

Оглавление

Введение	2
1. Инфологическое проектирование реляционной базы данных «Городская служба «03»	4
1.2. Концептуальная модель предметной области	4
1.2. Информационно-логическое проектирование БД «Городская служба «03»	8
2. Разработка структуры БД «Городская служба «03»	13
2.1. Таблицы БД «Городская служба «03» и связи между ними	14
2.2. Запросы БД «Городская служба «03»	17
2.3. Формы БД «Городская служба «03»	19
3. Криптографическая защита данных БД «Городская служба «03»	26
Заключение	30
Список использованной литературы	31

Введение

В настоящее время работа по внедрению в медицинскую деятельность современной техники и информационных технологий приобретает особое значение. Использование электронных вычислительных машин (ЭВМ) дает возможность сбора, хранения и анализа данных непосредственно на рабочем месте медицинского работника, что непосредственно сказывается на ускорении обработки данных, улучшении оформления документации, повышении производительности труда работников здравоохранения.

Настоящая курсовая работа посвящена проблеме разработки базы данных для службы скорой медицинской помощи и ее криптографической защите в среде MS ACCESS.

Целью данной курсовой работы является информационная поддержка работы городской службы «03» с реализацией защиты базы данных (БД) пациентов, обратившихся в медицинскую службу скорой помощи.

Основные задачи исследования:

- знакомство с предметной областью исследования;
- разработка информационно-логической модели (ИЛМ) данных;
- ведение базы данных пациентов службы скорой помощи (ввод, редактирование, поиск данных по заданным параметрам);
- обеспечение разделения прав доступа к базе данных для администратора БД и дежурных врачей городской службы «03».

В процессе разработки и создания базы данных использовались методы системного анализа и проектирования программного обеспечения.

Актуальность работы: появление на свет автоматизированных систем медицинской статистики является результатом двух тенденций: одна диктуется логикой развития самого здравоохранения, другая – независимым развитием информационных технологий. Новые информационные технологии сделали возможным и доступным автоматизированный сбор медицинских данных, который заменил ручной рутинный сбор и хранение большого объема данных о пациенте для ведения документации, статистики и анализа.

Наличие автоматизированной базы данных пациентов городской службы «03» позволяет все результаты лечения записывать в базу данных, что дает возможность врачу оперативно просмотреть статистику о принятых вызовах, оказания медицинской помощи по этим вызовам и результатам анамнеза больных до обращения в службу «03» и после оказания медицинской помощи врачами службы скорой медицинской помощи. Учет времени приема вызова и времени оказания медицинской услуги позволяет администрации службы «03» принять оперативные меры по улучшению работы городской службы «03».

1. Инфологическое проектирование реляционной базы данных «Городская служба «03»

1.2. Концептуальная модель предметной области

Концептуальная модель – это определённое множество понятий и связей между ними, являющихся смысловой структурой рассматриваемой предметной области.

Прежде, чем приступать к созданию системы автоматизированной обработки информации, разработчик должен сформировать понятия о предметах, фактах и событиях, которыми будет оперировать данная система. Для того, чтобы привести эти понятия к той или иной модели данных, необходимо заменить их информационными представлениями. Одним из наиболее удобных инструментов унифицированного представления данных, независимого от реализующего его программного обеспечения, является модель «сущность-связь» (entity – relationship model, ER – model).

Модель «сущность-связь» основывается на некой важной семантической информации о реальном мире и предназначена для логического представления данных. Она определяет значения данных в контексте их взаимосвязи с другими данными. Важным для нас является тот факт, что из модели «сущность-связь» могут быть порождены все существующие модели данных (иерархическая, сетевая, реляционная, объектная), поэтому она является наиболее общей.

Любой фрагмент предметной области может быть представлен как множество сущностей, между которыми существует некоторое множество связей. Дадим определения [3]:

Сущность (entity) – это объект, который может быть идентифицирован неким способом, отличающим его от других объектов. Примеры: врач, группа пользователей, вызов.

Набор сущностей (entity set) – множество сущностей одного типа (обладающих одинаковыми свойствами). Примеры: все врачи, группы пользова-

телей и т.д. Наборы сущностей не обязательно должны быть непересекающимися.

Множество значений (область определения) атрибута называется доменом.

Ключ сущности – группа атрибутов, такая, что отображение набора сущностей в соответствующую группу наборов значений является взаимно-однозначным отображением. Другими словами: ключ сущности – это один или более атрибутов, уникально определяющих данную сущность. В нашем примере ключом сущности *Врач* является уникальный атрибут *Код врача*.

Выделим базовые сущности рассматриваемой предметной области:

- Группы пользователей. Атрибуты: код группы, наименование группы, пароль.
- Врачи. Атрибуты: код врача; ФИО врача, адрес проживания врача, телефон врача.
- Диагнозы. Атрибуты: код заболевания по МКБ-10¹, описание заболевания.
- Вызовы (карта больного). Атрибуты: код вызова, код врача, дата и время вызова, ФИО пациента, дата рождения пациента, адрес пациента, жалобы пациента, дата и время оказания медицинской помощи, диагноз по МКБ-10, номер страхового полиса пациента.
- Анамнез пациента (продолжение карты больного). Атрибуты: код вызова, артериальное давление до оказания помощи, артериальное давление после оказания помощи, пульс до оказания медицинской помощи, пульс после оказания помощи, частота дыхания до оказания медицинской помощи, частота дыхания после оказания помощи, температура до оказания помощи, температура после оказания медицинской помощи.

Атрибуты сущностей содержат информацию, достаточную для решения поставленных задач.

¹ МКБ-10 – Международная классификация болезней 10-го пересмотра. Является общепринятой классификацией для кодирования медицинских диагнозов, разработана Всемирной организацией здравоохранения. МКБ-10 состоит из 21-го раздела, каждый из которых содержит подразделы с кодами заболеваний и состояний.

Связь (relationship) – это ассоциация, установленная между несколькими сущностями. Например, поскольку помощь по вызову оказывает дежурный врач скорой помощи, то между сущностями *Врач* и *Вызовы* существует связь «оказывает медицинские услуги» или другими словами *Врач–Вызовы*.

Набор связей (relationship set) – это отношение между n (причем n не меньше 2) сущностями, каждая из которых относится к некоторому набору сущностей.

В случае $n=2$, т.е. когда связь объединяет две сущности, она называется бинарной. Доказано, что n -арный набор связей ($n>2$) всегда можно заменить множеством бинарных, однако первые лучше отображают семантику предметной области.

То число сущностей, которое может быть ассоциировано через набор связей с другой сущностью, называют степенью связи. Рассмотрение степеней особенно полезно для бинарных связей. Могут существовать следующие степени бинарных связей [2]:

- один к одному (обозначается 1:1). Это означает, что в такой связи сущности с одной ролью всегда соответствует не более одной сущности с другой ролью;

- один ко многим (1:n). В данном случае сущности с одной ролью может соответствовать любое число сущностей с другой ролью. В рассмотренном выше примере это связь «оказывает медицинские услуги», поскольку один дежурный врач может оказать много медицинских услуг;

- многие ко многим (n:n). В этом случае каждая из ассоциированных сущностей может быть представлена любым количеством экземпляров.

