Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

УТВЕРЖДАЮ
Заведующий кафедрой
_____ Г. М. Цибульский
подпись
« ____» _____ 2016 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02.04 «Информационные системы и технологии в медиаиндустрии»

Разработка модуля отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов

Руководитель _____ ст. преподаватель Е. А. Мальцев
подпись, дата

Выпускник _____ Н. А. Юматова
подпись, дата

Нормоконтролер _____ Е. А. Мальцев
подпись, дата

Красноярск 2016

СОДЕРЖАНИЕ

Введение ... 3
1 Обзор предметной области ... 5
1.1 Назначение программы учета электронных компонентов ... 5
1.2 Описание процесса движения электронного устройства МСУ ... 6
1.3 База данных системы учета электронных компонентов ... 9
1.4 Архитектура системы учета электронных компонентов ... 10
1.5 Выявление проблем в области представления данных ... 13
1.6 Обзор аналогов компонентов для создания отчетов ... 15
1.6.1 Компонент Crystal Reports.NET ... 15
1.6.2 Программный продукт Access ... 17
1.6.3 Компонент FastReport.Net ... 19
1.6.4 Компонент Reporting ... 23
1.7 Вывод к главе 1 ... 25
2 Проектирование и разработка модуля отчетности ... 28
2.1 Анализ требований ... 28
2.2 Анализ средств разработки ... 28
2.3 Архитектура модуля отчетности ... 30
2.4 Процесс разработки модуля отчетности ... 31
2.5 Вывод к главе 2 ... 41
Заключение ... 42
Список использованных источников ... 43
Приложение А Плакаты презентации ... 45

ВВЕДЕНИЕ

В настоящее время главной целью увеличения производительности труда является автоматизация технологических процессов. Необходимость автоматизации в любой сфере заключается в том, что предприятия нуждаются в ускоренной и эффективной обработке значительных объемов информации с использованием более трудоемких способов и алгоритмов.

На сегодняшний день самым удобным и распространенным средством передвижения на дальние расстояния нашей страны и за ее пределы является железнодорожный транспорт. По мере того, как развиваются информационные технологии в сфере железных дорог, информация, содержащаяся в выходных отчетных документах, с течением времени усложняется и увеличивается в объеме. Также изменяется представление о самом печатном документе, так как он может содержать и такие компоненты, как графики, изображения, сложные вычисления, диаграммы и многое другое. По разным оценкам интернет-порталов этап работы с печатными документами занимает в среднем около 60 % времени сотрудников.

В рамках проекта от Министерства путей сообщения РФ по автоматизации производства под руководством Семченко В. В., генерального директора ЗАО «Дорожного центра внедрения Красноярской железной дороги», была разработана система учета электронных компонентов, предназначенных для ремонта электровозов. С увеличением количества печатных документов и информации, содержащейся в них, необходимо разработать модуль отчетности с использованием специальных программных компонентов, позволяющих автоматизировать процесс создания печатных документов, ссылаясь на информацию из баз данных (БД).

Целью данной бакалаврской работы является разработка модуля отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов.

В ходе работы необходимо выполнить следующие задачи:

- анализ имеющейся программы и обзор аналогов компонентов для создания отчетов;
- проектирование модуля отчетности для системы учета электронных компонентов;
- разработка модуля для данной системы.

1 Обзор предметной области

1.1 Назначение программы учета электронных компонентов

Основное назначение разработанной системы учета электронных компонентов, предназначенных для ремонта электровозов, — выполнение учета ремонта микропроцессорных систем управления (МСУ) и отдельных электронных устройств, из которых состоят МСУ. Модель системы охватывает весь цикл ремонта электронных устройств и МСУ в целом, производимый организацией. МСУ — это бортовые микропроцессорные системы управления и диагностирования, которые устанавливаются на современных локомотивах и позволяют осуществлять контроль над техническим состоянием подвижного состава, выявлять нарушения в случаях режимов его эксплуатации, а также обнаруживать предотказные состояния и различные неисправности [19].

Интерфейс программы учета электронных компонентов представлен на рисунке 1.

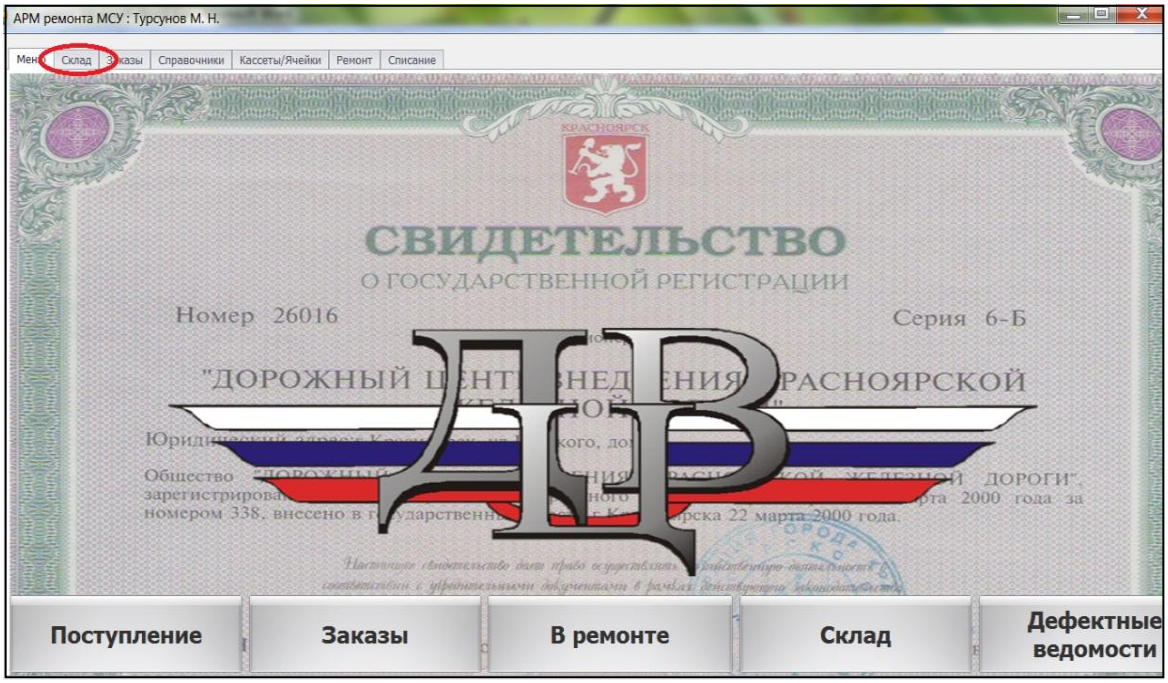


Рисунок 1 — Интерфейс программы учета электронных компонентов

1.2 Описание процесса движения электронного устройства МСУ

Для описания процесса движения компонентов используются диаграммы, которые позволяют наглядно представить весь процесс изнутри от начала, когда электронное устройство поступает в ремонт, до конца, когда отремонтированное изделие отправляется заказчику.

Диаграмма состояний электронного устройства МСУ представлена на рисунке 2.

