

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ
Н.Г.ЧЕРНЫШЕВСКОГО»

Кафедра дискретной математики и
информационных технологий

Модернизация программного компонента ИС «Экстренные извещения»

АВТОРЕФЕРАТ ДИПЛОМНОЙ РАБОТЫ

студента 6 курса 621 группы

направления 230101 «Вычислительные машины, комплексы, системы и сети»

факультета компьютерных наук и информационных технологий

Лысковцева Сергея Сергеевича

Научный руководитель

ассистент кафедры ДМиИТ

М.В. Карандашов

подпись, дата

Зав. кафедрой

к. ф.-м.н., доцент

Л.Б. Тяпаев

подпись, дата

Саратов 2016

ВВЕДЕНИЕ

Несмотря на то, что современные информационные системы (далее, ИС) совершенствуются с каждым годом, все еще встречаются их программные компоненты 20 летней давности. Эти программы еще работают, однако их функционал и совместимость с современными операционными системами перестают удовлетворять современным требованиям. Это можно объяснить развитием технологий и появлением задач, которые не ставились перед авторами данных программ во время разработки. В качестве примера можно привести задачи, связанные с обменом данными через сетевые протоколы. В такие программы обычно затруднен процесс интеграции новых модулей, это может быть обусловлено, отсутствием необходимых технологий в среде разработки. Так же проблемной частью обычно является интерфейс пользователя и составление отчетов. Учитывая все выше сказанное, рано или поздно возникает задача обновления программной части ИС.

В дипломной работе рассмотрен процесс модернизации программы, являющейся частью ИС, предназначенной для хранения данных о больных по учетной форме № 089/у-кв «Извещение о больном с вновь установленным диагнозом сифилиса, гонореи, трихомониаза, хламидиоза, герпеса уrogenитального, аногенитальными бородавками, микроспории, фавуса, трихофитии, микоза стоп, чесотки». Вышеупомянутая ИС используется врачами Саратовской области для сбора, хранения и анализа информации о людях, болеющих заболеваниями передаваемыми половым путем. Регистрация и учет половых инфекций необходимы для создания государственных программ для борьбы с этими заболеваниями. Статистические данные собираемые этой ИС позволяют прогнозировать эпидемиологическую обстановку и целенаправленно проводить мероприятия по снижению заболеваемости. Целью работы является модернизация программной части ИС «Экстренные извещения» для соответствия современным требованиям документооборота, так и программно-аппаратному оснащению медицинских учреждений.

Основная часть работы состоит из 4 глав:

1. Постановка задачи и обзор существующей программы
2. Разработка новой БД и перенос данных
3. Создание клиентского приложения
4. Реализация механизмов идентификации, аутентификации и авторизации

В первой части дипломной работы рассмотрена существующая программа и на основе ее анализа, а так же новых требований к разрабатываемой программе построена схема данных. Исходя из разработанной схемы данных, выработаны требования к системе управления базами данных (далее, СУБД).

Во второй части рассмотрены современные популярные СУБД, удовлетворяющие разработанным требованиям и выбрана наиболее подходящая, после чего рассмотрен процесс переноса данных в новую СУБД.

Процесс выбора библиотеки для реализации графического интерфейса пользователя, а так же сама реализация рассмотрены в третьей части дипломной работы.

В четвертой части рассмотрена реализация механизмов идентификации, аутентификации и авторизации.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Постановка задачи и обзор существующей программы

ИС «Экстренные извещения» используется медицинскими учреждениями Саратовской области для регистрации данных по форме № 089/у-кв и построения статистических отчетов. Программная часть этой ИС была разработана в 1994 году. Фирма ISPSoft, реализовавшая имеющуюся программу, на данный момент прекратила свою деятельность, исходного кода программы нет. В программе используется архитектура файл сервер и встраиваемая версия СУБД FoxPro. Форматирование отчетов реализовано с помощью символов псевдографики.

