

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего  
профессионального образования  
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И  
РАДИОЭЛЕКТРОНИКИ» (ТУСУР)  
Кафедра комплексной информационной безопасности электронно-вычислительных систем  
(КИБЭВС)

УТВЕРЖДАЮ

заведующий каф. КИБЭВС

\_\_\_\_\_ А.А. Шелупанов

« \_\_\_\_\_ » \_\_\_\_\_ 2015г.

ПРОЕКТИРОВАНИЕ, РАЗРАБОТКА БАЗЫ ДАННЫХ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
ДЛЯ ЭЛЕКТРОННОЙ РЕГИСТРАЦИИ НА ПРИЕМ К ВРАЧУ (ЭЛЕКТРОННАЯ  
РЕГИСТРАТУРА)

Курсовая работа по дисциплине «Безопасность систем баз данных»

Пояснительная записка к курсовой работе

Выполнила:

студентка гр. 722

\_\_\_\_\_ М.В. Мейта

« \_\_\_\_\_ » \_\_\_\_\_ 2015г.

Научный руководитель:

аспирант каф. КИБЭВС

\_\_\_\_\_ И.В. Горбунов

« \_\_\_\_\_ » \_\_\_\_\_ 2015г.

## РЕФЕРАТ

Курсовая работа содержит 29 страниц, 7 рисунка, 4 таблицы, 3 источника, 1 приложение.  
БАЗЫ ДАННЫХ, SQLITE, MONODEVELOP, C#, GTKSharp.

Цель работы — проектирование, разработка базы данных и клиентской части программного обеспечения для электронной регистрации на прием к врачу (электронная регистрация).

Результатом выполнения работы является база данных и графическое приложение для осуществления регистрации пациентов, записи на прием к специалисту, администрирования записей в базе данных (просмотр, удаление и добавление информации о сотрудниках поликлиники, пациентах, выданных талонах и др.).

В процессе работы были выполнены все вышепоставленные цели, разработана инфологическая модель данных для описания процесса регистрации и структуры базы данных, предусмотрены ограничения на ввод данных, а также применены средства обеспечения безопасности базы данных на уровне приложения.

Проект выполнен с использованием следующих средств разработки:

- ОС Linux Ubuntu 14.10;
- язык программирования C#;
- среда разработки MonoDevelop 4.0.12 [1];
- встраиваемая реляционная база данных SQLite [2];
- СУБД SQLiteman 1.2.2;
- кроссплатформенная библиотека элементов графического интерфейса GTKSharp;
- iTextSharp — .NET PDF библиотека для генерации PDF-документов [3];
- система контроля версий Git.

Пояснительная записка выполнена при помощи системы компьютерной вёрстки L<sup>A</sup>T<sub>E</sub>X.

## Содержание

ТЕХНИЧЕСКОЕ ЗАДАНИЕ . . . . .	6
1 Общие сведения . . . . .	7
1.1 Полное наименование системы и ее условное обозначение . . . . .	7
1.2 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты . . . . .	7
1.3 Требования, на основании которых создается система, и даты их утверждения . . . . .	7
1.4 Плановые сроки начала и окончания работы по созданию системы . . . . .	7
1.5 Сведения об источниках и порядке финансирования работ . . . . .	7
1.6 Порядок оформления и предъявления заказчику результатов работ по созданию системы (ее частей), по изготовлению и наладке отдельных средств (технических, программных, информационных) и программно-технических (программно-методических) комплексов системы . . . . .	7
2 Назначение и цели создания (развития) системы . . . . .	7
2.1 Назначение системы . . . . .	7
2.2 Цели создания системы . . . . .	8
3 Характеристика объектов автоматизации . . . . .	8
3.1 Краткие сведения об объекте автоматизации или ссылки на документы, содержащие такую информацию . . . . .	8
4 Требования к системе . . . . .	9
4.1 Общие требования к системе . . . . .	9
4.1.1 Входные и выходные данные . . . . .	9
4.1.2 Требования к транспортированию и хранению . . . . .	9
4.1.3 Требования к информационной и программной совместимости . . . . .	9
4.1.4 Требования к надежности . . . . .	9
4.1.5 Требования к эргономике и технической эстетике . . . . .	9
4.1.6 Требования к эксплуатации . . . . .	9
4.1.7 Требования к маркировке и упаковке . . . . .	9
4.2 Требования к функциям (задачам), выполняемым системой . . . . .	9
4.2.1 Требования к составу выполняемых функций . . . . .	9
4.3 Требования к видам обеспечения . . . . .	9
4.3.1 Требования к техническому обеспечению системы . . . . .	9
4.3.2 Требования к информационной и программной совместимости . . . . .	9
5 Состав и содержание работ по созданию системы . . . . .	9
5.1 Список исполнителей работ . . . . .	9
5.2 Перечень документов, оформленных по ГОСТ 34.201-89, предъявляемых по окончании соответствующих стадий производства . . . . .	9
6 Порядок контроля и приема системы . . . . .	9
6.1 Виды, состав, объем и методы испытаний системы и ее составных частей (виды испытаний в соответствии с действующими нормами, распространяющимися на разрабатываемую систему) . . . . .	9