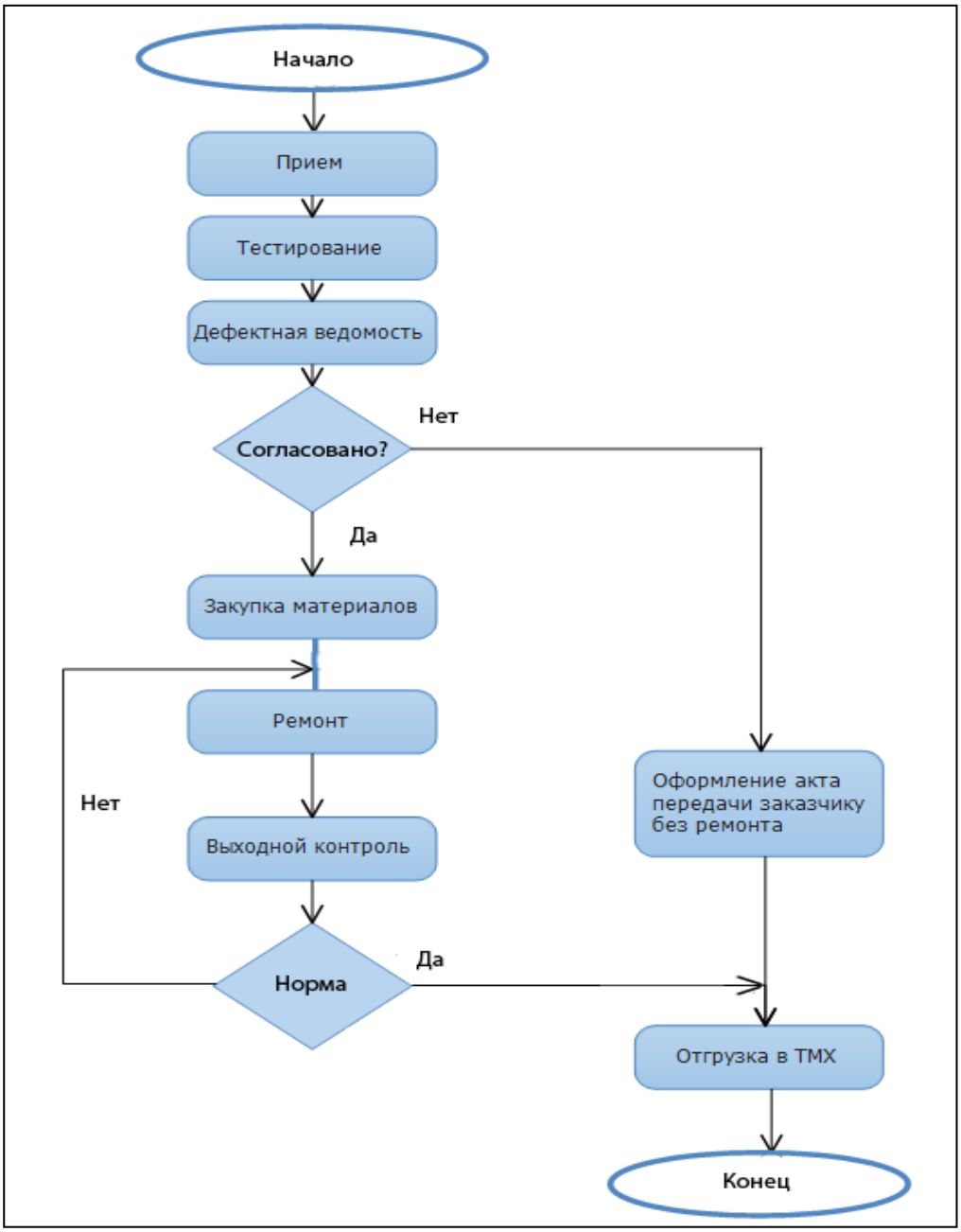


Рисунок 2 — Диаграмма состояний электронного устройства МСУ

Этапы процесса движения электронного устройства МСУ:

1) Первым этапом производится приемка от контрагента электронного устройства МСУ. На данном этапе формируется акт приема-передачи оборудования, спецификация. Электронное устройство вводится в систему с его серийным номером, указанием владельца, принадлежностью к МСУ и типу электронного устройства. Результатом выполнения является электронное устройство в состоянии «Принят» и сформированные документы.

2) Работником выполняется предварительный этап ремонта — тестирование, в результате чего электронное устройство МСУ принимает состояние «В ремонте», в системе заводится дефектная ведомость.

3) Окончание ремонта. Становится известен перечень материалов, требуемых для ремонта, а также перечень работ, произведенных над электронным устройством. Эта информация попадает в дефектную ведомость, которой присваивается номер. Здесь же формируются документы: протокол проведения стендовых испытаний, акт сдачи-приемки выполненных работ. Электронное устройство принимает состояние «Отремонтирован».

4) Отправление. Электронные устройства МСУ дожидаются партии и отправляются владельцу.

5) После того как оборудование покинуло предприятие ремонта и начало эксплуатироваться, есть вероятность его повторного выхода из строя. В этом случае, если электронное устройство снова попадает в ремонт, оно не вводится повторно: реализован механизм, блокирующий дублирование электронного устройства в базе данных. При попытке ввести серийный номер, который уже имеется в системе, программа предупредит, что электронное устройство с таким номером уже было введено и предложит перевести уже введенное электронное устройство в состояние «Принят». Далее его цикл повторяется. Таким образом, есть возможность отслеживать электронные устройства, попадающие в ремонт по нескольку раз.

Также стоит отметить, что в случае, когда ремонт электронного устройства МСУ задерживается из-за недостающих на складе материалов,

можно перевести их в состояние «В ожидании». По приходу необходимых деталей производится возврат в «Принятые».

Диаграмма процесса ремонта электронных устройств на предприятии представлена на рисунке 3.

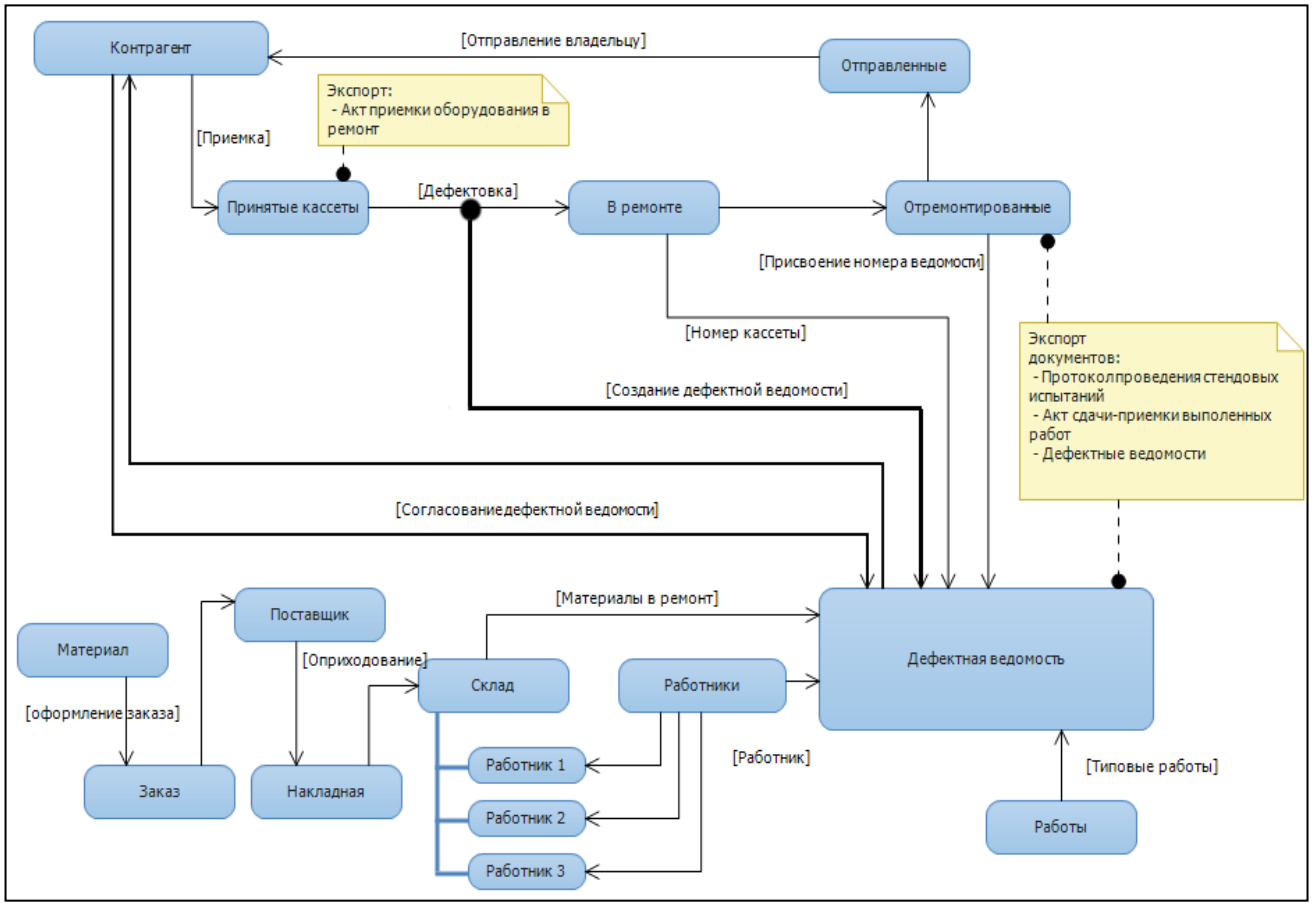


Рисунок 3 — Диаграмма процесса ремонта электронных устройств на предприятии

Диаграмма дополняет описание цикла: дефектная ведомость формируется непосредственно в процессе ремонта, в ней закрепляется список материалов, нужных для ремонта, которые поступают в ремонт со склада. Склад разделен на составляющие: основным складом считается склад администратора системы, а вторичными — склады, привязанные к каждому работнику, заведенному в системе. Прежде чем поступить на склад, материалы заказываются у поставщиков, затем по приходу создается накладная, на основании которой детали заносятся в склад.