В связи с отсутствием исходного кода программы было принято решение разработать программу дублирующую и расширяющую имеющиеся функции. Т.к. СУБД FoxPro встроена в программу, которую решено переписать, а данные хранящиеся в ней нужны для построения отчетов за предыдущие периоды, то принято решение перенести эти данные на новую СУБД. Так же в связи с вступлением в силу Приказа №305 от 19 августа 2015 года «Об утверждении Методических рекомендаций по использованию свободного программного обеспечения в деятельности федеральных органов исполнительной власти, включая критерии определения государственных информационных систем, при создании которых необходимо использовать свободное программное обеспечение, в том числе государственных информационных систем, предназначенных для оказания государственных и муниципальных услуг в электронном виде», было добавлено требование кроссплатформенности и об открытом исходном коде, как разрабатываемой программы, так и СУБД.

Разработка новой БД

Существующая версия программы хранит данные, используя СУБД FoxPro. Данная СУБД не удовлетворяет обновленным требованиям, а именно она имеет закрытый исходный код, имеет файловую архитектуру хранения данных. Следовательно, необходимо реализовать функционал программы,

используя новую СУБД, сохранив имеющиеся данные. Выбор СУБД следует делать на основе схемы данных.

Разработка схемы БД

Т.к. исходный код программы отсутствует, а FoxPro хранит каждую таблицу в отдельном файле, можно судить о структурной схеме базы данных (далее БД) по этим файлам. В существующей версии программы данные хранились в 13 таблицах, одна из которых хранила непосредственно записи, а остальные служили справочниками. В целом подобная схема и была использована далее. Однако она была дополнена под текущие требования.

Исходя из разработанной схемы данных, а также из постановки задачи заказчиком, были разработаны следующие требования, которым должна удовлетворять СУБД:

1. Реляционность;
2. Кроссплатформенность;
3. Открытый исходный код (Open Source);
4. Архитектура клиент-сервер.

Требования к архитектуре были составлены исходя из требования медицинского учреждения о централизованном хранении информации.

Выбор СУБД

После разработки схемы необходимо определиться с СУБД, с помощью которой будет реализована разработанная схема.

В таблице 1 приведен рейтинг популярности реляционных СУБД издания DB-Engines за май 2016 года.

Таблица 1 – Рейтинг популярности реляционных СУБД

Ранг	СУБД	Рейтинг	Open Source	Кроссплатформенность	Клиент серверная архитектура
1	Oracle	1462,02	-	+	+
2	MySQL	1371,83	+	+	+
3	Microsoft SQL Server	1142,82	-	-	+
4	PostgreSQL	307,61	+	+	+
5	DB2	185,96	-	+	+
6	Microsoft Access	131,58	-	-	-
7	SQLite	107,26	+	+	-
8	Teradata	73,74	-	-	+
9	SAP Adaptive Server	71,48	-	+	+
10	Hive	47,51	+	+	+
11	FileMaker	46,71	-	+	+
12	SAP HANA	41,37	-	-	+
13	MariaDB	33,97	+	+	+
14	Informix	30,58	-	+	+
15	Firebird	19,9	+	+	+

Из представленных СУБД под требования подходят лишь MySQL, PostgreSQL, Hive, MariaDB и Firebird.

После анализа вышеупомянутых СУБД было решено перенести имеющуюся базу данных на СУБД MySQL. На данный выбор более всего повлияло количество информации на русском языке о данной СУБД, методах работы и переноса данных.

Перенос данных в новую СУБД

Перенос данных из файлов формата DBF является достаточно тривиальной задачей, особенно популярна эта задача была в начале 2000-х годов, когда происходила массовая миграция на клиент серверные СУБД. Существует множество программ для решения этой задачи

В качестве примера можно привести программу DBF Converter. С помощью этой программы была перенесена основная таблица БД, далее она была дополнена в соответствии с новой схемой данных. Таблицы справочники были переписаны заново, учитывая новые требования. Данные в основной

таблице были дополнены и изменены в соответствии с новыми справочниками с помощью SQL запросов на обновление.

