данных, разработки новых прикладных программ, работающих с БД.

Сроки разработки и сдачи курсового проекта 29.03.16- 23.05.16.

Начало работы 14.03.16-30.05.16.

**1.3. Защита базы данных.**

Необходимость защиты баз данных заключается в том, чтобы не произошла утечка данных.

Методы защиты БД:

* Защита паролем;
* Шифрование данных и программ;
* Разграничение прав доступа к объектам БД;
* Защита полей и записей таблиц БД.
* Защита от SQL инъекций (расписать)

Защита паролем представляет собой простой и эффективный способ защиты БД от несанкционированного доступа. Пароли устанавливаются пользователями или администраторами БД. Учет и хранение паролей выполняется самой СУБД. Парольная защита БД считается слабым средством, особенно если пароль не шифруется. Основной ее недостаток состоит в том, что все пользователи, использующие одинаковый пароль, с точки зрения вычислительной системы неразличимы. Неудобство парольной защиты для пользователя состоит в том, что пароль надо запоминать или записать. При небрежном отношении к записям пароль может стать достоянием других.

Более мощным средством защиты данных от просмотра является их шифрование. Шифрование – это преобразование читаемого текста в нечитаемый текст, при помощи некоторого алгоритма; применяется для защиты уязвимых данных.

Процесс дешифрования восстанавливает данные в исходное состояние.

В целях контроля использования основных ресурсов СУБД во многих системах имеются средства установления прав доступа к объектам БД. Права доступа определяют возможные действия над объектами. Владелец объекта (пользователь, создавший объект), а также администратор БД имеют все права. Остальные пользователи к разным объектам могут иметь различные уровни доступа. Разрешение на доступ к конкретным объектам базы данных сохраняется в файле рабочей группы.

Файл рабочей группы содержит данные о пользователях группы и считывается во время запуска. Файл содержит следующую информацию: имена учетных записей пользователей, пароли пользователей, имена групп, в которые входят пользователи.

По отношению к таблицам могут предусматриваться следующие права доступа:

* просмотр (чтение) данных;
* изменение (редактирование) данных;
* добавление данных;

К данным, имеющимся в таблице, могут применяться меры защиты по отношению к отдельным полям и отдельным записям.

Защита данных в полях таблиц предусматривает следующие уровни прав доступа:

* полный запрет доступа;
* только чтение;
* разрешение всех операций (просмотр, ввод новых значений, удаление и изменение).

Для предотвращения SQL инъекций следует соблюдать два простых правила:

1. Не помещать в БД данные без обработки.

Это можно сделать либо с помощью подготовленных выражений, либо обрабатывая параметры вручную.

Если запрос составляется вручную, то

- все числовые параметры должны быть приведены к нужному типу

2. Не помещать в запрос управляющие структуры и идентификаторы, введенные пользователем.

В данном курсовом проекта защита информационной системы сделана с помощью логина и пароля для администратора и осуществление входа пользователя с помощью ввода соответственного пароля.

**1.4. Выбор и характеристика СУБД.**

В качестве СУБД для проекта выступает Microsoft SQL Server 2008 R2. Выбор обусловлен наличием опыта работы в данной СУБД и кодирования на языке SQL, а также удобством работы с ней в связке в выбранной IDE. Кроме того, СУБД существует в бесплатной редакции Express, а по студенческой подписке в сервисе Microsoft Azure можно бесплатно развернуть облачное хранилище на основе Microsoft SQL Server (Azure SQL). Из минусов СУБД следует отметить её низкую производительность на слабых машинах, а также отсутствие возможности работы на ОС, отличной от Windows.

**1.5. Выбор и характеристика среды разработки приложения.**

Lazarus – это открытая среда для разработки программного обеспечения на языке ObjectPascal для компилятора FreePascal. Интегрированная среда разработки предоставляет возможность кроссплатформенной разработки приложений в Delphi-подобном окружении. Позволяет не сложно переносить Delphi-программы с графическим интерфейсом в различные операционные системы.

Основан на библиотеке визуальных компонентов Lazarus Component Library (LCL). В настоящее время практически полностью поддерживает виджеты Win32, GTK1, GTK2, Carbon, Qt. В разработке находятся виджеты WinCE.

* Поддерживает преобразование проектов Delphi
* Реализован основной набор элементов управления
* Редактор форм и инспектор объектов максимально приближены к Delphi
* Интерфейс отладки (используется внешний отладчик GDB)
* Простой переход для Delphi программистов благодаря близости LCL к VCL
* Полностью юникодный (UTF-8) интерфейс и редактор и поэтому отсутствие проблем с портированием кода, содержащего национальные символы
* Мощный редактор кода, включающий систему подсказок, гипертекстовую навигацию по исходным текстам, автозавершение кода и рефакторинг
* Форматирование кода «из коробки», используя механизмы Jedi Code Format
* Поддержка двух стилей ассемблера: Intel и AT&T (поддерживаются со стороны компилятора)
* Поддержка множества типов синтаксиса Pascal: Object Pascal, Turbo Pascal, Mac Pascal, Delphi (поддерживаются со стороны компилятора)
* Имеет собственный формат управления пакетами
* Авто сборка самого себя (под новую библиотеку виджетов) нажатием одной кнопки
* Поддерживаемые для компиляции ОС: Linux, Microsoft Windows (Win32, Win64), Mac OS X, FreeBSD, WinCE, OS/2

# 2. СПЕЦИАЛЬНАЯ ЧАСТЬ

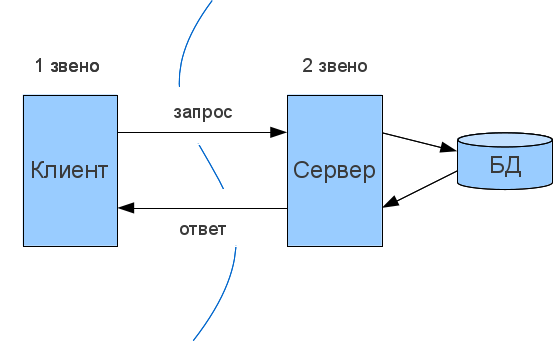
## **2.1. Постановка задачи.**

Спроектировать средствами MS SQL SERVER 2008 R2 базу данных и создать приложение, позволяющее добавлять гостиницы и комнаты в базу, резервировать комнаты в гостиницах, заполнять анкету постояльца, вести журнал по заселению, выселению постояльцев.

Предусмотреть возможность добавления, удаления, редактирования записей. Обеспечить защиту базы данных и приложения.

**2.2. Архитектура информационной системы.**

Архитектура «клиент-сервер» - вычислительная или сетевая архитектура, в которой задания или сетевая нагрузка распределены между поставщиком услуг, называемыми серверами, и заказчиками услуг, называемыми клиентами.



|  |  |
| --- | --- |
| **Преимущества** | **Недостатки** |
| Одновременный доступ | Высокая стоимость |
| Высокая производительность | Высокая нагрузка на сервер |
| Низкий трафик |  |
| Высокая защита |  |
| Высокая роль сервера |  |

**2.3. Логическая модель БД.**

Для построения логической модели БД используется средства автоматизации проектирования ErwinModeler.

//скрин ервина