При переходе в ремонт, электронное устройство МСУ обзаводится своей дефектной ведомостью, которая на данном этапе пуста (она содержит только информацию о работнике и электронном устройстве). После ремонта она пополняется списком работ и материалов, затраченных на ремонт электронного устройства, а также приобретает номер.

1.3 База данных системы учета электронных компонентов

База данных — хранилище структурированных и взаимосвязанных данных. БД имеют многочисленные преимущества, когда речь идет о больших объемах информации, так как они позволяют хранить большее количество данных по сравнению с другими хранилищами данных.

При разработке баз данных необходимо учитывать то, что эти данные в дальнейшем должны использоваться в каком-либо приложении, поэтому для отлаженной работы всей будущей системы БД должна быть организована грамотно: связи детально продуманы, а структура оптимальна.

База данных системы учета электронных компонентов, предназначенных для ремонта электровозов, содержит в себе достаточно большое количество таблиц и связей между ними, как и в любой другой системе, созданной для крупной организации. Это, в свою очередь, говорит о значительном множестве хранящихся в ней данных. Также, из диаграмм, представленных выше, видно, что в целом система имеет достаточно сложную структуру. Поэтому следует сделать вывод, что с составлением отчетных документов о каких-либо проделанных этапах вручную могут возникать определенные трудности, так как необходимо обрабатывать большое количество данных.

БД системы учета электронных компонентов, предназначенных для ремонта электровозов, организована следующим образом и представлена в виде UML-диаграммы на рисунке 4.

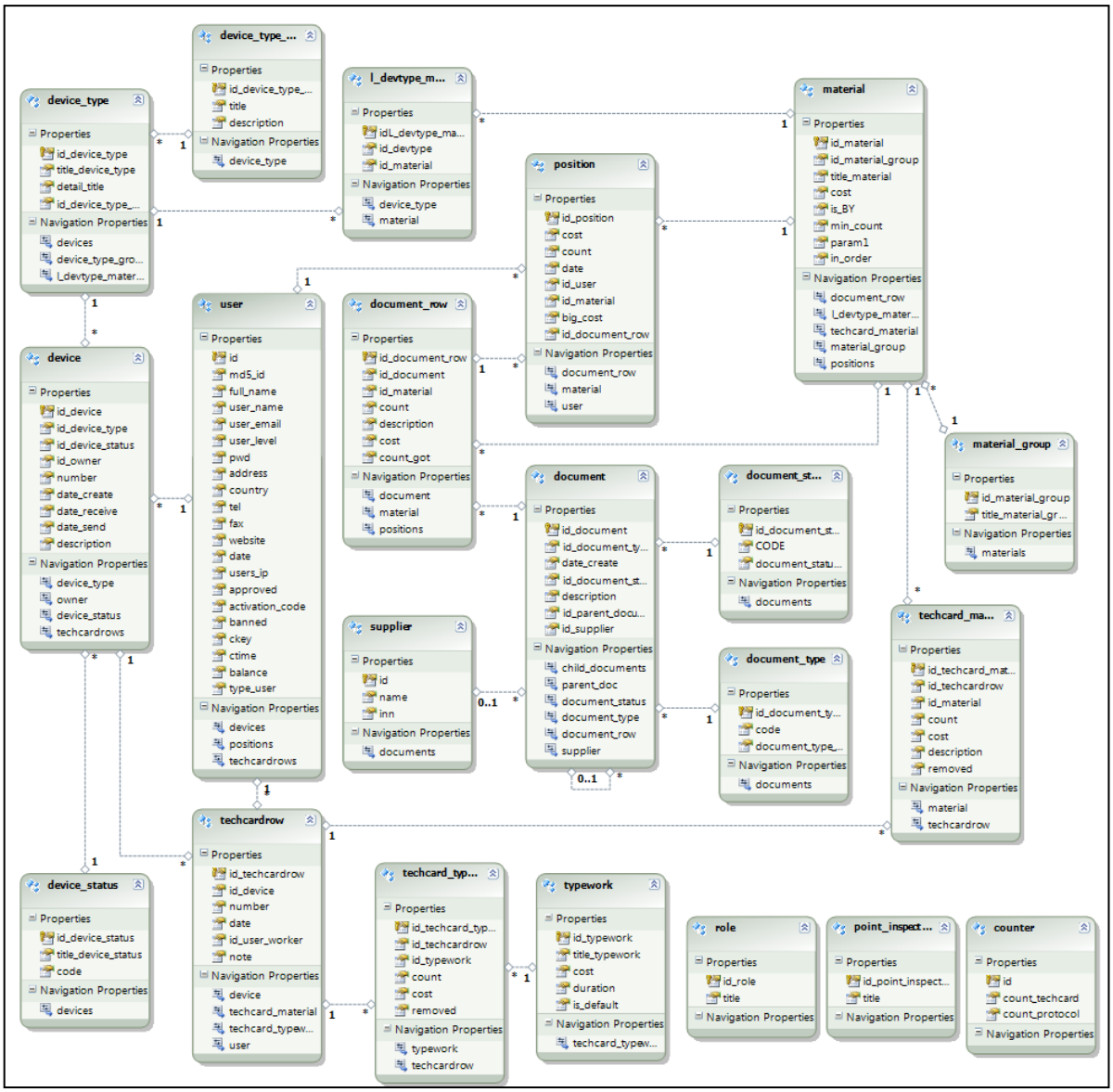


Рисунок 4 — БД системы учета электронных компонентов

1.4 Архитектура системы учета электронных компонентов

Система имеет трехуровневую архитектуру:

- уровень хранения данных (БД MySQL);
- уровень доступа к данным (слой ORM — Entity Framework);
- уровень представления данных (Windows Forms, приложение компании Developer Express).

На уровне хранения данных используется программное обеспечение (ПО) MySQL. Данное ПО, используемое в системе, представляет собой достаточно быстрый многопользовательский, многопоточный надежный SQL-сервер

баз данных. SQL — это язык структурированных запросов, который применяется для создания, модификации и управления данными в БД [6].

Сервер MySQL предназначен не только для сложных по структуре производственных систем с большой нагрузкой, но и для встраивания в ПО массового распространения. Программный продукт MySQL обладает достаточным количеством различных возможностей таких, как управление большими БД, полной поддержкой нескольких различных кодировок, использование привилегий и паролей для гибкости и безопасности, а также для доступа с удаленного компьютера и другие.

На уровне доступа к данным задействована современная технология такая, как Object-Relational Mapping Entity Framework (ORM EF) версии 6.0. ORM — объектно-реляционное отображение, технология программирования, которая связывает БД с концепциями объектно-ориентированных языков программирования. Тем самым данная технология позволяет создавать «виртуальные» БД.

Entity Framework, в свою очередь, представляет собой специальную объектно-ориентированную технологию, базирующуюся для работы с данными на фреймворке .NET — программной платформе, определяющей структуру программной системы, разработанной корпорацией Microsoft. Технология EF являет собой более высокий уровень абстракции, который позволяет абстрагироваться от самой БД и работать с данными независимо от типа хранилища. Если на физическом уровне операции производятся с таблицами, индексами, первичными и внешними ключами, то на концептуальном уровне, который предлагает Entity Framework, работа осуществляется непосредственно с самими объектами [5].

Работа с EF при помощи интегрированной среды разработки Visual Studio предоставляет возможности, позволяющие создавать модели БД. В программе учета электронных компонентов, предназначенных для ремонта электровозов, используется технология Database-First (DB First) — один из подходов по проектированию БД. Такой подход с помощью различных инструментов

позволяет автоматически генерировать класс модели по имеющейся базе данных. С одной стороны такой класс содержит в себе запросы к базе данных, с другой — методы для управления данными из программы-интерфейса пользователя. В результате технология EF позволяет работать с таблицами БД как с классами C#, а с полями этих таблиц — как со свойствами классов.

Уровень представления данных отличается использованием специального приложения от компании Developer Express (DevExpress). Данная компания занимается разработкой компонентов и дополнений для таких платформ, как WinForms, ASP.NET, WPF, Silverlight, Windows 8 XAML, HTML JS, для анализа и графического представления данных, проектирования отчетов и других продуктов, а также для создания своих приложений [11].