Создание клиентского приложения

После реализации БД необходимо написать программу клиент, в которой будет реализован графический интерфейс для ввода информации и построения отчетов.

Выбор среды разработки

В процессе постановки задачи одно из требований заключалось в кроссплатформенности разрабатываемого приложения, а именно в возможности запуска приложения на ОС семейства Windows и Linux. Учитывая это требование, а также мой опыт разработки на C++, выбор средств разработки был ограничен кроссплатформенными фреймворками поддерживающих язык C++. После проведения предварительного анализа средств разработки было выделено несколько наиболее популярных решений в этой области, а именно: GTK+, wxWidgets, FLTK, Juce, Qt.

Для выбора оптимального решения был произведен сравнительный анализ фреймворков по таким параметрам как: наличие готовых (встроенных) классов для работы с SQL, наличие готовых драйверов для работы с MySQL, наличие документации на русском языке. Результаты анализа представлены в таблице 2.

Таблица 2 – Сравнительный анализ выбранных фреймворков

Фреймворк	Классы для работы с SQL	Скомпилированные драйвера для MySQL	Документация на русском языке
GTK+	Да [15]	Нет	Да (Единичные экземпляры)
wxWidgets	Да [16]	Нет	Да (Единичные экземпляры)
FLTK	Нет	Нет	Да (Единичные экземпляры)
Juce	Нет	Нет	Да (Единичные экземпляры)
Qt	Да [17]	Да [18]	Да

Как видно из таблицы 1, наиболее оптимальной средой разработки является фреймворк Qt, поэтому именно он был использован для разработки данного клиентского приложения.

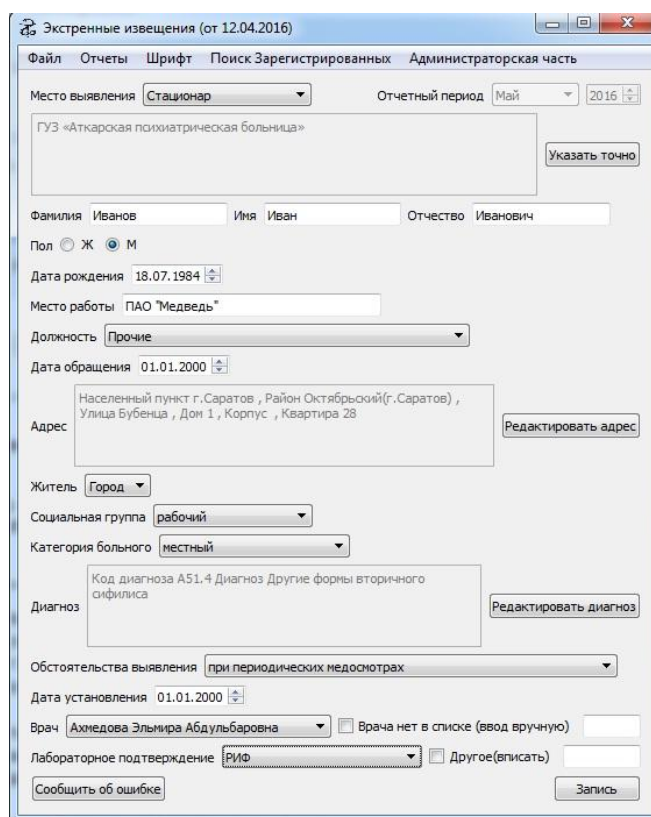
Создание интерфейса пользователя

После выбора инструмента разработки программы необходимо реализовать интерфейс пользователя на основе существующей программы. Для этого необходимо создать формы для ввода информации, построения отчетов и редактирования записей. Так же при создании этих форм необходимо учитывать новые требования к объему вводимой информации.

Создание формы ввода извещений

Основной формой для работы с программой является форма ввода извещений. Именно с этой формой работают операторы, заполняющие БД. Разработана форму, дублирующая существующий функционал, а так же дополнена новыми полями.

Конечный вид формы для ввода данных представлен на рисунке 1. Все показанные данные не являются подлинными.