6.2	Общие требования к приему работ по стадиям (перечень участвующих предприятий и организаций, место и сроки проведения), порядок согласования и утверждения приемочной документации . . . . .	10
6.3	Статус приемной комиссии (государственная, межведомственная, ведомственная) . .	10
7	Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие . . . . .	10
8	Требования к документированию . . . . .	10
8.1	Согласованный разработчиком и заказчиком системы перечень подлежащих разработке комплектов и видов документов . . . . .	10
8.2	Требования к документированию . . . . .	10
9	Источники разработки . . . . .	10
	ВВЕДЕНИЕ . . . . .	12
1	Проектирование инфологической модели данных . . . . .	13
2	Проектирование даталогической модели данных . . . . .	16
3	Описание базы данных . . . . .	17
3.1	Таблица «patient» . . . . .	17
3.2	Таблица «passport» . . . . .	17
3.3	Таблица «policy» . . . . .	17
3.4	Таблица «talon» . . . . .	17
3.5	Таблица «timetable» . . . . .	17
3.6	Таблица «employee» . . . . .	17
3.7	Роли базы данных . . . . .	18
3.8	Защита от SQL-инъекций . . . . .	18
4	Описание процесса деятельности . . . . .	19
4.1	Постановка задачи . . . . .	19
4.2	Описание данных программы . . . . .	19
4.2.1	Входные данные . . . . .	19
4.2.2	Выходные данные . . . . .	19
4.3	Основные технические решения . . . . .	22
4.3.1	Алгоритм . . . . .	22
4.3.2	Численность, функции и квалификация персонала . . . . .	23
4.3.3	Обеспечение потребительских характеристик системы . . . . .	23
4.3.4	Функции, выполняемые системой . . . . .	23
4.3.5	Комплекс технических средств . . . . .	23
4.3.6	Информационное обеспечение системы . . . . .	23
4.3.7	Программное обеспечение системы . . . . .	23
5	Мероприятия по подготовке персонала . . . . .	23
6	Руководство пользователя . . . . .	24
7	Руководство программиста . . . . .	25
8	Перспективы применения программы . . . . .	26
	ЗАКЛЮЧЕНИЕ . . . . .	27
	Список использованных источников . . . . .	28

Приложение А Компакт-диск . . . . .	29
-------------------------------------	----

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего  
профессионального образования  
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И  
РАДИОЭЛЕКТРОНИКИ» (ТУСУР)  
Кафедра комплексной информационной безопасности электронно-вычислительных систем  
(КИБЭВС)

УТВЕРЖДАЮ

Заведующий каф. КИБЭВС,  
доктор технических наук, профессор

\_\_\_\_\_ А.А. Шелупанов

«\_\_\_\_\_» \_\_\_\_\_ 2015г.

УТВЕРЖДАЮ

Руководитель группы,  
студентка гр. 722

\_\_\_\_\_ М.В. Мейта

«\_\_\_\_\_» \_\_\_\_\_ 2015г.

ПРОЕКТИРОВАНИЕ, РАЗРАБОТКА БАЗЫ ДАННЫХ И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
ДЛЯ ЭЛЕКТРОННОЙ РЕГИСТРАЦИИ НА ПРИЕМ К ВРАЧУ (ЭЛЕКТРОННАЯ  
РЕГИСТРАТУРА)

Курсовая работа по дисциплине «Безопасность систем баз данных»

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на 11 листах

Действует с 1.03.2015

СОГЛАСОВАНО

аспирант каф. КИБЭВС

\_\_\_\_\_ Горбунов И.В.

«\_\_\_\_\_» \_\_\_\_\_ 2015г.

## 1 Общие сведения

### 1.1 Полное наименование системы и ее условное обозначение

Полное название программы: «Электронная регистратура».

Условное обозначение: «hospital\_register».

### 1.2 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты

Разработчик: студентка гр.722 ФБ ТУСУРа: Мейта Марина Валерьевна.

Заказчик: Томский государственный университет систем управления и радиоэлектроники (ТУСУР), факультет безопасности (ФБ), в лице аспиранта кафедры комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС) Горбунова И. В.

### 1.3 Требования, на основании которых создается система, и даты их утверждения

Задание на выполнение курсового проекта по дисциплине «Безопасность систем баз данных» утверждено Горбуновым И. В. 1 марта 2015 г.

### 1.4 Плановые сроки начала и окончания работы по созданию системы

Дата начала работы — 1 марта 2015 года, дата окончания работы — 1 июня 2015 года.

### 1.5 Сведения об источниках и порядке финансирования работ

Финансирование осуществляется лицами, заинтересованными в разработке программного средства, а именно разработчиком из собственных средств.

### 1.6 Порядок оформления и предъявления заказчику результатов работ по созданию системы (ее частей), по изготовлению и наладке отдельных средств (технических, программных, информационных) и программно-технических (программно-методических) комплексов системы

Предоставляется промежуточная отчетность по завершении каждого установленного заказчиком этапа разработки. Документы предъявляются на бумажных носителях и в электронном виде не позднее установленных сроков. Этапы и сроки сдачи отчетности приведены в таблице 1.1.

## 2 Назначение и цели создания (развития) системы

### 2.1 Назначение системы

Программное обеспечение предназначено для работы с базой данных, содержащей в себе информацию, необходимую для электронной регистрации в поликлинике.

Таблица 1.1 – Этапы разработки

Содержание этапа	Сроки	Отчетный документ
Подготовительный этап. Постановка задачи, сбор и анализ требований к разработке, проработка прототипа ПО, проработка прототипа БД. Разработка технического задания.	1.03 — 21.03	Техническое задание. Прототипы ПО и БД.
Проектирование	21.03 — 14.04	Технический проект. Пересмотренные прототипы. ПО и БД.
Реализация спроектированного приложения и базы данных. Написание программной справки. Тестирование.	14.04 — 02.05	Версия программного продукта.
Определение соответствия, разработанного ПО заданным критериям качества.	02.05 — 21.05	Версия программного продукта. Результаты исследований. Результаты тестирования.
Оформление пояснительной записки. Прием работы.	21.05 — 01.06	Пояснительная записка.