Продукты от компании DevExpress обладают удобством и простотой использования, а также большой функциональностью. В состав таких продуктов, использующихся в системе учета электронных компонентов, входят XtraGrid — для работы с выводом таблиц, Rich Edit Control — для работы с текстом и другие.

Обобщенная архитектура системы учета электронных компонентов представлена на рисунке 5.

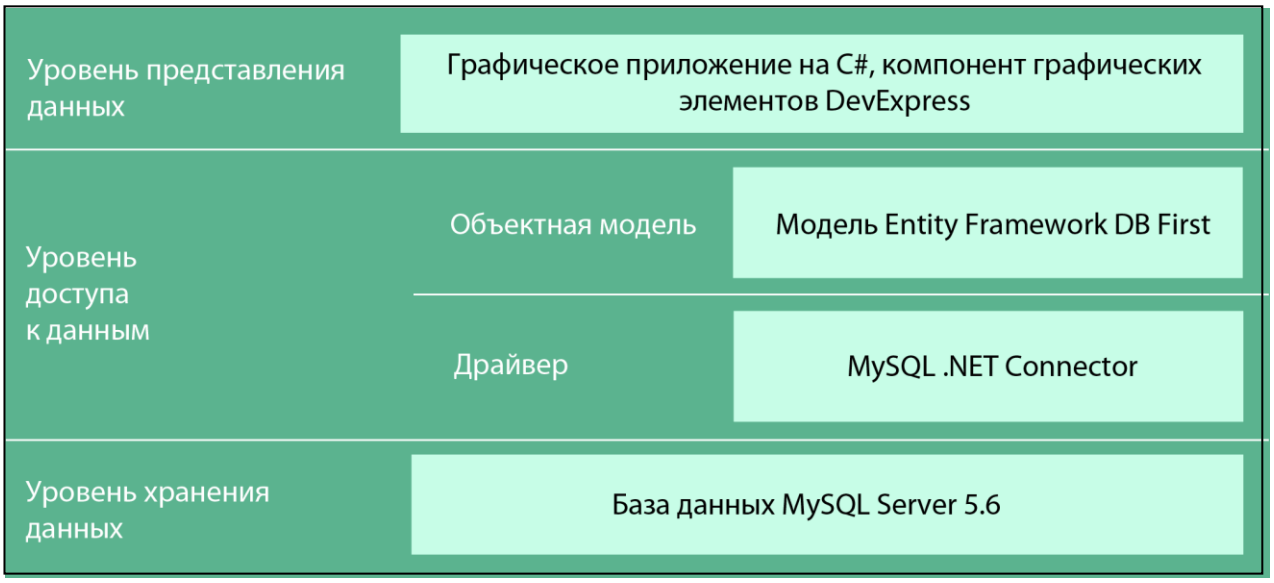


Рисунок 5 — Обобщенная архитектура системы учета электронных компонентов

1.5 Выявление проблем в области представления данных

У корпоративных информационных систем существует множество задач, и одной из главных является оперативное представление информации, которая необходима для принятия различных решений. Но в то же время структура современных СУБД, которые служат основой информационных систем, ориентирована в первую очередь на компактное хранение необходимой информации, но никак не на оптимизацию произвольной выборки и представления этих данных. Поэтому, как правило, информация, которая хранится в корпоративных информационных системах, не используется эффективно. Следовательно, главная проблема заключается в том, что акцент делается именно на хранение информации, а не на предоставление ее конечному пользователю в удобочитаемом виде, то есть в виде отчета.

Отчет — информация, представленная в структурированном и в удобном для чтения виде, содержащая не только табличные или кросс-табличные данные, но и графики и множество другой различной информации полезной для анализа, выводящаяся на экран, в печать или файлы различных форматов. Таким образом, под отчетом понимают любую страницу с информацией [4].

Обычно информационные системы, имеющие масштабы предприятия, владеют ограниченным числом встроенных отчетов, которые предоставляют различную отчетную информацию государственным службам или вышестоящим организациям. В реальной жизни зачастую требования бизнеса имеют способность изменяться чуть ли не каждый день, поэтому специфика таких аналитических отчетов, позволяющих облегчить процесс принятия решений, состоит в их быстрой изменчивости. Как правило, заказами на разработку таких отчетов занимаются отдел автоматизации предприятия, если такой существует, либо разработчики других, сторонних, фирм. Стоит отметить, что потребности в таких аналитических отчетах увеличиваются по мере их создания в геометрической прогрессии. Поэтому решить данную проблему путем встроенных в среду разработки средств, таких как, например,

компонент для создания итоговых отчетов DataReport, который имеется в составе программного продукта Visual Basic 6, практически невозможно. Для таких задач более всего подходят специализированные генераторы отчетов.

Генератор отчетов — особый класс программ, который позволяет представить информацию в структурированном и удобочитаемом виде. То есть создать из необходимых данных информацию (отчет), которую можно распечатать или сохранить в различных электронных форматах [20].

Но такие программные продукты, не смотря на их преимущества и массовость использования, имеют ряд проблем, часто возникающих при генерации отчетов, которые составить иногда бывает очень проблематично. Трудности связаны не столько с недостатками самих электронных компонентов для создания отчетов, сколько с человеческим фактором и организацией БД. Некоторые из этих проблем представлены ниже:

- при проектировании БД разработчики не всегда детально продумывают какие-либо связи;
- БД бывает чрезмерно усложнена (в таблицах существует избыток ненужных связей, некоторые из которых даже не используются в создании отчетных документов);
- обработка SQL-запросов для генерации отчетов выполняется дольше из-за неоптимальной структуры БД.

Таким образом, из всего вышесказанного следует вывод, что при разработке любой сложной системы для какого-либо предприятия или организации на представление данных конечному пользователю в удобочитаемом виде не уделяется должного внимания. Из-за этого информация, содержащаяся в БД, не может быть использована достаточно эффективно. Исходя из анализа данной области, существующую проблему можно решить путем использования специальных программных продуктов для создания отчетов, которые позволят работать с данными, хранящимися в БД, более продуктивно. Поэтому не менее важно грамотно разработать

структуру самой БД, чтобы при помощи генератора отчетов разработчик смог корректно построить любой отчетный документ.

1.6 Обзор аналогов компонентов для создания отчетов

Существует достаточно много программных продуктов, компонентов для создания отчетов с целью представить конечному пользователю информацию в готовом виде.

1.6.1 Компонент Crystal Reports.NET

Программный продукт Crystal Reports долгие годы был связан с программным обеспечением крупнейшей корпорации Microsoft. С 1993 года компонент Crystal Reports входит в состав интегрированной среды разработки Visual Studio.

Пакет Crystal Reports для программной платформы Visual Studio .NET — это дополнение к комплекту уже существующих инструментальных средств технологии Visual Studio .NET, которое в полной мере использует все достоинства среды разработки .NET. При помощи конструктора отчетов Crystal Reports Designer можно создавать и модифицировать отчеты, находясь в пределах интегрированной среды разработки Visual Studio .NET.

Данный программный продукт позволяет создавать различные виды отчетов: от простых списков, которые могут содержать всего лишь несколько столбцов, до сложных административных актов, использующие множество таблиц и графиков. В зависимости от существующих потребностей, генератор отчетов может быть использован для представления выходных документов и любых типов отчетов [18].

Возможности технологии Crystal Reports.NET:

1) Тесная интеграция с технологией Visual Studio .NET.

Все функциональные средства, необходимые для проектирования отчетов, предоставляются в полной мере. Они позволяют осуществлять программное управление внешним видом отчета, а также способами его

создания и просмотра и прочими другими эффектами. Для проектирования отчетов не требуется открывать отдельное приложение.

2) Модули просмотра Windows-отчетов, служащих для отправки сообщений о возникших ошибках в компанию Microsoft, и Web-отчетов, интерактивных документов, совместимых с большинством браузеров.

В технологию Crystal Reports.NET включен модуль просмотра отчетов, который ориентирован на платформу WinForms. Такая возможность позволяет получить полное впечатление от просмотра отчета. Данный модуль построен на базе классов платформы WinForms и предоставляет такие функциональные возможности, которые необходимы каждому пользователю, а именно, переход к более высокому уровню детализации, поиск, экспортирование и тому подобное.

3) Упрощенное развертывание.