Для доступа к реализуемой системе необходимо предоставить различные права пользования системой: администратор БД должен иметь полный доступ к средствам разработки базы данных, а дежурные врачи должны работать со справочной информацией (справочниками врачей и диагнозов по МКБ-10) и картой больного. При обращении пациента в городскую службу

«03» на него заводится карта пациента, содержащая персональные данные больного и данные анамнеза больного. Задачи поиска информации по заданным параметрам решаются исходя из данных карты пациента.

В процедуре заполнения карты больного (сущности «вызовы») участвуют связи между сущностью «вызовы» и справочниками врачей и диагнозов. Эти связи осуществляют выбор атрибуты справочников и подстановку их в соответствующий атрибут сущности «вызовы». Очевидно, что в этом случае мы имеем дело со связью «один ко многим».

Важным свойством модели «сущность–связь» является то, что она может быть представлена в виде графической схемы. Это значительно облегчает анализ предметной области. Существует несколько вариантов обозначения элементов диаграммы «сущность–связь», каждый из которых имеет свои положительные черты. На графической схеме сущности со связями соединяются прямыми линиями. В процессе построения диаграммы можно выделить несколько очевидных этапов:

- идентификация представляющих интерес сущностей и связей;
- идентификация семантической информации в наборах связей (например, является ли некоторый набор связей отображением 1:n);
- определение кардинальностей связей.

ER–диаграмма предметной области городской службы «03» приведена на рисунке 1.

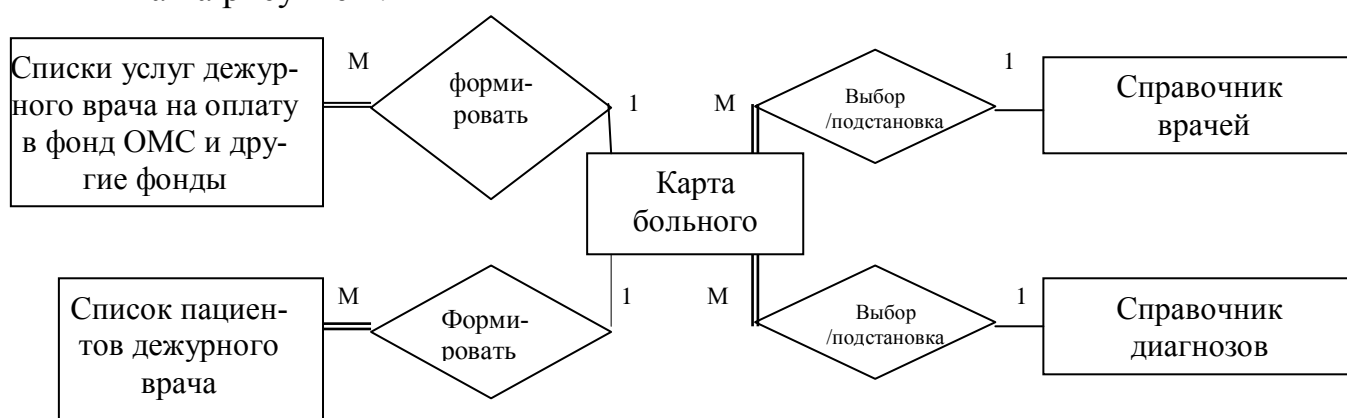


Рисунок 1 – ER–диаграмма предметной области городской службы «03»

Вывод: исследуемая предметная область состоит из организованного

набора данных: указанных выше сущностей с атрибутами и установленными между ними связями, отраженными на ER–диаграмме предметной области городской службы «03» (рис. 1).

1.2. Информационно-логическое проектирование

БД «Городская служба «03»

На этапе логического проектирования разрабатывается логическая структура БД, соответствующая логической модели программного обеспечения. Решение этой задачи существенно зависит от модели данных, поддерживаемой выбранной СУБД ACCESS (в нашем случае – это реляционная модель данных). На этапе логического проектирования базы данных «Городская служба «03» необходимо определить состав и взаимосвязи таблиц, отражающие содержание описанных выше информационных сущностей. Каждая таблица должна содержать наименование полей, идентификатор каждого поля и его шаблон. По каждой таблице должна быть информация о ключевом поле, длине одной записи.

Ниже приведено описание таблиц логической модели проектируемой БД «Городская служба «03». Для всех реквизитов выделенных сущностей предметной области задали форматы (символьные, числовые или дата/время).

Присвоим реквизитам сущности «Группы» имена, тип данных и формат данных (табл. 1).

Таблица 1 – Параметры реквизитов сущности «Группы»

Имя	Тип	Длина	Точность	Смысл
КодГруппы	Числовой	Счетчик (длинное целое)		Код пользовательской группы (уникальный)
Группа	Символьный	50		Наименование группы (администратор или дежурный врач)

Имя	Тип	Длина	Точность	Смысл
Пароль	Символьный	50		Пароль пользователя (для администратора – admin03, для дежурного врача – user03)

Присвоим реквизитам объекта «Врачи» имена, тип данных и формат данных (табл. 2).

Таблица 2 – Параметры реквизитов объекта «Врачи»

Имя	Тип	Длина	Точность	Смысл
КодВрача	Числовой	Счетчик (длинное целое)		Код врача (уникальный)
ФИО	Символьный	50		ФИО врача
Адрес	Символьный	50		Адрес проживания врача
Телефон	Символьный	50		Телефон врача (сотовый, домашний)

Присвоим реквизитам объекта «Диагнозы» имена, тип данных и формат данных (табл. 3).

Таблица 3 – Параметры реквизитов объекта «Диагнозы»

Имя	Тип	Длина	Точность	Смысл
КодМКБ	Символьный	10		Код болезни по классификатору МКБ-10 (уникальный)
Описание	Символьный	255		Описание болезни

Присвоим реквизитам объекта «Вызовы» имена, тип данных и формат данных (табл. 4).

Таблица 4 – Параметры реквизитов объекта «Вызовы»

Имя	Тип	Длина	Точность	Смысл
КодВызова	Числовой	Счетчик (длинное целое)		Код вызова (уникальный)
КодВрача	Числовой	Длинное целое		Код врача (внешний ключ)
ДатаВызова	Дата/время	Полный формат даты		Дата вызова врача скорой помощи
ФИОпациента	Символьный	50		ФИО пациента
ДатаРождения	Дата/время	Краткий формат даты		Дата рождения пациента

Имя	Тип	Длина	Точность	Смысл
АдресПациента	Символьный	50		Адрес пациента
Жалобы	Символьный	255		Жалобы пациента на недомогание
ДатаОказания	Дата/время	Полный формат даты		Дата оказания помощи врачом скорой помощи
ДиагнозПоМКБ	Символьный	10		Код болезни по классификатору МКБ-10 (внешний ключ)
НомерПолиса	Символьный	16		Номер страхового полиса нового единого образца

Присвоим реквизитам объекта «Анамнез» имена, тип данных и формат данных (табл. 5).