## 2.2 Цели создания системы

Поставлены следующие цели: уменьшить ожидание в очередях к столу регистрации в поликлинике, облегчить процесс регистрации пациентов, обеспечить возможность просматривать и добавлять информацию о сотрудниках (врачах) и вносить изменения в расписание приемов.

## 3 Характеристика объектов автоматизации

### 3.1 Краткие сведения об объекте автоматизации или ссылки на документы, содержащие такую информацию

Объектом автоматизации является база данных, к которой обращается программное обеспечение за чтением или записью данных.



## 4 Требования к системе

### 4.1 Общие требования к системе

#### 4.1.1 Входные и выходные данные

#### 4.1.2 Требования к транспортированию и хранению

#### 4.1.3 Требования к информационной и программной совместимости

#### 4.1.4 Требования к надежности

#### 4.1.5 Требования к эргономике и технической эстетике

#### 4.1.6 Требования к эксплуатации

#### 4.1.7 Требования к маркировке и упаковке

### 4.2 Требования к функциям (задачам), выполняемым системой

#### 4.2.1 Требования к составу выполняемых функций

### 4.3 Требования к видам обеспечения

#### 4.3.1 Требования к техническому обеспечению системы

#### 4.3.2 Требования к информационной и программной совместимости

## 5 Состав и содержание работ по созданию системы

### 5.1 Список исполнителей работ

Руководитель проекта, документатор, программист: Мейта М.В., гр. 722.

### 5.2 Перечень документов, оформленных по ГОСТ 34.201-89, предъявляемых по окончании соответствующих стадий производства

- техническое задание;
- руководство пользователя;
- руководство программиста;
- пояснительная записка к курсовому проекту.

## 6 Порядок контроля и приема системы

### 6.1 Виды, состав, объем и методы испытаний системы и ее составных частей (виды испытаний в соответствии с действующими нормами, распространяющимися на разрабатываемую систему)

Текущий подход к контролю качества подразумевает следующие этапы проекта:

- подсистема готова к демонстрации заказчику;
- подсистема готова к эксплуатации.

## 6.2 Общие требования к приему работ по стадиям (перечень участвующих предприятий и организаций, место и сроки проведения), порядок согласования и утверждения приемочной документации

Прием работ по стадиям осуществляется проверкой отчетных документов по стадиям.

## 6.3 Статус приемной комиссии (государственная, межведомственная, ведомственная)

Кафедральная комиссия. Состав комиссии — научный руководитель: аспирант кафедры КИ-БЭВС Горбунов И. В.

## 7 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

Для работы с системой необходимо переместить директорию программы с CD на жесткий диск компьютера и запустить исполняемый файл `hospital_register.exe`.

## 8 Требования к документированию

### 8.1 Согласованный разработчиком и заказчиком системы перечень подлежащих разработке комплектов и видов документов

- техническое задание;
- руководство пользователя;
- руководство программиста;
- пояснительная записка к курсовому проекту.

### 8.2 Требования к документированию

В структуру технического задания необходимо включить разделы, предусмотренные ГОСТ 34.602-89. Отчетные документы оформляются в соответствии с общими стандартами ТУСУР.

## 9 Источники разработки

1) Давыдова Е.М., Новгородова Н.А., Мещеряков Р.В. Учебно-методические указания к выполнению курсовой работы по дисциплинам «Безопасность систем баз данных» (5 семестр), «Информатика» (5 семестр) — Томск, 2010, 18 с.;

2) ОС ТУСУР 01-2013 — Томск, 2013, 57с.

## РАЗРАБОТАНО

Наименование организации, предприятия	Должность    ис- полнителя	Ф.И.О.	Подпись	Дата
ТУСУР      Каф. КИБЭВС	Руководитель проекта,      до- кументатор, программист	Мейта М.В.		«__»____2015 г.

## СОГЛАСОВАНО

Наименование организации, предприятия	Должность    ис- полнителя	Ф.И.О.	Подпись	Дата
ТУСУР      Каф. КИБЭВС	Аспирант      каф. КИБЭВС	Горбунов И. В.		«__»____2015 г.

## ВВЕДЕНИЕ

В качестве задания на курсовую работу была поставлена задача разработать базу данных и программу пользователя для осуществления электронной регистрации (записи на прием к врачу) в поликлинике.

## 1 Проектирование инфологической модели данных

Инфологическая (концептуальная) модель предметной области представляет собой информационную модель наиболее высокого уровня абстракции и в сущности является как образом реальности, так и образом проектируемой базы данных для этой реальности. Она включает в себя описание информационных объектов или понятий предметной области и связей между ними, а также описание ограничений целостности, т.е. требований к допустимым значениям данных и к связям между ними.

Описание бизнес-процессов в системе электронной регистрации пациентов представлено на диаграммах IDEF0, DFD IDEF3 (рисунки 1.1-1.5).