Технология Crystal Reports.NET содержит некоторое число модулей слияния, что существенно упрощает реализацию установочных проектов. Вместо того чтобы вручную определять, какие требуются DLL-библиотеки — динамические библиотеки, которые позволяют использовать различные программные приложения многократно — и другие компоненты, обеспечивающие функционирование отчетов, созданных при помощи технологии Crystal Reports, в установочный проект пользователю достаточно добавить один из существующих модулей слияния.

4) Источник данных ADO.NET.

С появлением технологии доступа к данным ADO.NET, неотъемлемой части платформы .NET Framework, доступ к любым данным значительно упростился. Технология Crystal Reports.NET также может воспользоваться преимуществами технологии ADO (ActiveX Data Objects — объекты данных фреймворка ActiveX) и объекта пересылки данных DataSet, входящего в технологию ADO.NET.

Что касается основных недостатков компонента Crystal Reports.NET, то данная технология не имеет столь значимых недостатков, которые могли бы нарушить всю структуру создания отчетов. Существуют некоторые ошибки, недочеты, которые производители данного программного продукта исправляют в ближайших обновлениях версии программы.

Пример использования компонента Crystal Reports.NET представлен на рисунке 6.

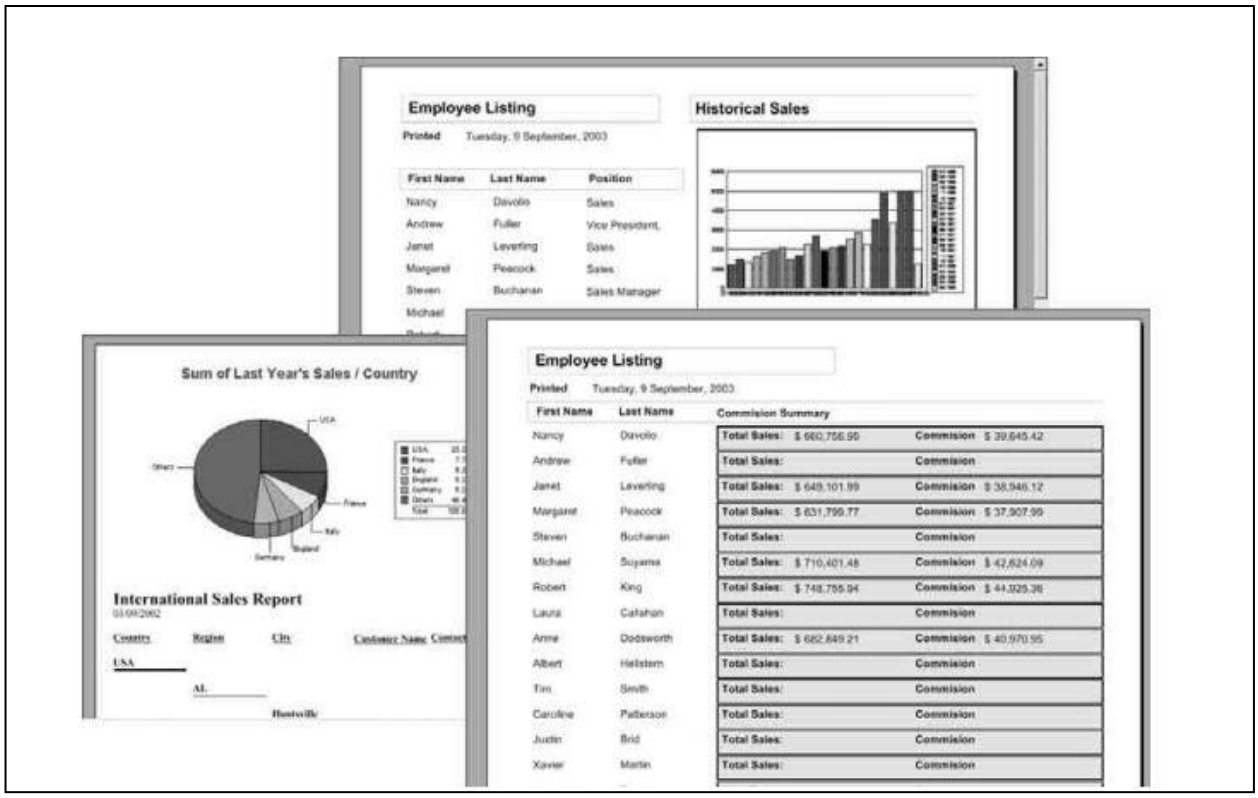


Рисунок 6 — Пример использования компонента Crystal Reports.NET

1.6.2 Программный продукт Access

Программный продукт Access от компании Microsoft — это генератор отчетов для создания простых выходных отчетных документов для получения и анализа данных [16].

В отчете, спроектированном при помощи программы Access, необходимые данные отображаются из запроса или таблицы с добавлением к ним текстовых элементов, которые упрощают его восприятие. Отчеты могут

содержать подробные сведения об отдельных записях, сводные сведения о большой группе записей либо и то, и другое. Кроме того, такие отчеты также можно использовать при создании наклеек для списков рассылок и многого другого. Стоить отметить, что данная технология для создания отчетов предоставляет возможности группировки, форматирования объектов, создания формул из большого набора. Идеология построения отчетов аналогичная технологии Crystal Report.

Возможности программы Microsoft Access для создания отчетов:

1) Прикрепление изображений к базам данных и дальнейшее применение этих изображений к нескольким объектам с последующим использованием в отчете. Обновление одного образа обновляет его во всех местах, где оно используется в пределах всей базы данных, что значительно ускоряет процесс разработки отчетов.

2) Программа Access также позволяет использовать стандартные темы офисного пакета приложений Microsoft Office одновременно с применением профессионально разработанных наборов шрифтов и цветов для всех форм и отчетов.

3) Программный продукт Microsoft Access включает различные средства для выделения данных в отчете. Пользователь может добавить до 50 правил условного форматирования для каждого элемента управления или группы элементов управления, а в клиентских отчетах существует возможность добавления строк для сравнения данных по записи.

4) В программном продукте Access существует метод конструктора по умолчанию для отчетов, то есть размещение элементов управления в макете. Эти сетки помогают выровнять и легко установить размеры элементов управления, необходимых для отчетов.

Основные недостатки программы Microsoft Access для создания отчетов:

- поддержка СУБД только программы Access;

- не подходит для создания отчетов, основывающихся на базах данных, имеющих большое количество таблиц и более сложные связи;

- отсутствует возможность создания кросс-таблиц.

Исходя из возможностей и недостатков данной программы для создания отчетов, можно сделать вывод, что программный продукт Microsoft Access удобен в использовании при создании достаточно простых отчетных документов, основывающихся на СУБД программы Access. Но, по сравнению с другими генераторами отчетов, программный продукт Access имеет недостаточно возможностей и функций для решения более сложных задач в построении отчетов.

Пример использования программы Access корпорации Microsoft для создания отчетных документов представлен на рисунке 7.