Экстренные извещения (от 12.04.2016)

Место выявления: Стационар Отчетный период: Май 2016

ГУЗ «Аткарская психиатрическая больница»

Фамилия: Иванов Имя: Иван Отчество: Иванович

Пол: Ж М

Дата рождения: 18.07.1984

Место работы: ПАО «Медведь»

Должность: Прочие

Дата обращения: 01.01.2000

Адрес: Населенный пункт г.Саратов, Район Октябрьский(г.Саратов), Улица Бубенца, Дом 1, Корпус, Квартира 28

Житель: Город

Социальная группа: рабочий

Категория больного: местный

Диагноз: Код диагноза A51.4 Диагноз Другие формы вторичного сифилиса

Обстоятельства выявления: при периодических недосмотрах

Дата установления: 01.01.2000

Врач: Ахмедова Эльмира Абдульбаровна Врача нет в списке (ввод вручную)

Лабораторное подтверждение: РИФ Другое(вписать)

Сообщить об ошибке Запись

Рисунок 1 – Внешний вид формы для ввода данных

Создание формы просмотра отчетов

В разработанной программе отчеты отображаются с помощью языка разметки HTML. Данный подход был реализован для обеспечения кроссплатформенности и независимости от стороннего программного обеспечения, например комплектов программ MS Office или Open Office.

Была разработана универсальная форма, которая позволяет конструировать отчеты на основе имеющихся путем добавления к ним дополнительных условий.

Функционал конструктора запросов в клиентском приложении реализован на основе динамических SQL запросов. Динамический SQL запрос – это SQL код, который создается и сохраняется в переменной, пока не возникнет необходимость его выполнения. Программно динамические отчеты

могут быть реализованы как на стороне сервера, с помощью хранимых процедур, так и непосредственно в клиенте программы. Однако последний подход подразумевает разрешение на выполнение запросов со стороны программы клиента, в то время как разрешение выполнения исключительно процедур повышает безопасность работы с СУБД.

На рисунке 2 представлен пример формы с выведенным отчетом.

Рисунок 2 – Форма с выведенным отчетом

Создание формы для редактирования записи

Для редактирования записей, а так же просмотра журнала обращений была разработана форма, внешний вид которой с подписью групп основных виджетов представлен на рисунке 3.