Таблица 5 – Параметры реквизитов объекта «Анамнез»

Имя	Тип	Длина	Точность	Смысл
КодВызова	Числовой	Счетчик (длинное целое)		Код вызова (уникальный)
АД_до	Символьный	20		Артериальное давление, мм.рт.ст. (до оказания помощи)
АД_после	Символьный	20		Артериальное давление, мм.рт.ст. (после оказания помощи)
Пульс_до	Символьный	20		Пульс, уд.в мин. (до оказания помощи)
Пульс_после	Символьный	20		Пульс, уд.в мин. (после оказания помощи)

Имя	Тип	Длина	Точность	Смысл
ЧастотаДыханияДо	Символьный	20		Частота дыхания, в мин. (до оказания помощи)
ЧастотаДыханияПосле	Символьный	20		Частота дыхания, в мин. (после оказания помощи)
ТемператураДо	Символьный	20		Температура, С (до оказания помощи)
ТемператураПосле	Символьный	20		Температура, С (после оказания помощи)

Для окончательного создания логической модели надо определить связи между объектами. В таблице 6 определены главные и подчиненные объекты в связях типа 1:М.

Таблица 6 – Связи информационных объектов

Главный ИО	Детальный ИО	Тип связи
Вызовы	Врачи	1:М
Вызовы	Диагнозы	1:М

Связь «один к одному» отражает взаимодействие сущностей «Вызовы» и «Анамнез» и используется для разделения таблиц по смысловому значению.

В соответствии с выявленными информационными объектами и связями в среде CASE-средства ERWin построена информационно–логическая модель (ИЛМ) данных, обеспечивающая решение задачи информационной поддержки городской службы «03» (рис. 2).

Логическая структура реляционной БД получена адекватным отображением ИЛМ данных (рис. 2). Она определяется совокупностью логически взаимосвязанных реляционных таблиц. Каждая реляционная таблица имеет структуру, определяемую реквизитным составом одного из информационных объектов (ИО) полученной ИЛМ. Логические связи таблиц соответствуют структурным связям между объектами [4]. На схеме (рис. 2) логические связи

отображены линиями между одинаковыми полями связи от главной таблицы к подчиненной.

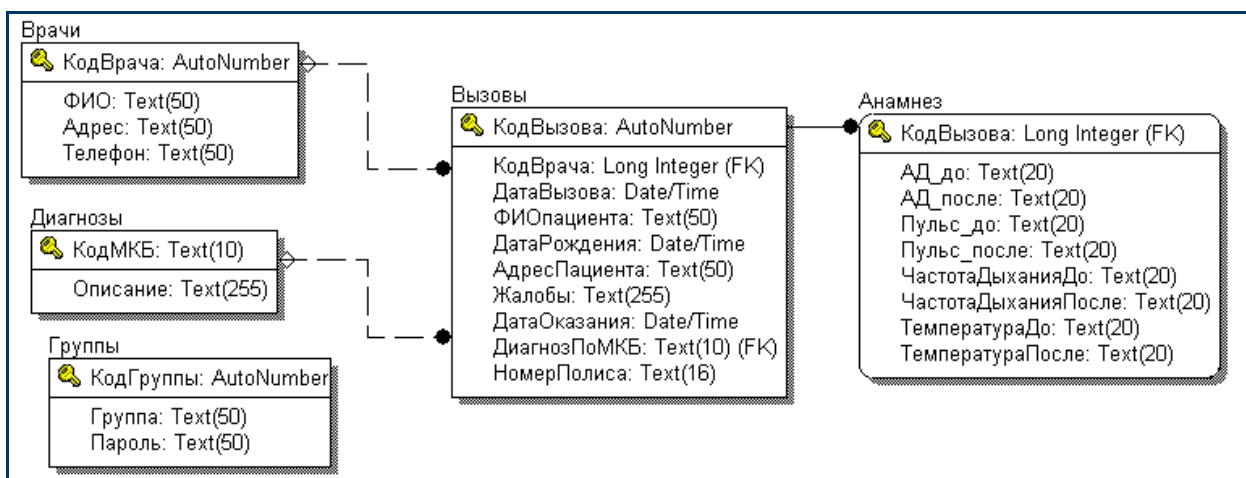


Рисунок 2 – Информационно–логическая модель данных

Логическая структура реляционной базы данных задачи информационной поддержки городской службы «03» отражает реквизиты базовых сущностей, уникальные и внешние ключи, а также связи между ИО в исследуемой предметной области.

2. Разработка структуры БД «Городская служба «03»

Физическому созданию БД предшествует информационно-логический этап ее проектирования. Плохая проработка структуры базы приводит к существенным затратам времени на ее переработку в дальнейшем. В целом разработка базы данных включает в себя следующие этапы:

- определение назначения базы данных;
- принятие решения о том, какие таблицы должна содержать БД. Например, всю возможную информацию из карты больного можно включить в одну таблицу, которая будет очень громоздкой, а можно распределить данные по двум тематическим таблицам, связанным между собой по ключу (например, вызовы и анамнез больного);
- определение состава полей таблиц БД. Поля БД соответствуют реквизитам информационных объектов предметной области;
- назначение связей между таблицами и окончательный просмотр полученной структуры. В реляционной БД таблицы могут быть связаны друг с другом. Эта связь устанавливается с помощью уникальных полей. Уникальные поля – это такие поля, в которых значения не могут повторяться, например, поле «КодВрача» в таблице «Врачи»;
- практическое создание таблиц, связывание их между собой и экспериментальное наполнение БД пробными данными. Для того чтобы определить, насколько структура базы данных соответствует поставленной задаче и насколько удобно с этой базой работать, необходимо ввести несколько простейших записей;
- создание форм, отчетов и запросов для операций с введенными данными. На заключительном этапе создают формы для ввода информации в базу, отчеты для вывода информации и запросы, с помощью которых производится выборка информации из нескольких таблиц.

Целью создания БД «Городская служба «03» является автоматизированная информационная поддержка деятельности службы скорой помощи средствами СУБД ACCESS. Такого рода автоматизация позволяет повысить

оперативность выбора необходимой информации из базы данных, провести анализ некоторых ситуаций, отраженных в заполненной базе данных «Городская служба «03».

2.1. Таблицы БД «Городская служба «03» и связи между ними

В соответствии с заданием на курсовую работу была создана база данных «СкораяПомощь03.mdb» в среде ACCESS. База данных состоит из 5 таблиц, структура которых приведена ниже.

- Таблица «Врачи» (рис. 3).

Врачи : таблица			
	Имя поля	Тип данных	Описание
	КодВрача	Счетчик	уникальное поле
	ФИО	Текстовый	ФИО врача
	Адрес	Текстовый	Адрес проживания
	Телефон	Текстовый	Телефон врача (сотовый, домашний)

Рисунок 3 – Структура таблицы «Врачи»

- Таблица «Диагнозы» (рис. 4).