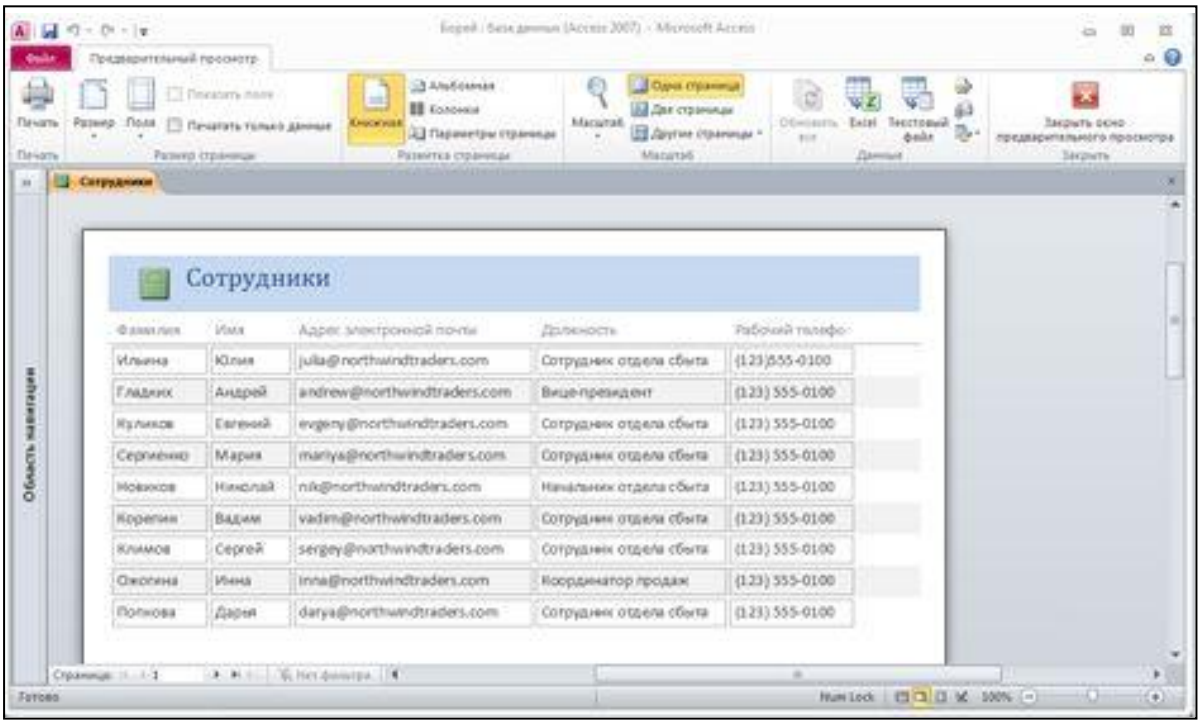


Рисунок 7 — Пример использования программы Microsoft Access для создания отчетных документов

1.6.3 Компонент FastReport.Net

Программный продукт FastReport.Net компании Fast Reports — это генератор отчетов, имеющий различные возможности для платформ WinForms,

ASP.NET и MVC Framework. Стоит отметить, что данный компонент является отечественным продуктом [2].

Возможности компонента FastReport.Net:

1) Компонент FastReport.Net содержит визуальный дизайнер для создания и модификации отчетов. Созданное разработчиком приложение может запустить дизайнер отчетов из кода программы.

2) Используя программный продукт FastReport.Net, можно создавать отчеты, работающие независимо от приложения, то есть данный программный продукт может быть использован как самостоятельное решение для построения отчетов.

3) Содержит онлайн дизайнер отчетов для платформы ASP.Net.

4) Имеется возможность подключения к любой базе данных, использования любых ее таблиц или создания своих собственных запросов на языке структурированных запросов SQL.

5) Также есть возможность добавления одной или нескольких диалоговых форм в отчет для запроса параметров перед запуском отчета.

6) При помощи встроенного скрипта, пользователь может управлять взаимодействием между элементами управления диалоговых форм и производить трудоемкую обработку данных.

7) Полученный результат можно увидеть в окне предварительного просмотра, вывести отчет на печать или сохранить его в различных форматах.

Если говорить о недостатках компонента FastReport.Net, то данный компонент так же, как и программный продукт Crystal Reports.NET, не имеет существенных недостатков. Но так как он обладает еще более расширенным функционалом, чем программный продукт корпорации Microsoft, не исключена возможность появления большего количества различных ошибок при использовании данного генератора отчетов, так как, чем по структуре своей объемнее программа, тем велика вероятность не учесть все нюансы всех функций и свойств.

Пример использования компонента для создания отчетов FastReport.Net представлен на рисунке 8.

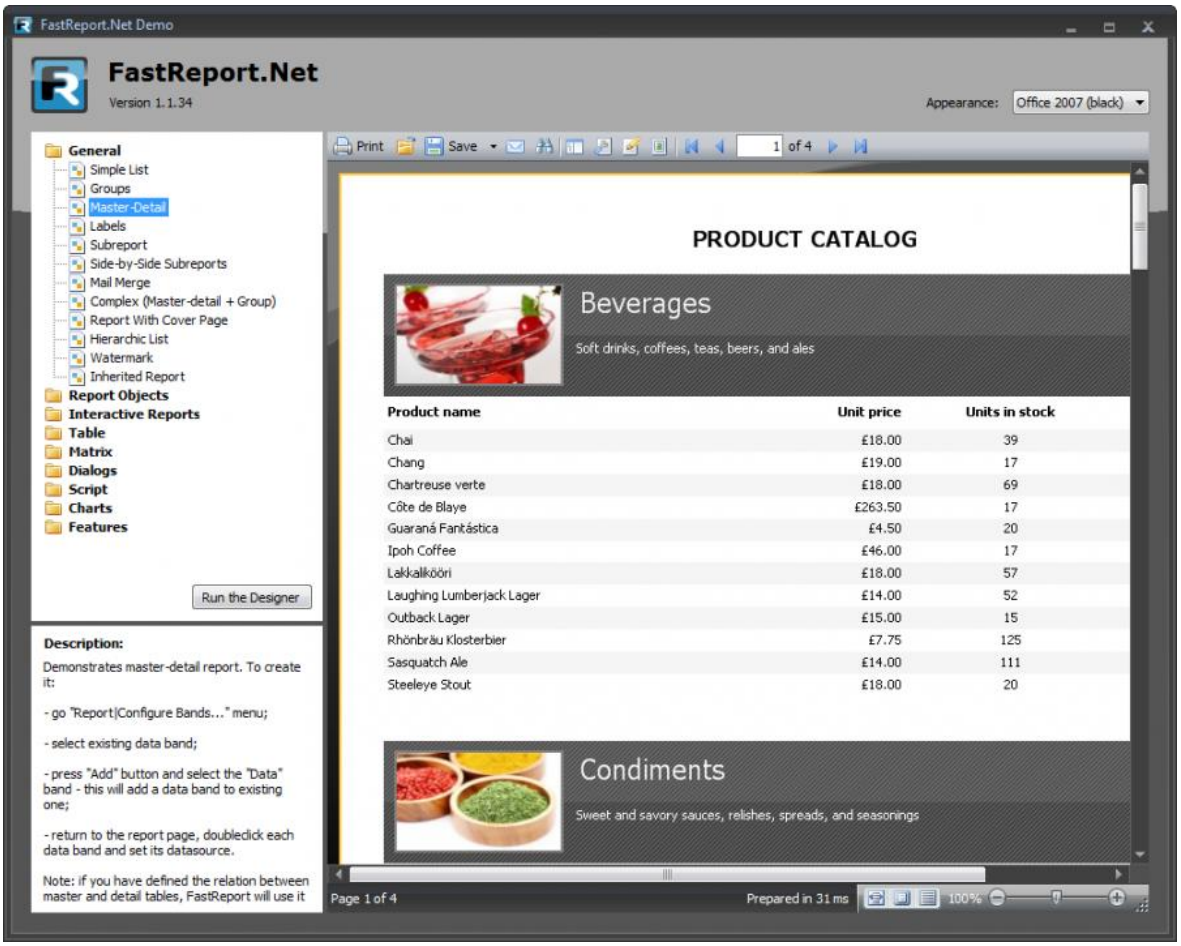


Рисунок 8 - Пример использования компонента FastReport.Net

Также существуют некоторые особенности данного программного продукта FastReport.Net, которые позволяют использовать другие дополнительные технологии и еще большие возможности:

1) Компонент FastReport.Net поставляется с исходными кодами (версия Professional). Его можно адаптировать под собственные нужды.

2) Разумная цена и лицензионная политика. В стоимость лицензии входит визуальный дизайнер, который позволяет пользователям самостоятельно разрабатывать отчеты.

3) Генератор отчетов FastReport.Net позволяет добавить в отчет географические карты с помощью объекта maps.

4) Интегрированная в программном продукте для составления отчетности FastReport.Net поддержка облачных сервисов дает возможность сохранять различные отчеты в один из трех ресурсов для хранения данных: Google Drive, SkyDrive, DropBox.

5) Расширяемая архитектура компонента FastReport.Net позволяет создавать и подключать к отчету свои объекты, фильтры экспорта, функции. Возможности также можно расширить.

6) Поддержка импорта отчетов компонента Crystal Reports.

Из вышеперечисленного можно сделать вывод, что данный компонент обладает достаточно большим и разнообразным функционалом. Использование дополнительных возможностей программного продукта FastReport.Net таких как, например, добавление географических карт с помощью объекта maps, представлен на рисунке 9.