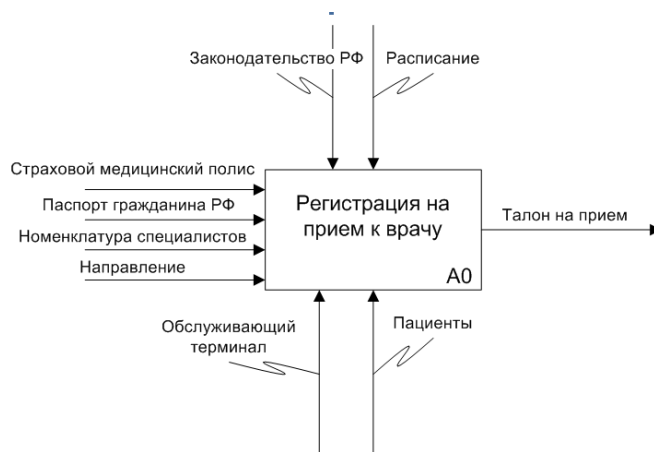


Рисунок 1.1 – «Черный ящик»

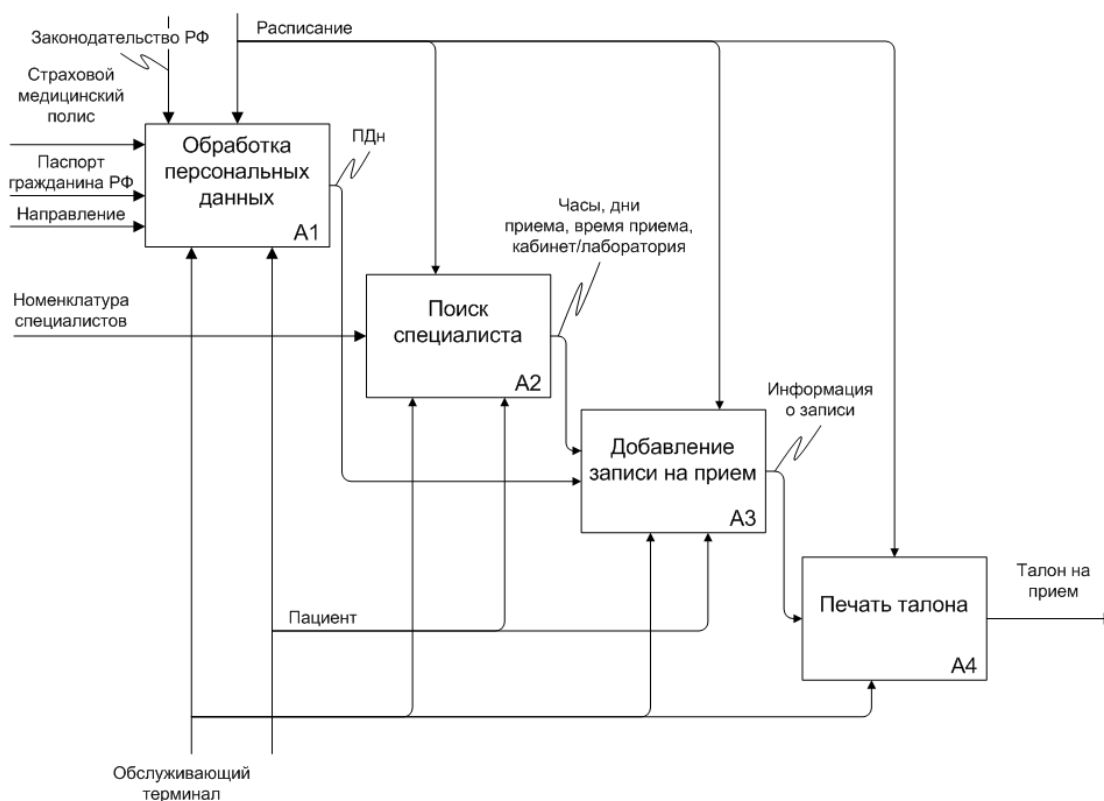


Рисунок 1.2 – Диаграмма IDEF0

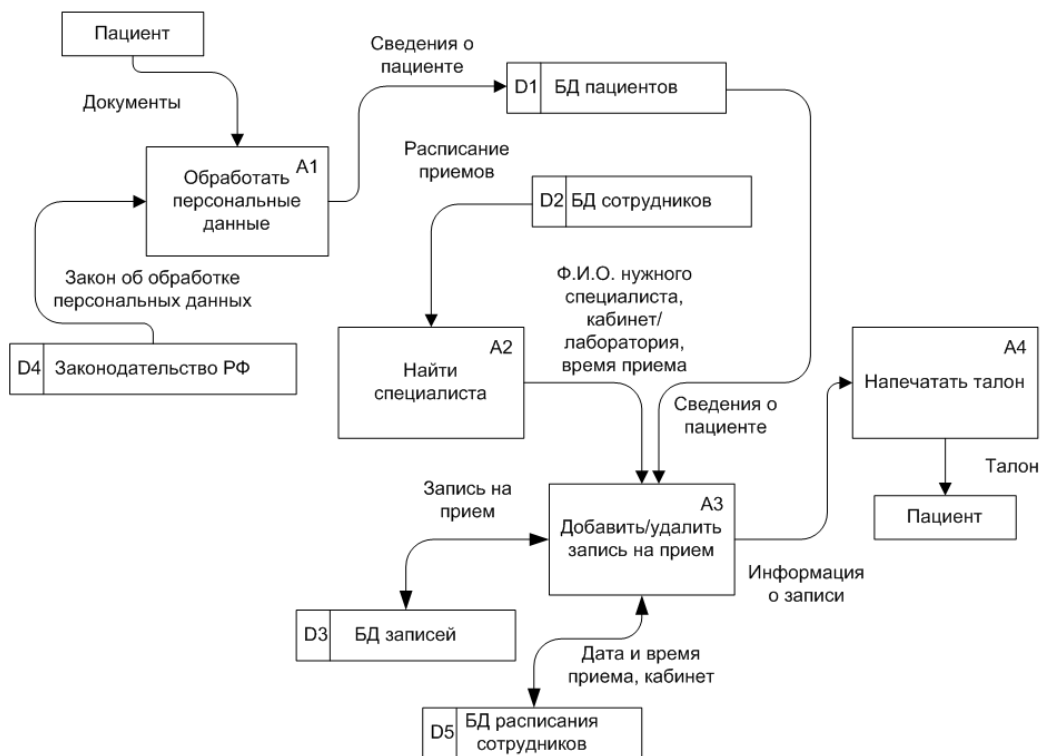


Рисунок 1.3 – DFD-диаграмма бизнес-процессов

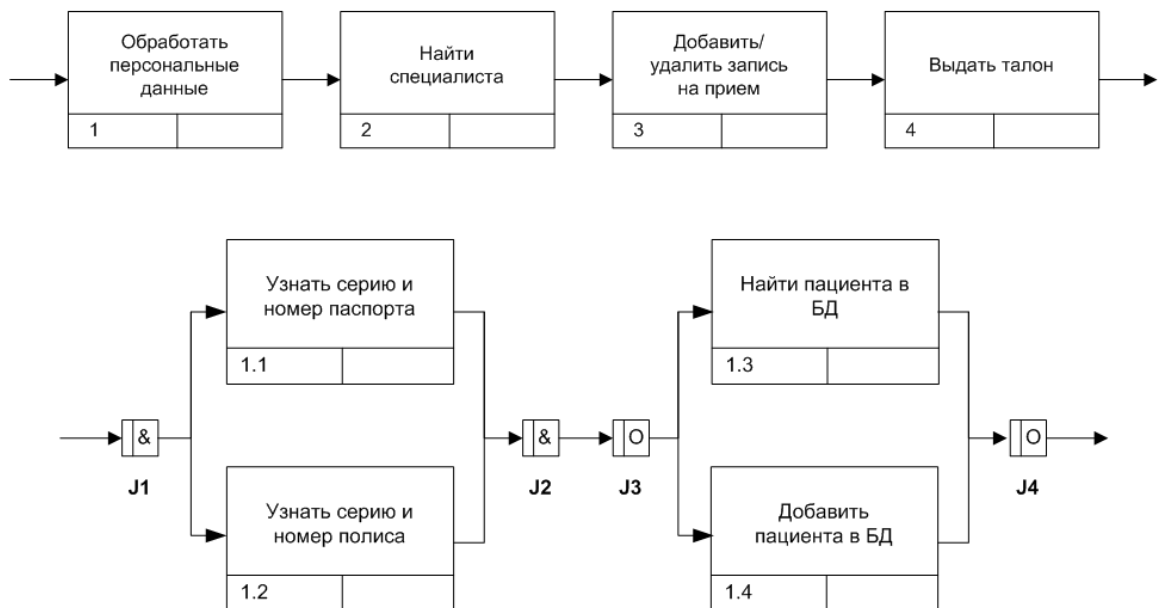


Рисунок 1.4 – Диаграмма IDEF3 (часть 1)