Диагнозы : таблица			
	Имя поля	Тип данных	Описание
	КодМКБ	Текстовый	код болезни по справочнику МКБ-10
	Описание	Текстовый	описание болезни

Рисунок 4 – Структура таблицы «Диагнозы»

- Таблица «Группы» (рис. 5).

Группы : таблица			
	Имя поля	Тип данных	Описание
	КодГруппы	Счетчик	уникальный код группы пользователей
	Группа	Текстовый	наименование группы пользователей программы
	Пароль	Текстовый	пароль пользователя

Свойства поля	
Общие	Подстановка
Размер поля	50
Формат поля	
Маска ввода	Пароль
Подпись	
Значение по умолчанию	
Условие на значение	
Сообщение об ошибке	
Обязательное поле	Да
Пустые строки	Да
Индексированное поле	Нет
Сжатие Юникод	Да
Режим IME	Нет контроля
Режим предложений IME	Нет
Смарт-теги	

Обязательный ввод данных в это поле

Рисунок 5 – Структура таблицы «Группы»

– Таблица «Вызовы» (рис. 6).

Вызовы : таблица			
	Имя поля	Тип данных	Описание
?	КодВызова	Счетчик	уникальное поле
	КодВрача	Числовой	код врача
	ДатаВызова	Дата/время	дата вызова
	ФИОПациента	Текстовый	ФИО пациента
	ДатаРождения	Дата/время	дата рождения
	АдресПациента	Текстовый	Адрес пациента
	Жалобы	Текстовый	Жалобы пациента на недомогание
	ДатаОказания	Дата/время	Дата оказания помощи
	ДиагнозПоМКБ	Текстовый	диагноз заболевания
	НомерПолиса	Текстовый	номер полиса единого образца

Рисунок 6 – Структура таблицы «Вызовы»

– Таблица «Анамнез» (рис. 7).

Анамнез : таблица			
	Имя поля	Тип данных	Описание
?	КодВызова	Числовой	уникальное поле
	АД_до	Текстовый	Артериальное давление, мм.рт.ст (до оказания помощи)
	АД_после	Текстовый	Артериальное давление, мм.рт.ст (после оказания помощи)
	Пульс_до	Текстовый	Пульс, уд.в мин. (до оказания помощи)
	Пульс_после	Текстовый	Пульс, уд.в мин. (после оказания помощи)
	ЧастотаДыханияДо	Текстовый	Частота дыхания, в мин.(до оказания помощи)
	ЧастотаДыханияПосле	Текстовый	Частота дыхания, в мин.(после оказания помощи)
	ТемператураДо	Текстовый	Температура, С (до оказания помощи)
	ТемператураПосле	Текстовый	Температура, С (после оказания помощи)

Рисунок 7 – Структура таблицы «Анамнез»

Связи между созданными таблицами отражены на схеме данных (рис. 8).

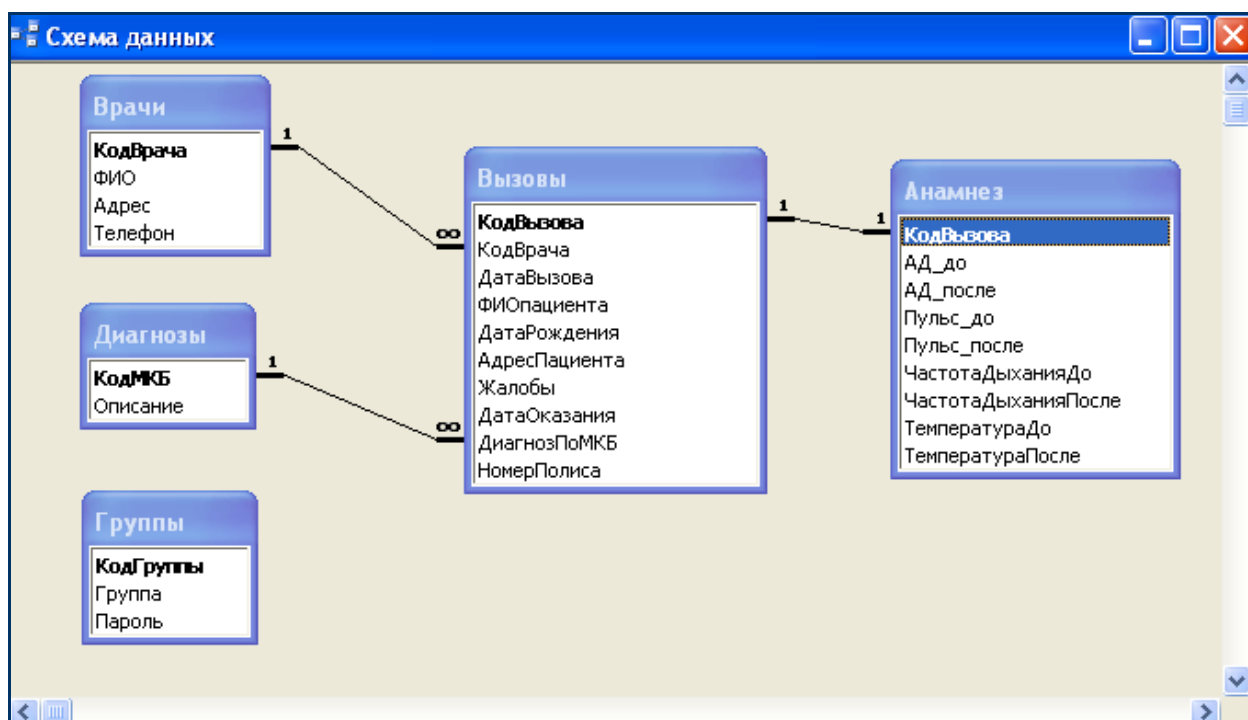


Рисунок 7 – Схема данных БД «Городская служба «03»

Как уже отмечалось, база данных имеет реляционную структуру, связи таблиц между собой осуществляются с помощью первичных (primary key) ключей.

Для выявления в таблице отдельной записи в реляционной базе данных используется ключ. Первичный ключ (Primary Key, PK) — это столбец, однозначно определяющий каждую запись в таблице.

Важную роль в реляционных базах данных играет еще один ключ – Внешний ключ (Foreign Key, FK). Внешний ключ – это столбец одной таблицы, который ссылается на первичный ключ другой таблицы. Для обеспечения целостности данных БД внешние ключи должны удовлетворять ограничению ссылочной целостности. Оно означает, что каждому значению внешнего ключа в одной таблице должно соответствовать значение существующего первичного ключа в другой таблице. Это важнейшее из всех ограничений, так как оно обеспечивает непротиворечивость перекрестных ссылок между таблицами [5].

Если корректность значений внешнего ключа не проверить, может нарушиться ссылочная целостность данных БД. Целостность данных является одним из самых важных требований, предъявляемых к базам данных. Для определения условий целостности данных в базе данных проекта могут быть использованы установленные между таблицами отношения. Ограничение ссылочной целостности должно поддерживаться автоматически. Каждый раз при вводе или изменении данных БД средства управления проверяют ограничения и убеждаются в их соблюдении. Если ограничения нарушаются, изменение данных запрещается. Именно определение связей между таблицами является первым шагом при формировании таблиц. Связи между таблицами реализованной базы данных, отраженные на рисунке 7, устанавливаются отношениями между совпадающими значениями в ключевых полях.