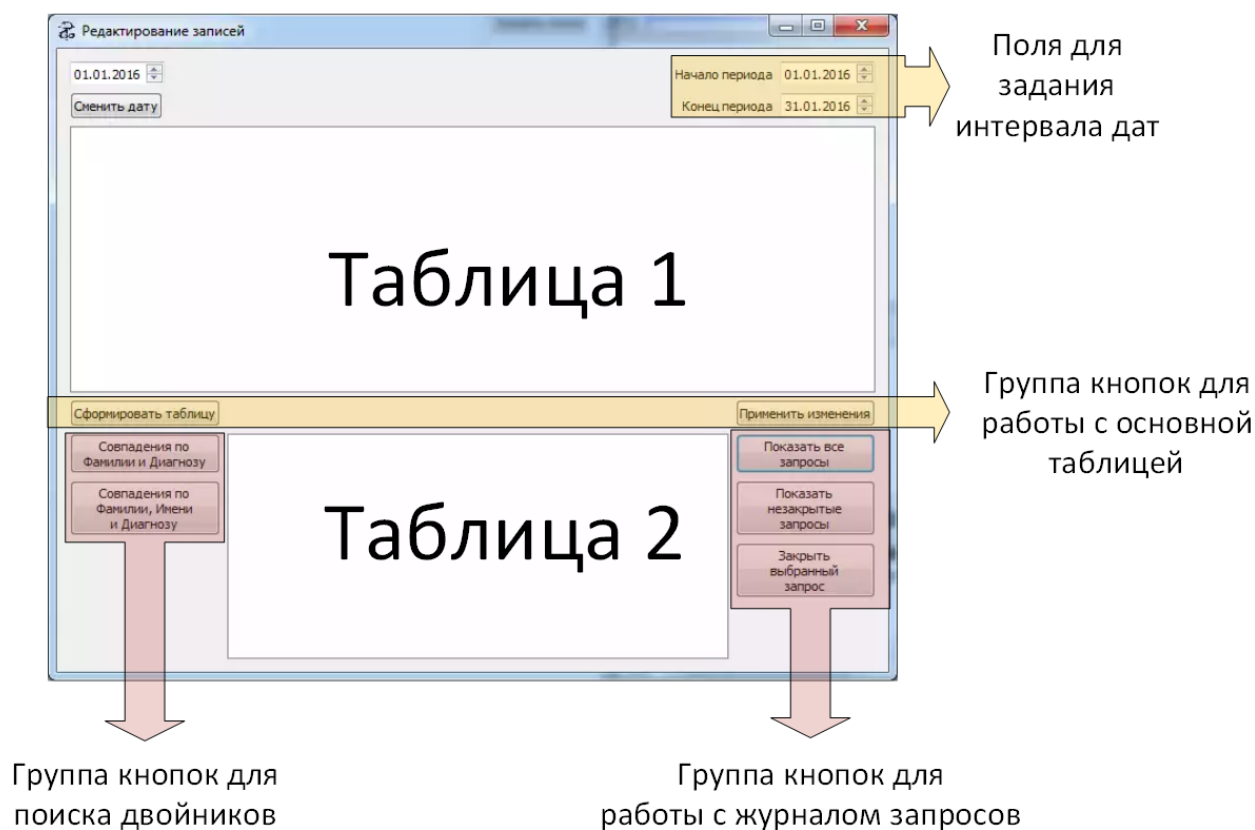


Рисунок 3 – Основные группы виджетов на форме.

Как видно из рисунка 3 на форме присутствуют две таблицы. Таблица 1 используется для отображения и редактирования записей основной таблицы. Таблица 2 служит для отображения вспомогательной информации, такой как данные о дублирующихся записях и отображения журнала запросов. Поля для задания интервала дат учувствуют в составлении запросов для отображения основной таблицы и таблицы дубликатов. Группы кнопок служат для действий в соответствии с их подписями.

Реализация механизмов идентификации, аутентификации и авторизации

Идентификацию и аутентификацию можно считать основой программно-технических средств безопасности.

Именно от корректности решения этих двух задач (распознавания и проверки подлинности) зависит, можно ли разрешить доступ к ресурсам системы конкретному пользователю

В роли сервера авторизации выступает сервер СУБД MySQL.

После запуска программы происходит примерно следующее:

1. Сервер проверяет SSL сертификат клиента и если он был подписан корневым центром сертификации, сертификат которого в свою очередь установлен на сервере, тогда устанавливается защищенное шифрованием соединение между клиентом и сервером.

Возможные ошибки: неподписанный сертификат. Результат: запрет доступа.

2. Клиент отправляет на сервер связку логин/пароль для доступа к серверу авторизации, эта связка хранится непосредственно в программе. После этого программа/пользователь получает права на выполнение 2-х процедур.

Процедура list возвращает список зарегистрированных в ИС сотрудников. Этот список выводится пользователю. Пользователь выбирает из списка нужную запись и вводит пароль. Пароль шифруется одной из хеш-функций (в данный момент это – sha1), к получившейся строке добавляется мусор и эта строка отправляется на сервер с помощью 2-ой доступной процедуры.

Процедура login принимает на входе 2 аргумента, ИД выбранного сотрудника и строку полученную из хэша пароля и мусора. Если хэш пароля совпадает с сохраненным в таблице пользователей, тогда процедура возвращает логин и пароль подходящий для этого типа пользователей зашифрованные симметричным алгоритмом функций (в данный момент это aes128) с ключом сформированным на основе полученного хэша.

Возможные ошибки: неправильно введен пароль пользователя

Результат: процедура login вернет ошибку.