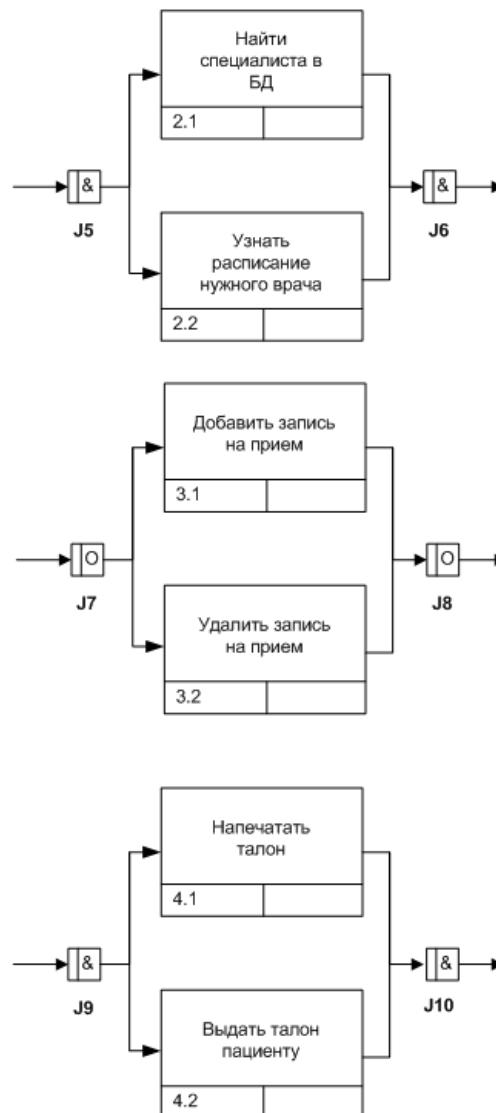


Рисунок 1.5 – Диаграмма IDEF3 (часть 2)

## 2 Проектирование даталогической модели данных

Логическая (даталогическая) модель — это схема базы данных на основе конкретной модели данных, набор схем отношений с указанием первичных ключей, а также «связей» между отношениями, представляющих собой внешние ключи.