В реализованной базе данных между таблицами можно устанавливать отношения, которые будут поддерживаться при создании форм и запросов, возможно редактирование отношений. Установление отношений между таб-

лицами позволяет не вводить повторяющиеся данные в каждую таблицу. Разработчик теории реляционных баз данных Е.Кодд доказал, что для простоты манипулирования данными в таблицах реляционных баз данных при создании их рекомендуется нормализовать эти данные, т.е. исключить избыточную информацию. Для определения структуры каждой таблицы необходимо выполнить анализ функциональных зависимостей, т.е. выяснить, какие поля зависят от других полей, а затем поля с одинаковой зависимостью организовать в отдельную таблицу. Таким образом, в одну и ту же таблицу не нужно включать поля, являющиеся производными от других полей. В результате количество необходимых таблиц определяется числом функциональных зависимостей. Формально нормализация данных обеспечена, если набор таблиц удовлетворяет первым трем правилам, которые называют нормальными формами.

Первая нормальная форма требует отсутствия повторяющихся полей в первичном ключе.

Вторая – чтобы каждый столбец таблицы зависел от своего первичного ключа.

Третья – чтобы все непервичные поля зависели только от первичных.

После определения связей между данными таблиц были созданы запросы и формы, в которых выводятся данные из одной или нескольких таблиц сразу.

2.2. Запросы БД «Городская служба «03»

Одним из важных моментов разработанной БД «Городская служба «03» является организация поиска информации. Эта организация реализована на основе запросов к базе данных. С помощью конструктора запросов ACCESS можно формировать критерии различной сложности для выбора записей из одной или нескольких таблиц, указывая при этом, какие поля должны быть отражены в запросе. Над полями, выбираемыми из таблиц с помощью запросов, можно выполнять различные вычисления. Результатом запроса является

визуальная таблица.

В качестве примеров запросов, используемых в программе, приведем следующие:

– запрос на выборку: «Медицинские услуги дежурного врача, оплата которых происходит из фонда Обязательного Медицинского Страхования (ОМС). Данную оплату определяет наличие страхового полиса у больного. SQL-код запроса:

```
SELECT      Вызовы.КодВрача,      Вызовы.ДатаВызова,      Вызо-
вы.ФИОпациента,      Вызовы.ДатаРождения,      Вызовы.ДатаОказания,      Вызо-
вы.ДиагнозПоМКБ,      Вызовы.НомерПолиса
```

```
FROM Вызовы
```

```
WHERE (((Вызовы.НомерПолиса) Is Not Null))
```

```
ORDER BY Вызовы.ДатаВызова, Вызовы.ФИОпациента;
```

Результат запроса:

ОплатаОМС						
Врач	Дата вы- зова	ФИО пациента	Дата ро- ждения пациента	Да- та/вре- мя оказа- ния помощи	Диагноз	Номер полиса
Алексее- ва Анна Сергеев- на	10.07.2013 10:30:10	Костров Виктор Алексеевич	11.05.1990	10.07.2013 10:31:00	Острый назофа- рингит	555976324187102 5
Быстров Николай Иванович	11.07.2013 11:45:00	Лапина Викто- рия Григорьев- на	01.05.1985	11.07.2013 12:10:00	Врожденный син- дром йодной не- достаточности	552179842421245 2
Волков Иван Егор- ович	15.07.2013 9:20:00	Новиков Антон Константино- вич	05.05.1988	15.07.2013 10:02:00	ОСТРЫЕ РЕСПИ- РАТОРНЫЕ ИН- ФЕКЦИИ ВЕРХ- НИХ ДЫХАТЕЛЬ- НЫХ ПУТЕЙ	554878742454575 4
Быстров Николай Иванович	16.07.2013 15:00:00	Чубаков Денис Михайлович	12.10.1955	16.07.2013 15:20:00	Острый назофа- рингит	558878210011122 3
Волков Иван Егор- ович	16.07.2013 20:55:47	Зайцев Алек- сандр Нико- лаевич	15.03.1968	16.07.2013 21:10:00	БОЛЕЗНИ ОРГА- НОВ ДЫХАНИЯ	558742146741242 4

– запрос на выборку: «Медицинские услуги дежурного врача, оплата которых происходит не из фонда Обязательного Медицинского Страхования. Данную оплату определяет отсутствие страхового полиса у больного. SQL-

код запроса:

```
SELECT      Вызовы.КодВрача,      Вызовы.ДатаВызова,      Вызо-
вы.ФИОпациента,      Вызовы.ДатаРождения,      Вызовы.ДатаОказания,      Вызо-
вы.ДиагнозПоМКБ,      Вызовы.НомерПолиса

FROM      Вызовы

WHERE      (((Вызовы.НомерПолиса) Is Null))

ORDER BY      Вызовы.ДатаВызова,      Вызовы.ФИОпациента;
```

Результат запроса:

Врач	Дата вызова	ФИО пациента	Дата рождения пациента	Дата/время оказания помощи	Диагноз
Волков Иван Егорович	15.07.2013 17:00:00	Цаплин Игорь Юрьевич	02.05.1977	15.07.2013 17:25:00	ИНФЕКЦИОННЫЕ И ПАРАЗИТАРНЫЕ БОЛЕЗНИ
Алексеева Анна Сергеевна	15.07.2013 20:03:00	Федорова Анна Дмитриевна	15.05.1975	15.07.2013 21:00:00	КИШЕЧНЫЕ ИНФЕКЦИИ
Быстров Николай Иванович	17.07.2013 0:42:24	Дубова Лидия Ивановна	11.11.1968	17.07.2013 1:10:00	БОЛЕЗНИ ВЕК, СЛЕЗНЫХ ПУТЕЙ И ГЛАЗНИЦЫ

Тестирование БД по реализованному запросу показало работоспособность созданной БД «Городская служба «03».

2.3. Формы БД «Городская служба «03»

Формы – это средства для ввода и отображения данных. Форма представляет собой бланк, подлежащий заполнению, или маску, накладываемую на набор данных. Форма-бланк позволяет упростить процесс заполнения БД, что дает возможность поручить ввод информации персоналу невысокой квалификации. С помощью формы-маски можно ограничить объем информации, доступной пользователю, обращающемуся к базе данных.

Разработка удобного интерфейса для программного продукта информационной поддержки городской службы скорой помощи заключается в следующем:

- создание формы входа в информационную систему (ИС) «Городская

служба «03», служащей для ввода имени пользователя и его пароля, и последующего вызова форм, предназначенных для выбранного пользователя.

- создание программных форм, содержащих главные навигационные кнопки для работы выбранного пользователя с базой данных.
- создание экранных форм для работы с картой больного, справочниками врачей и диагнозов (добавление, изменение, удаление записей);
- создание дополнительных информационных программных форм, которые являются результатом выполнения реализованных запросов.