3. После получения пары логин/пароль клиентская программа разрывает соединение с сервером, расшифровывает полученные значения и подключается в СУБД, используя полученную пару логин/пароль

На рисунке 4 показана модель взаимодействия программы с сервером в зашифрованном канале связи.



Рисунок 4 – Модель взаимодействия программы клиента и сервера авторизации.

ЗАКЛЮЧЕНИЕ

В дипломной работе был рассмотрен процесс модернизации существующей программы являющейся частью ИС «Экстренные извещения». В ходе модернизации было решено переписать существующую программу с нуля, разделив БД и программу для ее заполнения.

БД была дополнена в соответствии с современными требованиями и перенесена на СУБД с клиент-серверной архитектурой, что позволило реализовать централизованное хранение информации. Это облегчило процедуру передачи данных из территориально отдаленных районов, а так же повысило степень актуальности выдаваемых данных.

Использование кроссплатформенной СУБД с открытым исходным кодом позволило выполнить требования Приказа Минкомсвязи России №305 от 19 августа 2015 года «Об утверждении Методических рекомендаций по использованию свободного программного обеспечения в деятельности федеральных органов исполнительной власти, включая критерии определения государственных информационных систем, при создании которых необходимо использовать свободное программное обеспечение, в том числе государственных информационных систем, предназначенных для оказания государственных и муниципальных услуг в электронном виде»

Использование кроссплатформенного фреймворка при разработке приложения позволило скомпилировать версии программы, работающие на ОС семейств Linux и Windows.

Разработка приложения с нуля позволила переделать интерфейс пользователя, сделав его более удобным, а так же корректно отображающимся на современных ОС. Так же были добавлены необходимые поля и отчеты в соответствии с современными требованиями.

Внедрение в программу механизма авторизации позволило повысить безопасность хранения данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. DB-Engines[Электронный ресурс] : информационный сайт. Статья «DB-Engines Ranking of Relational DBMS, May 2016 » URL: <http://db-engines.com/en/ranking/relational+dbms> Загл. с экрана. Яз. Англ.
2. Википедия [Электронный ресурс] : свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike 3.0 Unported ; Wikimedia Foundation, Inc., некоммерческой организации. Wikipedia® Статья «MySQL» URL: <http://en.wikipedia.org/wiki/MySQL> Загл. с экрана. Последнее изменение 07:54, 19 мая 2015. Язык Англ.
3. Википедия [Электронный ресурс] : свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike 3.0 Unported ; Wikimedia Foundation, Inc., некоммерческой организации. Wikipedia® Статья «PostgreSQL» URL: <https://ru.wikipedia.org/wiki/PostgreSQL> Загл. с экрана. Последнее изменение 07:43, 27 апреля 2015. Язык Рус.
4. Официальный сайт PostgreSQL[Электронный ресурс] Статья « PostgreSQL 9.2.11 Documentation Chapter 8. Data Types 8.14. JSON Type» URL: <http://www.postgresql.org/docs/9.2/static/datatype-json.html> Загл. с экрана. Язык Англ.
5. Портал Национальной библиотеки им. Н. Э. Баумана [Электронный ресурс] : информационный портал. Статья «Apache Hive» URL: http://ru.bmstu.wiki/Apache_Hive Загл. с экрана. Последнее изменение 20:47, 20 декабря 2015. Яз. Рус.
6. Официальный сайт MariaDB [Электронный ресурс] Статья « The MongoDB 3.0 Manual» URL: <http://docs.mongodb.org/manual/t> Загл. с экрана. Язык Англ
7. Википедия [Электронный ресурс] : свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike 3.0 Unported ; Wikimedia Foundation, Inc., некоммерческой организации. Wikipedia® Статья «Firebird» URL: <https://ru.wikipedia.org/wiki/Firebird> Загл. с экрана. Последнее изменение 06:44, 11 мая 2016. Язык Рус.