Модель «Сущность-связь» (ER-модель) представлена на рисунке 2.1.

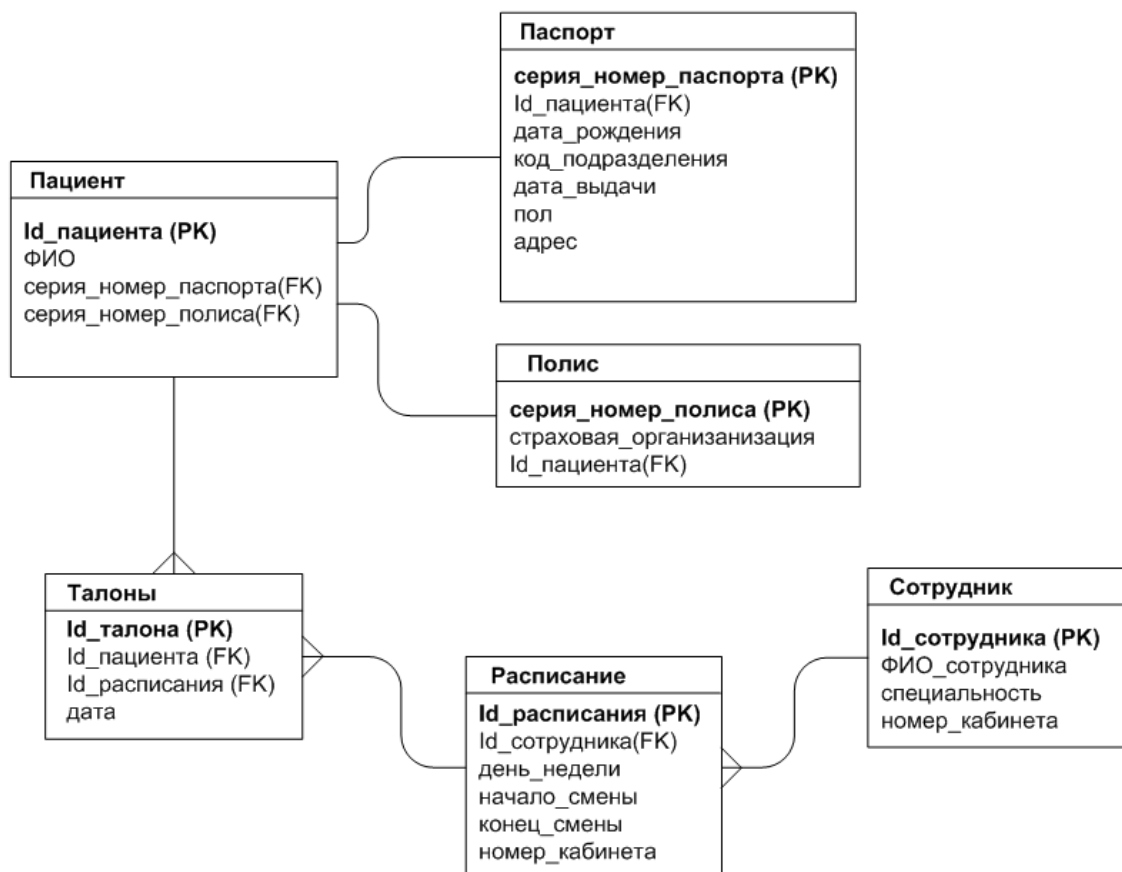


Рисунок 2.1 – Диаграмма IDEF1X (модель «Сущность-связь»)



### 3 Описание базы данных

В данном разделе рассмотрены ограничения, накладываемые на входные данные записей в различных таблицах проектируемой БД, типы входных данных и другие особенности создаваемых таблиц.

#### 3.1 Таблица «patient»

Ограничения на таблицу «patient» (пациент):

- ID пациента не меньше единицы;
- фамилия, имя отчество не должно превосходить 50 символов;
- ни один из атрибутов не должен быть пустым (NULL).

#### 3.2 Таблица «passport»

Ограничения на таблицу «passport» (паспорт):

- ID паспорта не меньше единицы;
- серия паспорта не должна превосходить 4 символа;
- номер паспорта не должен превосходить 6 символов;
- адрес места жительства не должен превосходить 100 символов;
- атрибут «пол» должен состоять из 1 символа (М/Ж);
- ни один из атрибутов не должен быть пустым (NULL).

#### 3.3 Таблица «policy»

Ограничения на таблицу «policy» (полис):

- ID полиса не меньше единицы;
- название страховой медицинской компании не должно превосходить 100 символов;
- ни один из атрибутов не должен быть пустым (NULL).

#### 3.4 Таблица «talon»

Ограничения на таблицу «talon» (талон):

- ID талона не меньше единицы;
- ни один из атрибутов не должен быть пустым (NULL).

#### 3.5 Таблица «timetable»

Ограничения на таблицу «timetable» (расписание):

- ID расписания не меньше единицы;
- день недели должен состоять из 2-ух символов («Пн», «Вт» и т.д.);
- ни один из атрибутов не должен быть пустым (NULL).

#### 3.6 Таблица «employee»

Ограничения на таблицу «employee» (сотрудник):

- ID сотрудника не меньше единицы;
- специальность сотрудника не должна превосходить 50 символов;
- фамилия, имя отчество не должно превосходить 50 символов;
- номер рабочего кабинета должен состоять из 3 символов;
- ни один из атрибутов не должен быть пустым (NULL).