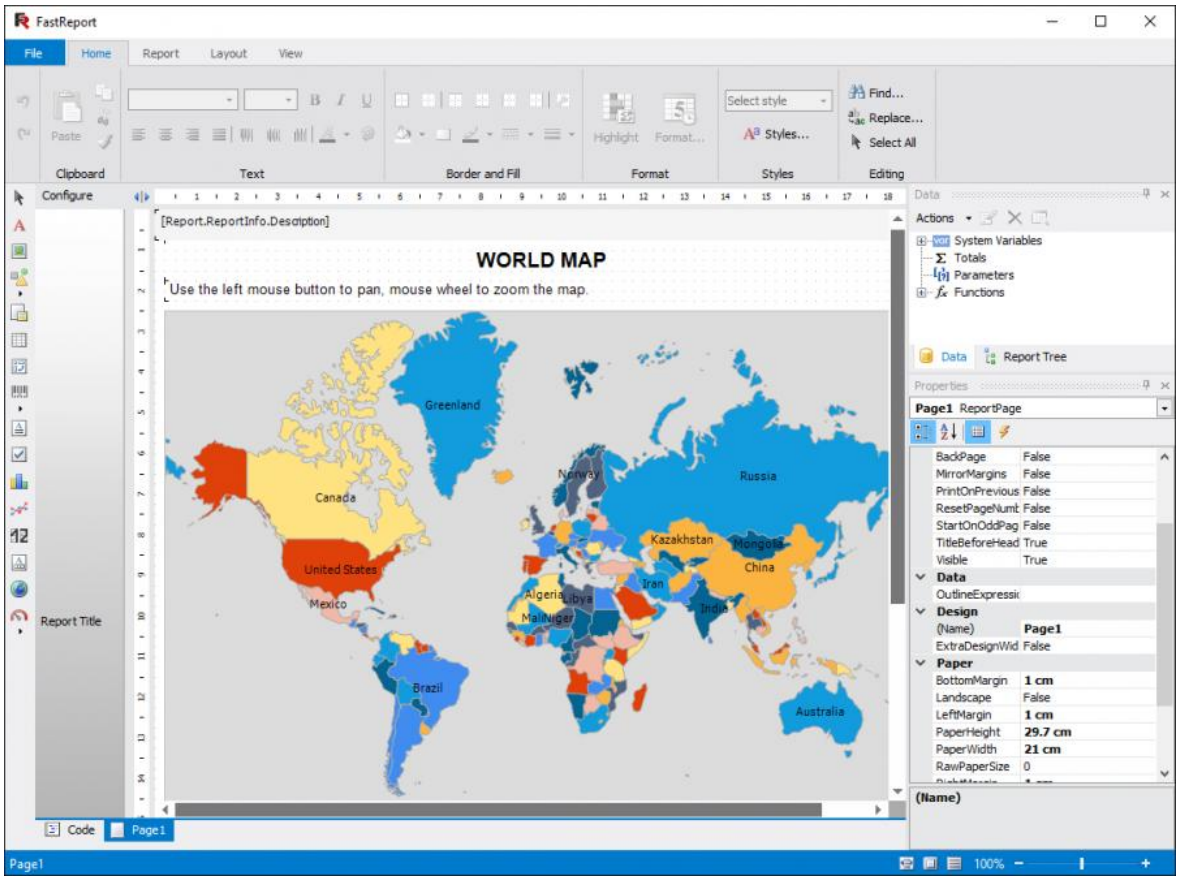


Рисунок 9 - Пример использования объекта maps в программном продукте FastReport.Net

1.6.4 Компонент Reporting

Компонент Reporting, входящий в состав целого набора всевозможных компонентов от компании DevExpress, — это решение для генерирования различных отчетов для всех ведущих платформ .NET (WinForms, ASP.NET, MVC, WPF, Silverlight). Инструмент для создания отчетов был разработан с целью помочь разработчикам составлять отчеты для конечных пользователей в кратчайшие сроки. Поставляется с набором различных функций [1].

Возможности компонента Reporting:

1) Данный продукт работает с любым из распространенных источников и сервисов данных, таких как MS SQL Server, MS Access, Oracle, DB2, PostgreSQL, MySQL, Sybase, Firebird, Interbase, XML или через интерфейс API.

2) Компонент DevExpress Reporting обеспечивает создание отчетов в формате XML или любом другом, поддерживающем интерфейсы IList, IList[t] и IEnumerable[t].

3) Инструмент для создания макетов отчетных документов XtraReports, входящий в состав компонента отчетности Reporting, содержит стандартные файлы кода, что позволяет упростить процесс программирования и создания отчетных документов.

4) Компонент Reporting позволяет обеспечить доступность данных из любого ресурса через объекты данных программной платформы Visual Studio .NET.

5) Программный продукт Reporting предлагает возможность для работы с графическим представлением данных в виде диаграмм, графиков и других видов изображений, позволяющих наглядно представить имеющуюся информацию для ее визуального анализа.

6) Компонент Reporting предоставляет пользователю возможность использовать любые изображения, хранящиеся в БД, для создания отчетных документов. Благодаря данному программному продукту пользователю не нужно помещать каждое изображение в отчет самостоятельно, достаточно

создать необходимое подключение к БД, и все изображения будут сами загружаться в отчет в том порядке, в каком они хранятся в БД.

7) Конструктор отчетов XtraReports предлагает конечному пользователю достаточно широкий набор различных функций и возможностей таких, как автоматический подсчет итогов, нахождение среднего, максимального и минимального значений, форматирование текста, изменение масштаба отчетов и другие преимущества.

8) Компонент Reporting поддерживает форматы различных видов таких, как Word, Excel, PDF, HTML, Plain Text, RTF и другие.

9) Инструмент XtraReports предоставляет конечному пользователю множество различных инструментов для работы с отчетными документами: Label (отображение текста), Picture Box (вставка изображений), Table (работа с таблицами), Line (отрисовка линий), Shape (работа с геометрическими фигурами), Chart (создание гистограмм).

Данный компонент имеет ряд и других преимуществ, выше были представлены лишь некоторые из них.

Что касается недостатков программного продукта Reporting, то он имеет незначительные недочеты в виде некоторых ошибок при использовании, которые разработчики в последующих версиях программного продукта исправляют, а также существуют недоработанные функции, которые на данный момент не весьма практичны, но с последними версиями разработчики их совершенствуют. Стоит отметить, что компания DevExpress является развивающейся и с каждой новой версией своих продуктов разработчики предлагают пользователям еще большие возможности работы с отчетными документами. Поэтому можно сделать вывод, что на данный момент существенных недостатков у данного компонента для создания отчетности не существует.

Пример использования компонента DevExpress Reporting представлен ниже на рисунке 10.