Разработанные формы – это организованное и отформатированное представление некоторых или всех полей из одной или нескольких таблиц или запросов. Формы интерактивно взаимодействуют с таблицами базы данных. Для ввода новой информации, редактирования или удаления существующей информации используются элементы управления формой. Формы значительно облегчают пользователям ввод, получение, отображение и печать информации, хранящейся в таблицах. Форма – это окно, в которое можно поместить элементы управления, которые либо предоставляют пользователям информацию, либо получают информацию, которую они вводят. Формы обычно предоставляют подробную информацию о записях и предназначены для тех людей, которые реально работают с базой данных.

Формы включают элементы управления статического текста (метки), которые говорят пользователю, какого типа информацию от них требуется ввести, а также элементы управления текстовых полей, в которых можно эту информацию внести. Созданные формы включают такие элементы управления, как кнопки. Для навигации по функциям приложения базы данных используются кнопочные формы.

В ходе проектирования БД были созданы формы, интерфейс которых приведен ниже.

Разработанная база данных «Городская служба «03» позволяет организовать автоматизированную информационную поддержку деятельности службы скорой помощи, упростить учет оплаты оказанных медицинских ус-

луг больным, наладить контроль и анализ результатов работы дежурных врачей (в карте больного фиксируется время вызова и время оказания помощи). Работа пользователя с разработанным программным обеспечением заключается в следующем:

- работа с базой данных (занесение, изменение, поиск и удаление данных);
- работа с формами учета статистических данных (учета услуг, оказанных, каждым дежурным врачом, и фондов, из которых будет проходить оплата этих услуг);

При установке программы пользователь может размещать базу данных «СкораяПомощь03.mdb» в компьютере в любой раздел жесткого диска и в любую его директорию. Работа программы начинается с загрузки формы входа в систему (рис. 8), происходящей при запуске файла «СкораяПомощь03.mdb». Программное обеспечение работает в среде MICROSOFT Office ACCESS версии не ниже 2003.

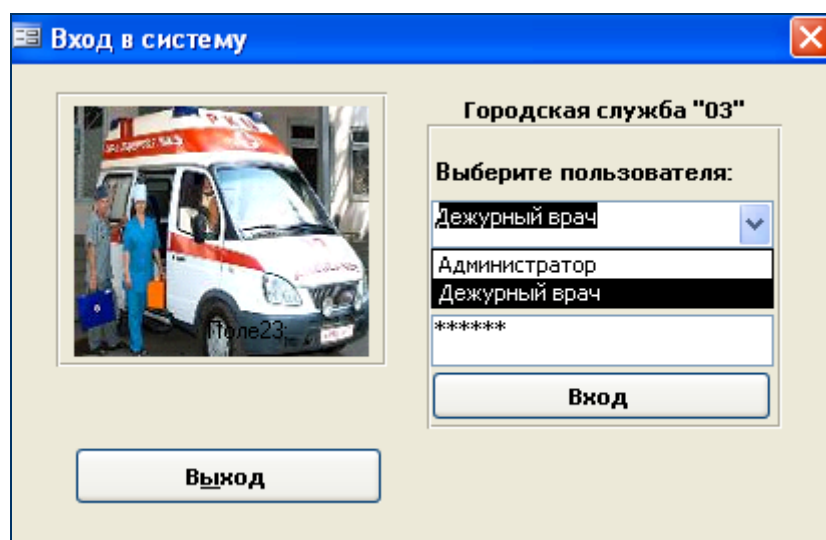


Рисунок 8 – Форма входа в информационную систему «Городская служба «03»

В основу интерфейса заложен принцип кнопочной формы. Кнопки главной формы «Меню дежурного врача» (рис. 9) служат способом доступа к функциям системы. Форма «Меню администратора» (рис. 10) служит для доступа администратора БД к объектам базы данных.

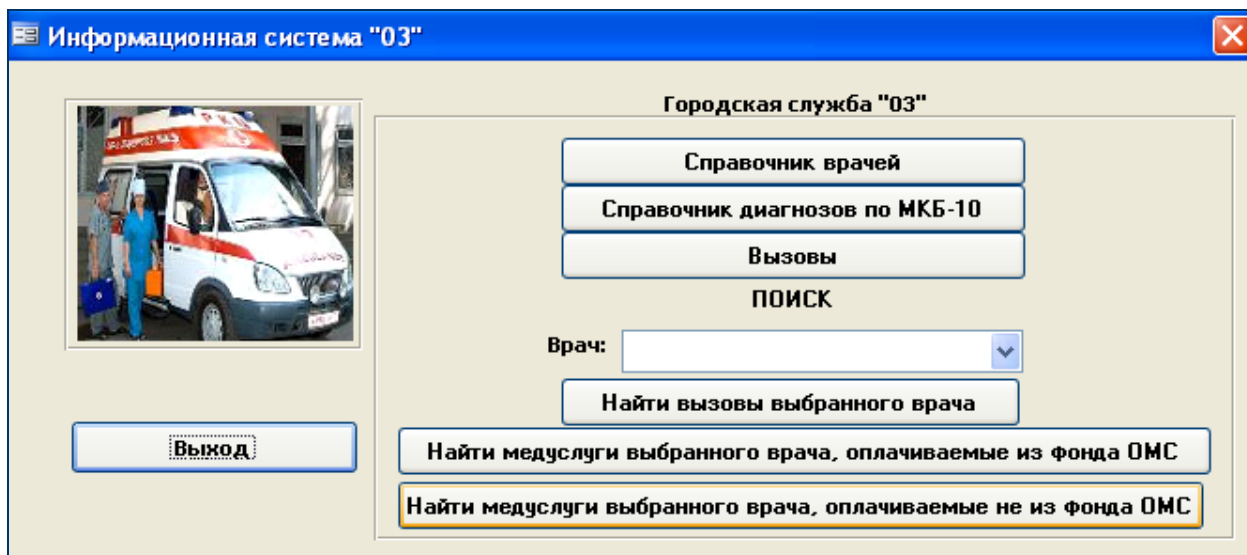


Рисунок 9 – Форма «Меню дежурного врача» ИС «Городская служба «03»

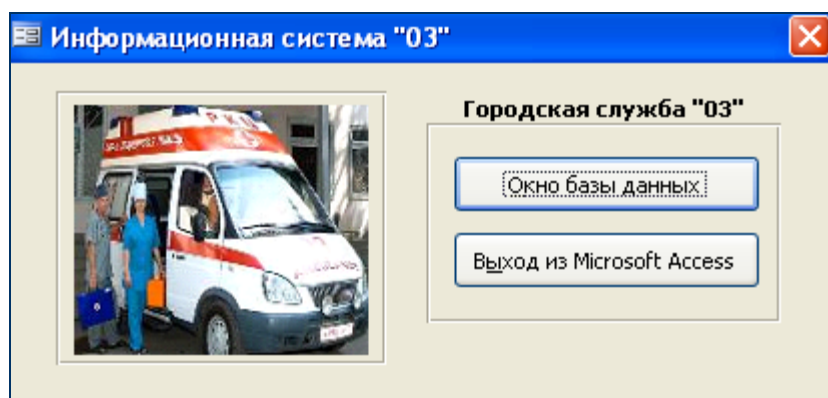


Рисунок 10 – Форма «Меню администратора» ИС «Городская служба «03»

Все окна имеют стандартные возможности управления (изменить размер, развернуть, свернуть, закрыть). Также можно изменять относительные размеры левой и правой части окна.