8. DBF Viewer2000 [Электронный ресурс] сайт. Статья «DBF to SQL» » URL: <http://www.dbf2002.com/dbf-converter/convert-dbf-to-sql.html> Загл. с экрана. Язык Англ
9. Википедия [Электронный ресурс] : свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike 3.0 Unported ; Wikimedia Foundation, Inc., некоммерческой организации. Wikipedia® Статья «GTK+» URL: https://ru.wikipedia.org/wiki/GTK%2B#cite_note-mainpage-6 Загл. с экрана. Последнее изменение 17:45, 17 февраля 2016. Язык Рус.
10. Официальный сайт The GNOME Project [Электронный ресурс] Статья «GTK+ 2.18.0 released» URL: <https://mail.gnome.org/archives/gtk-devel-list/2009-September/msg00054.html> Загл. с экрана. Язык Англ.
11. Википедия [Электронный ресурс] : свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike 3.0 Unported ; Wikimedia Foundation, Inc., некоммерческой организации. Wikipedia® Статья «wxWidgets» URL: <https://ru.wikipedia.org/wiki/WxWidgets> Загл. с экрана. Последнее изменение 13:22, 2 февраля 2016. Язык Рус.
12. Википедия [Электронный ресурс] : свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike 3.0 Unported ; Wikimedia Foundation, Inc., некоммерческой организации. Wikipedia® Статья «FLTK» URL: <https://ru.wikipedia.org/wiki/FLTK> Загл. с экрана. Последнее изменение 13:27, 3 июня 2015. Язык Рус.
13. Википедия [Электронный ресурс] : свободная энциклопедия / текст доступен по лицензии Creative Commons Attribution-ShareAlike 3.0 Unported ; Wikimedia Foundation, Inc., некоммерческой организации. Wikipedia® Статья «Juce» URL: <https://ru.wikipedia.org/wiki/Juce> Загл. с экрана. Последнее изменение 08:58, 18 ноября 2015. Язык Рус.
14. Викиучебник [Электронный ресурс] Текст доступен по лицензии Creative Commons Attribution-ShareAlike Статья «Qt» URL: <https://ru.wikibooks.org/wiki/Qt> Загл. с экрана. Последнее изменение 06 февраля 2016 Язык Рус.

15. Сайт library.gnome.org [Электронный ресурс] Статья «SQL parser» URL: <https://developer.gnome.org/libgda/stable/libgda-provider-parser.html> Загл. с экрана. Язык Англ.
16. Сайт <http://wxwidgets.info/> [Электронный ресурс] Статья «Работа с базами данных в wxWidgets – DatabaseLayer – Минимальное приложение» URL: <http://wxwidgets.info/rabota-s-bazami-dannyx-v-wxwidgets-databaselayer-minimalnoe-prilozhenie/> Загл. с экрана. Язык Рус.
17. Сайт <http://doc.qt.io/> [Электронный ресурс] Статья « QSqlQuery Class» URL: <http://doc.qt.io/qt-4.8/qsqquery.html> Загл. с экрана. Язык Англ.
18. Сайт <http://doc.qt.io/> [Электронный ресурс] Статья « SQL Database Drivers» URL: <http://doc.qt.io/qt-5/sql-driver.html> Загл. с экрана. Язык Англ.
19. Сайт CrossPlatform.RU [Электронный ресурс] : Статья " Руководство по Qt Designer" URL : <http://doc.crossplatform.ru/qt/4.6.x/designer-manual.html> Загл. с экрана Яз. рус.
20. Хабр [Электронный ресурс] : многофункциональный сайт. Статья «Сигналы и слоты в Qt» URL: <http://habrahabr.ru/post/50812/> Загл. с экрана. Последнее изменение в 23:55 ,1 февраля 2009. Яз. рус.
21. Сайт lessons-tva.info [Электронный ресурс] : Статья «Создание отчетов в Access» URL: http://www.lessons-tva.info/edu/e-inf2/m2t4_6.html Загл. с экрана. Яз. рус.
- 22 Сайт CrossPlatform.RU [Электронный ресурс]: Статья “Таблицы стилей Qt” URL: <http://doc.crossplatform.ru/qt/4.4.3/stylesheet.html> Загл. с экрана Яз. рус.