### 3.7 Роли базы данных

SQLite является встраиваемой реляционной базой данных и 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. Таким образом, в качестве протокола обмена используются вызовы функций (API) библиотеки SQLite. Такой подход уменьшает накладные расходы, время отклика и упрощает программу. SQLite хранит всю базу данных (включая определения, таблицы, индексы и данные) в единственном стандартном файле на том компьютере, на котором выполняется программа.[2]

В SQLite отсутствует разграничение ролей как таковых, поэтому потребовалось вводить меры по защите БД непосредственно в программном приложении для работы с базой. На уровне приложения были условно созданы 2 роли: «администратор» и «пациент». «Администратор» при вводе пароля (хэш от пароля сравнивается с хэшем, хранимым в БД) получает доступ к программному изменению и просмотру данных в базе. «Пациент» имеет свободный доступ к форме для записи на прием и регистрации.

### 3.8 Защита от SQL-инъекций

Поскольку база данных SQLite является встраиваемой реляционной базой данных и представляет собой файл, провести SQL-инъекцию невозможно. Однако необходимо ограничить доступ к базе на уровне файловой системы, чтобы исключить возможность несанкционированного доступа (копирования, изменения, удаления данных или самой базы).

## 4 Описание процесса деятельности

### 4.1 Постановка задачи

База данных и программа «hospital\_register» создается для внедрения в поликлиниках в качестве электронной регистратуры.

### 4.2 Описание данных программы

#### 4.2 Входные данные

В программе «hospital\_register» возможны следующие входные данные: Ф.И.О. сотрудника, специальность сотрудника, день записи, серия и номер паспорта пациента, номер кабинета, день недели, ID сотрудника, начало и конец смены сотрудника, ID пациента, ID расписания и ID талона, Ф.И.О. пациента, день рождения пациента, месяц рождения пациента, год рождения пациента, пол пациента, код подразделения, адрес места жительства пациента, серия полиса пациента, номер полиса пациента, страховая компания. Далее приведен диапазон допустимых значений. Все входные значения не должны быть пустыми (NULL).

В главном окне программы MainWindow возможен только один входной параметр — пароль администратора. На него заведомо накладывается только одно ограничение — пароль не должен быть пустой строкой.

В окне управления базой AdminWindow входные параметры следующие: Ф.И.О. сотрудника, специальность сотрудника, номер кабинета, день недели, ID сотрудника, начало и конец смены сотрудника, ID пациента, ID расписания и ID талона (таблица 4.2).

В окне для записи на прием к врачу EnrollWindow входные параметры: Ф.И.О. сотрудника, специальность сотрудника, день записи, серия и номер паспорта пациента (таблица 4.3).

В окне регистрации пациента PatientRegisterWindow входные параметры: Ф.И.О. пациента, день рождения пациента, месяц рождения пациента, год рождения пациента, пол пациента, серия паспорта пациента, номер паспорта пациента, код подразделения, адрес места жительства пациента, серия полиса пациента, номер полиса пациента, страховая компания (таблица 4.4).

#### 4.2 Выходные данные

Выходные данные сохраняются в PDF-файл «talon.pdf» в рабочей директории программы. Данный файл в дальнейшем может быть распечатан терминалом при завершении операции записи пациента.

В файл «talon.pdf» выводятся следующие данные: Ф.И.О. врача, специальность, номер кабинета, рабочие часы, дата и день недели, Ф.И.О. пациента и страховой медицинский полис пациента.

Таблица 4.2 – Таблица признаков для AdminWindow

Наименование признака	Описание признака	Диапазон допустимых значений
employee_name	Ф.И.О. сотрудника	Символы русского или латинского алфавита
speciality	Специальность сотрудника	Символы русского или латинского алфавита
office_number	Номер кабинета сотрудника	3 цифры от 0 до 9
week_day	День недели	Одно из заранее определенных значений («Пн», «Вт», «Ср», «Чт», «Пт», «Сб», «Вс»)
employee_id	Идентификатор сотрудника	10 цифр от 0 до 9
patient_id	Идентификатор пациента	10 цифр от 0 до 9
timetable_id	Идентификатор расписания	10 цифр от 0 до 9
talon_id	Идентификатор талона	10 цифр от 0 до 9
shift_begining	Начало смены сотрудника	Строка формата «ЧЧ:ММ», где ЧЧ (часы) и ММ (минуты) — цифры от 0 до 9
shift_ending	Конец смены сотрудника	Строка формата «ЧЧ:ММ», где ЧЧ (часы) и ММ (минуты) — цифры от 0 до 9

Таблица 4.3 – Таблица признаков для EnrollWindow

Наименование признака	Описание признака	Диапазон допустимых значений
employee_name	Ф.И.О. сотрудника	Одно из заранее определенных значений
speciality	Специальность сотрудника	Одно из заранее определенных значений
week_day	День записи	Одно из заранее определенных значений («Пн», «Вт», «Ср», «Чт», «Пт», «Сб», «Вс»)
passport	Серия паспорта пациента	4 цифры от 0 до 9
passport	Номер паспорта пациента	6 цифр от 0 до 9