Для редактирования персональных данных врачей и диагнозов созданы два справочника, работа с которым происходит в окнах, отраженных на рисунках 11-12.

При обращении больного в службу скорой помощи заводится карта пациента, заполнение которой ведется в соответствии с окном, изображенным на рисунке 13. При заполнении карты больного значения полей *Врач* и *Диагноз* заполняются из справочника (ниспадающее меню выбора).

ФИО	Адрес	Телефон
Алексеева Анна Сергеевна	644029, г. Омск, ул. Кирова, 87-93	(8)913-632-98-53, д.т. 54-35-59
Быстров Николай Иванович	644033, г. Омск, ул. Пушкина, 107-93	(8)950-654-98-87, д.т. 33-99-88
Волков Иван Егорович	644023, г. Омск, ул. Транспортная, 78-99	(8)952-654-95-87, д.т. 54-99-88
*		

Добавить Сохранить Удалить

Запись: 1 из 3

Рисунок 11 – Форма «Справочник врачей» ИС «Городская служба «03»

Код МКБ	Описание
A00-A09	КИШЕЧНЫЕ ИНФЕКЦИИ
A00-B99	ИНФЕКЦИОННЫЕ И ПАРАЗИТАРНЫЕ БОЛЕЗНИ
B00	Инфекции, вызванные вирусом герпеса [herpes simplex]
B00.0	Герпетическая экзема
B00.1	Герпетический везикулярный дерматит
B00.2	Герпетический гингивостоматит и фаринготонзиллит
B00.7	Диссеминированная герпетическая болезнь
B00.8	Другие формы герпетических инфекций
B00.9	Герпетическая инфекция неуточненная
B00-B09	ВИРУСНЫЕ ИНФЕКЦИИ, ХАРАКТЕРИЗУЮЩИЕСЯ ПОРАЖЕНИЯМИ КОЖИ И СЛИЗИСТЫХ ОБОЛОЧЕК
C00-D48	НОВООБРАЗОВАНИЯ
E00	Врожденный синдром йодной недостаточности
E00.0	Синдром врожденной йодной недостаточности, неврологическая форма

Добавить Сохранить Удалить

Запись: 11 из 83

Рисунок 12 – Форма «Справочник диагнозов» ИС «Городская служба «03»

Врач: Алексеева Анна Сергеевна

Дата/время вызова: 10.07.2013 10:30:10

ФИО пациента: Костров Виктор Алексеевич

Дата рождения пациента: 11.05.1990

Адрес пациента: г. Омск, ул. Герцена, 93-85

Жалобы: Температура 39, рвота, озноб

Дата/время оказания помощи: 10.07.2013 10:31:00

Диагноз: Острый назофарингит

Номер полиса: 5559763241871025

	До оказания помощи	После оказания помощи
Артериальное давление, мм.рт.ст.	100/70	120/80
Пuls, уд. в мин.	50-55	60-80
Частота дыхания, в мин.	25-30	18-20
Температура, С	38,8	36,8

Удалить запись Добавить запись Сохранить запись

Запись: 1 из 8

Рисунок 13 – Форма для заполнения карты больного ИС «Городская служба «03»

Заполненные карточки пациентов в дальнейшем используются для поиска оказанных услуг каждым дежурным врачом (рис. 14) и определения

фондов, из которых будет происходить оплата этих услуг. Наличие страхового полиса у пациента определяет фонд обязательного медицинского страхования (рис. 15), отсутствие страхового полиса определяет другие фонды оплаты медицинских услуг, например, бюджетные (рис. 16).

Врач	Дата/время вызова	ФИО пациента	Адрес пациента	Дата/время оказания помощи	Диагноз
Быстров Николай Иванович	11.07.2013 11:45:00	Лапина Виктория Григорьевна	г. Омск, ул. Нефтезаводская, 79	11.07.2013 12:10:00	Врожденный синдром йодной не...
Быстров Николай Иванович	16.07.2013 15:00:00	Чубаков Денис Михайлович	644055, г. Омск, ул. Жукова, 101	16.07.2013 15:20:00	Острый назофарингит
Быстров Николай Иванович	17.07.2013 0:42:24	Дубова Лидия Ивановна	г. Омска, ул. Б.Хмельницкого, 201	17.07.2013 1:10:00	БОЛЕЗНИ ВЕК, СЛЕЗНЫХ ПУТ...

Рисунок 14 – Учет медицинских услуг, оказанных дежурным врачом

Врач	Дата вызова	ФИО пациента	Дата рождения пациента	Дата/время оказания помощи	Диагноз	Номер полиса
Быстров Николай Иванович	11.07.2013 11:45:00	Лапина Виктория Григорьевна	01.05.1985	11.07.2013 12:10:00	Врожденный синдром йодной недост	552179842421245
Быстров Николай Иванович	16.07.2013 15:00:00	Чубаков Денис Михайлович	12.10.1955	16.07.2013 15:20:00	Острый назофарингит	558878210011122
	18.07.2013 9:49:01					

Рисунок 15 – Учет медицинских услуг, оказанных дежурным врачом и оплачиваемых из фонда ОМС

Врач	Дата вызова	ФИО пациента	Дата рождения пациента	Дата/время оказания помощи	Диагноз
Быстров Николай Иванович	17.07.2013 0:42:24	Дубова Лидия Ивановна	11.11.1968	17.07.2013 1:10:00	БОЛЕЗНИ ВЕК, СЛЕЗНЫХ ПУТЕЙ И ГЛАЗНИЦЫ
	18.07.2013 9:51:17				

Рисунок 16 – Учет медицинских услуг, оказанных дежурным врачом и оплачиваемых не из фонда ОМС

Программное обеспечение по информационной поддержке деятельности городской службы скорой помощи позволяет вести учет времени приема вызова и времени оказания медицинской помощи, а также позволяет оценить эффективность проведенной медицинской услуги: организован учет показателей здоровья больного до и после оказания помощи (артериального давления, температуры, частоты дыхания и пульса).

Переход между ячейками ввода на формах осуществляется клавишами <Enter>, <Tab> или <↓↑>. Для удаления записи на всех формах имеется соответствующая кнопка, удаление записи происходит после подтверждения пре-

дупреждения об удалении.

Смысл названия всех кнопок окон ИС «Городская служба «03» соответствует их назначению.

3. Криптографическая защита данных БД «Городская служба «03»

С помощью системы управления базами данных (СУБД) MICROSOFT ACCESS можно использовать различные методы защиты баз данных. Для защиты базы данных, которая совместно используется лишь небольшой группой пользователей обычно оказывается достаточным установка парольной защиты и шифрования.

Установка пароля для открытия базы данных является простым и надежным способом защиты. После того как пароль установлен, при каждом открытии базы данных будет появляться диалоговое окно, в которое требуется ввести пароль. Только те пользователи, которые введут правильный пароль, смогут открыть базу данных. Этот способ достаточно надежен (MICROSOFT ACCESS шифрует пароль, так что к нему нет прямого доступа при чтении файла базы данных), но он применяется только при открытии базы данных. После открытия базы данных все объекты становятся доступными для пользователя. Для базы данных, которая совместно используется небольшой группой пользователей установка пароля обычно оказывается достаточной.

Защита на уровне пользователя – наиболее гибкий и распространенный метод реализации средств защиты приложений MICROSOFT ACCESS [1].

Двумя основными причинами использования защиты на уровне пользователей являются:

- защита приложения от повреждения из-за неумышленного изменения пользователями таблиц, запросов, форм, отчетов и макросов, от которых зависит работа приложения;
- защита конфиденциальных сведений в базе данных.

При использовании защиты приложения на уровне пользователей пользователь должен ввести пароль при запуске MICROSOFT ACCESS. В СУБД ACCESS мастер защиты (Сервис–Защита–Мастер) обеспечивает быстрое и легкое применение новейших средств безопасности приложения ACCESS Мастер защиты помогает назначить разрешения и создать учетные записи

пользователей и учетные записи групп. После запуска мастера для базы данных и существующих в ней таблиц, запросов, форм, отчетов и макросов имеется возможность вручную назначить или удалить разрешения на доступ для учетных записей пользователей и групп в рабочей группе. Также могут быть установлены разрешения на доступ, по умолчанию присваиваемые вновь создаваемым объектам базы данных.

Группам и пользователям предоставляются разрешения, определяющие возможность их доступа к каждому объекту базы данных. Например, члены группы «Users» могут иметь разрешения на просмотр, ввод или изменение данных в таблице «Вызовы», но им не будет разрешено изменять структуру этой таблицы. Члены группы «Admins» имеют разрешения на доступ ко всем объектам приложения. Имеется возможность установить более разветвленную структуру управления, создавая собственные учетные записи групп, предоставляя этим группам соответствующие разрешения и добавляя пользователей в группы.

Другой способ защиты доступа к информации базы данных – программная реализация входа в ИС средствами VBA.

Ниже приведен код VBA для хранения групп пользователей с паролями в таблице *Группы*, выбора при входе группы и ввода пароля и запуск разных форм в соответствии с выбранной группой. В таблице *Группы* занесены два пользователя *Администратор* и *Дежурный врач*, пароли для них соответственно *admin03* и *user03*. В ИС «Городская служба «03» созданы две формы-меню: для *Администратора* и для *Дежурного врача*. При запуске ИС «Городская служба «03» стартует форма *Вход*, на которой расположены:

- поле со списком *ВыборГруппы*, для которого реализована подстановка наименования группы *Группа* (*КодГруппы* и *Пароль* скрыты);
- поле *ВводПароля* с соответствующей маской ввода *Пароль* и
- кнопка *ОК*.

Код VBA, запускаемый при нажатии на кнопку *ОК*:

```
Private Sub Кнопка20_Click()
```

```

'Проверяем выбрана ли группа, если нет, выдаем сообщение
If IsNull(Ме.ВыборГруппы) Then
MsgBox "Выберите пользователя!"
Else
    'Проверяем введен ли пароль, если нет, выдаем сообщение
    If IsNull(Ме.ВводПароля) Then
    MsgBox "Введите пароль!"
    Else
        'Сверяем введенный пароль с паролем, сохраненным в таблице
        Группы,
        'т.е. обращаемся к 3-му столбцу поля со списком ВыборГруппы.
        Column(2), если пароль не верный, выдаем сообщение
        If Ме.ВводПароля <> Ме.ВыборГруппы.Column(2) Then
        MsgBox "Пароль не верный!"
        Else
            'В зависимости от выбранной группы открываем соответствующую
            форму
            Select Case Ме.ВыборГруппы
            Case 1
            DoCmd.OpenForm "МенюАдминистратора"
            Case 2
            DoCmd.OpenForm "МенюВрача"
            End Select
        End If
    End If
End If
End Sub

```

В СУБД MS ACCESS возможно шифрование базы данных (Сервис–Защита–Закодировать или раскодировать базу данных). Шифрование базы данных защищает ее от несанкционированного просмотра с помощью слу-

жебных программ или текстовых редакторов. При шифровании базы данных ее файл сжимается, а работа с базой замедляется. Шифрование незащищенной базы данных неэффективно, так как ее можно открыть в MS ACCESS и получить полный доступ ко всем объектам. Шифрование защищенной базы данных – это хорошая защита от несанкционированного вмешательства при передаче по линии связи, хранении на съемных носителях информации и т.п.

Заключение

В данной курсовой работе была разработана база данных «Городская служба «03» для работы с медицинскими картами пациентов службы скорой помощи. Созданная база данных предназначена для создания и сопровождения медицинских карт пациентов, а также для учета статистических данных, характеризующих деятельность дежурного врача службы скорой помощи. БД «Городская служба «03» позволяет осуществлять ввод, редактирование и поиск информации.

Основным этапом выполнения курсовой работы стала физическая разработка программных модулей базы данных «Городская служба «03». Ей предшествовали этапы разработки концептуальной модели предметной области (деятельности службы скорой помощи), логического проектирования БД, подробно описанные в главе 1. Разработаны и отлажены таблицы, схема данных, программные формы и запросы базы данных «Городская служба «03» под операционную систему Windows.

Разработанная программа позволяет автоматизировать следующие функции:

- работа с базой данных (хранение, добавление, редактирование, удаление информации о пациентах, врачах и диагнозах);
- выбор интересующей информации из базы данных для дальнейшей статистической отчетности по оказанным медицинским услугам дежурных врачей и определения фонда, из которого будет проводиться оплата этих услуг.

Прикладное значение разработанного программного обеспечения: создана база данных автоматизированной информационной поддержки деятельности городской службы «03», реализован авторизованный вход в систему и удобный интерфейс для работы с БД, решены задачи заполнения справочников врачей и диагнозов, карт больных и учета услуг, оказываемых дежурными врачами службы скорой помощи.

Список использованной литературы

1. MICROSOFT ACCESS 2003. Шаг за шагом /Практическое пособие/ Пер.с англ. – М: «СП ЭКОМ», 2007 – 432с.
2. Дейт К. Дж. Введение в системы баз данных. 8-е изд. – М.: «Вильямс», 2006 – 679с.
3. Коннолли Т., Бегг К., Страчан А. Базы данных: проектирование, реализация, сопровождение. Теория и практика, 2-е изд.: Пер. с англ.: Уч. пос. – М.: Изд. дом «Вильямс», 2007. – 1120с.
4. Крёнке Д. Теория и практика построения баз данных. – СПб.: Питер, 2005. – 800с.
5. Чекалов А. Базы данных. От проектирования до разработки приложений. – СПб.: ВHV-Санкт-Петербург, 2003 – 437с.