Таблица 4.4 – Таблица признаков для EnrollWindow

Наименование признака	Описание признака	Диапазон допустимых значений
patient_name	Ф.И.О. сотрудника	Символы русского или латинского алфавита
birth_date	День рождения пациента	Число от 1 до 31
birth_date	Месяц рождения пациента	Одно из заранее определенных значений
birth_date	Год рождения пациента	Число от 1900 до 2020
sex	Пол пациента	Одно из заранее определенных значений («М» или «Ж»)
passport	Серия паспорта пациента	4 цифры от 0 до 9
passport	Номер паспорта пациента	6 цифр от 0 до 9
issue_date	День выдачи паспорта	Число от 1 до 31
issue_date	Месяц выдачи паспорта	Одно из заранее определенных значений
issue_date	Год выдачи паспорта	Число от 1900 до 2020
policy	Серия полиса пациента	5 цифр от 0 до 9
policy	Номер полиса пациента	6 цифр от 0 до 9
insurance_agency	Название страховой медицинской компании	Символы русского или латинского алфавита

### 4.3 Основные технические решения

#### 4.3 Алгоритм

Предварительного заполнения базы данных не требуется, однако для возможности добавления записи и регистрации пациента администратору сначала необходимо заполнить базу сотрудников и расписаний.

##### Шаг 1

При авторизации от имени пациента в главном окне программы (MainWindow) появится окно для записи на прием к врачу (EnrollWindow). Необходимо выбрать нужного специалиста, найти его расписание и правильно заполнить предлагаемую форму. Если входные значения были введены неверно, появится сообщение об ошибке.

##### Шаг 2

Программа проверяет, есть ли уже в БД введенные пользователем серия и номер паспорта. Если данные о пациенте в базе уже есть и форма заполнена верно, появится сообщение об успешной записи, либо сообщение об ошибке записи, если такая запись уже была добавлена в БД.

##### Шаг 3

Если данных о пациенте в базе нет, то появится форма регистрации пациента (PatientRegisterWindow). Если все поля формы заполнены верно, пациент будет добавлен в базу и в окне записи на прием при повторной операции записи появится уведомление об успешном завершении операции, в результате которой будет сгенерирован PDF-файл (талон), готовый к печати.

##### Шаг 4

При авторизации от имени администратора программа сверяет хеш введенного пароля с хешем, который хранится в БД. Если они совпадают, то открывается окно управления базой данных (AdminWindow), если нет, то появляется уведомление об ошибочной авторизации.

##### Шаг 5

Окно управления базой данных (AdminWindow) позволяет администратору просматривать, удалять и добавлять данные в базу. Для этого необходимо вводить корректные данные, в противном случае операции удаления и добавления выполняться не будут, появится сообщение об ошибке ввода или обращения к базе.

Блок-схема алгоритма работы программы представлена на рисунке 4.1.

Рисунок 4.1 – Блок-схема алгоритма работы программы

### 4.3 Численность, функции и квалификация персонала

Для использования системы необходим администратор, который будет добавлять новых сотрудников и менять расписание уже добавленных, следить за целостностью системы, удалять ПДн пациентов по мере необходимости.

### 4.3 Обеспечение потребительских характеристик системы

Надежность обеспечивается путем следования стандартам написания кода и использования блоков try-catch для обработки исключительных ситуаций. Производительность системы обеспечивается путем использования оптимальных алгоритмов.

### 4.3 Функции, выполняемые системой

Функциями, выполняемыми программой «hospital\_register», являются регистрация пациентов, добавление записей на прием, печать талонов, управление расписанием и номенклатурой сотрудников (врачей), а также управление записями в БД.

### 4.3 Комплекс технических средств

Для функционирования системы необходимы следующие аппаратные средства:

- ОС Linux Ubuntu/Debian;
- процессор x86-архитектуры;
- объем ОЗУ для выполнения программы: не менее 300Мб;
- объём видеопамяти: не менее 300Мб;
- память на жестком диске: не менее 100Мб для файлов БД и программы;
- монитор с разрешением 800х600 или выше;
- мышь, клавиатура;
- устройство для чтения CD;
- принтер.

### 4.3 Информационное обеспечение системы

Система поставляется с руководством пользователя и программиста.

### 4.3 Программное обеспечение системы

Система разворачивается на компьютере с ОС Linux Ubuntu/Debian.

## 5 Мероприятия по подготовке персонала

Провести ознакомление персонала с руководством пользователя.

## 6 Руководство пользователя



## 7 Руководство программиста

## 8 Перспективы применения программы

## ЗАКЛЮЧЕНИЕ

В дальнейшем программу следует доработать следующим образом: сконфигурировать и отладить решение для успешного запуска программы на ОС Windows, добавить автоматический вывод на печать сгенерированного талона, доработать адаптивный интерфейс программы (изменить контейнеры в GTK-окнах), оптимизировать программный код. Можно также расширить возможности программы для добавления талонов на конкретное время, автоматическое удаление устаревших записей и др.

## Список использованных источников

- 1 Mono Documentation [Электронный ресурс] // docs.go-mono.com:[сайт]. [2015]. URL: <http://docs.go-mono.com/index.aspx?link=T.Gtk.Entry>.
- 2 SQLite Documentation [Электронный ресурс] // www.sqlite.org:[сайт]. [2015]. URL: <https://www.sqlite.org/docs.html>.
- 3 iTextSharp, a .NET PDF library [Электронный ресурс] // www.sourceforge.net:[сайт]. [2015]. URL: <http://sourceforge.net/projects/itextsharp/?source=typ<sub>r</sub>edirect>.

Приложение А  
(Обязательное)  
Компакт-диск

Компакт-диск содержит:

- электронную версию пояснительной записки в форматах \*.tex и \*.pdf;
- актуальную версию клиентской программы с графическим интерфейсом;
- базу данных, содержащую тестовые данные.