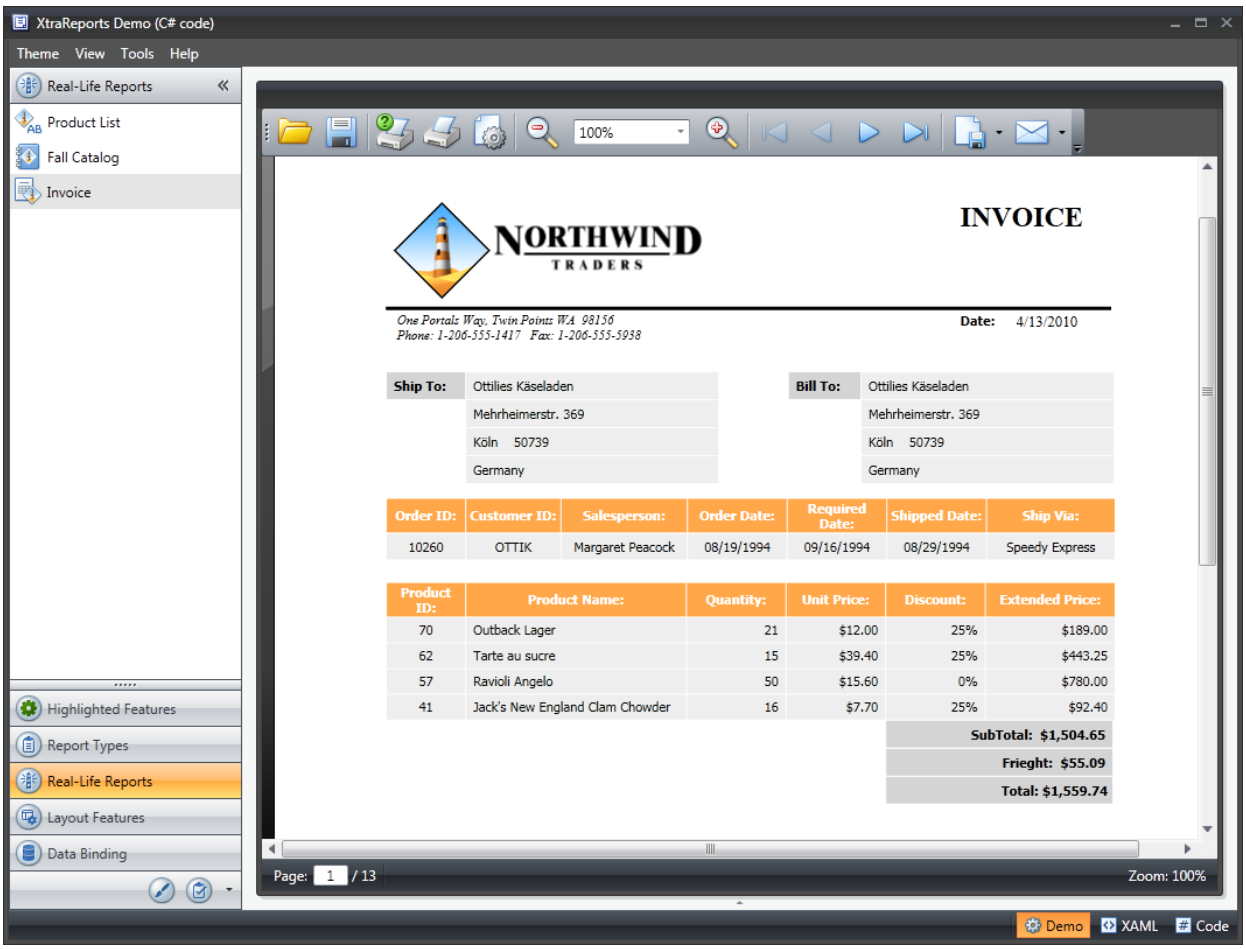


Рисунок 10 — Пример использования компонента DevExpress Reporting

1.7 Вывод к главе 1

Любое разработанное приложение или программа запускаются конечным пользователем во многом ради получения определенных данных. Форма представления данных существенным образом влияет на возможность их анализа и принятия на этой основе соответствующего решения. Одним из наиболее распространенных способов представления данных является отчет, который используется в подавляющем большинстве бизнес-приложений.

Для генерации отчетов существует достаточно большое количество различных программных компонентов, которые позволяют автоматизировать процесс создания отчетов вручную и сэкономить на их разработке достаточно времени.

Выше были рассмотрены несколько из наиболее известных программных продуктов для решения проблем в области представления данных. Итоговая сравнительная таблица 1 представлена ниже.

Таблица 1 — Итоговая сравнительная таблица

Возможности	Название компонента отчетности			
	Crystal Reports.NET (For Visual Studio(VS))	Access (Microsoft Corp.)	FastReport.Net (Fast Reports, Inc)	Reporting (Developer Express Inc.)
Поддержка СУБД	MS SQL Server, MS Access, Oracle, MySql	Access	MS SQL Server, MS Access, Oracle, DB2, PostgreSQL, MySQL и др.	MS SQL Server, PostgreSQL, MS Access, Oracle, DB2, MySQL и др.
Дизайнер отчетной формы	Встроен в VS.NET	Отсутствует	Приложение запускает дизайнер отчетов из кода	Конструктор отчетов входит в состав компонента
Вставка изображений (поддержка форматов)	Из графических файлов при помощи ссылок, хранящихся в БД(JPEG и др.)	С помощью элемента управления «Image» (JPG, BMP, GIF и др.)	При помощи объекта «Рисунок» (BMP, PNG, JPG и др.)	С помощью инструмента Picture Box (JPEG, GIF, PNG и др.)
Модуль просмотра отчетов	WinForms Viewer/Web Forms Viewer	Отсутствует	Стандартное окно предварительно просмотра	Специальное окно просмотра
Представление данных в виде графических изображений	Графики, диаграммы, гистограммы	Графики, диаграммы	Диаграммы, гистограммы, графики, линии, фигуры	Графики, диаграммы, гистограммы, линии, фигуры
Совместимость с Visual Studio	Совместим	Не предназначен для работы с VS	Совместим	Совместим
Автоматический подсчет итогов	Сумма, среднее значение, max, min	Отсутствует	Сумма, среднее значение, max, min	Сумма, среднее значение, max, min
Содержание стандартных файлов кода	Не содержит	Не содержит файлы кода	Содержит только в версии Professional	Содержит
Форматы составления отчетов	PDF, Word, Excel, RTF, HTML, Plain Text и др.	Word, Excel	Word, Excel, PDF, HTML, Plain Text, RTF и др.	Word, PDF, Excel, Plain Text, HTML, RTF и др.

В данной системе автоматизированного рабочего места по ремонту МСУ, разработанной для учета электронных компонентов, предназначенных для ремонта электровозов, отсутствует модуль для создания отчетных документов. Поэтому в связи с требованиями «ДЦВ Красноярской железной дороги» необходимо разработать модуль, который позволит проектировать и создавать любые отчеты с использованием одного из представленных выше программных компонентов, ссылаясь на информацию из БД.

Для разработки данного модуля, несмотря на схожие возможности с другими программными продуктами, выбран компонент для создания отчетных документов Reporting от компании DevExpress в связи с тем, что данный программный продукт имеет функционал, который позволит решить такую задачу, как разработка модуля отчетности для системы учета электронных компонентов. Данный компонент совместим с интегрированной средой разработки Visual Studio, что немаловажно, так как вся система разработана именно в этой среде. Программный компонент Reporting позволяет производить автоматический подсчет итогов, что является необходимым для любых отчетов, связанных с финансами или подсчетом общего количества компонентов. А также компонент для создания отчетов Reporting благодаря своим возможностям таким, как наличие дизайнера отчетной формы, содержание стандартных файлов кода, наличие модуля просмотра отчетов и других преимуществ, достаточно удобен в использовании.

Одной из основных причин выбора данного компонента является то, что программный продукт для создания отчетов Reporting поставляется в составе с другими компонентами от производителя DevExpress, которые являются полезными для создания качественной системы и уже используются в программе учета электронных компонентов. Другими словами, вся система построена при помощи компонентов компании DevExpress, которые дополняют друг друга посредством своих возможностей, следовательно, внедрение в программу дополнительного компонента для отчетности Reporting увеличит эффективность от использования разработанной системы в целом.

2 Проектирование и разработка модуля отчетности

Для выявления требований к проектируемому модулю было проведено интервью с представителем «ДЦВ Красноярской железной дороги». Впоследствии были сформулированы предварительные требования и средства разработки.

2.1 Анализ требований

Функциональные требования:

1) модуль должен предоставлять конечному пользователю информацию в структурированном и удобочитаемом виде;

2) модуль должен предоставить конечному пользователю возможность работать с оформлением отчетов, сохранять их в нужном формате, производить поиск по документу;

3) модуль должен предоставить пользователю возможность работать с данными, хранящимися в БД.

Нефункциональные требования:

1) исходный код программного модуля должен быть написан на языке программирования C# с использованием платформы WinForms;

2) модуль должен быть разработан с использованием компонента для создания отчетов Reporting от компании DevExpress версии 11.1.8;

3) модуль должен быть разработан в среде .NET Framework 4.0.

2.2 Анализ средств разработки

Для разработки данного модуля отчетности выбрана специальная объектно-ориентированная технология Entity Framework (EF) на базе программной платформы .NET, которая позволяет более качественно работать с различными данными, хранящимися в БД, и используется в системе учета электронных компонентов, предназначенных для ремонта электровозов. Технология EF предоставляет возможность работы непосредственно с БД через

объектно-ориентированный код языка программирования C#, что позволяет такому подходу обладать рядом существенных преимуществ. В качестве способа взаимодействия с БД выбрана технология Database First. Данный подход подразумевает создание графической модели EDMX из уже существующей БД, что позволяет использовать удобный графический интерфейс для взаимодействия с БД в виде диаграмм и объектную модель в виде классов языка программирования C#. EDMX — файл XML, который определяет концептуальную модель, режим хранения и их сопоставление, создающийся при помощи мастера моделей EDM (Entity Data Model). Более подробно данные технологии рассмотрены в первой обзорной главе.

Для разработки модуля отчетности выбран компонент Reporting от производителя DevExpress. Данный программный продукт обладает разнообразным функционалом, позволяющим создавать отчеты в кратчайшие сроки. Для работы непосредственно с самим отчетом используется инструмент XtraReport Class v11.1. Конструктор отчетов XtraReports — это следующее поколение мультиплатформенных решений для создания отчетности. Данный инструмент предоставляет полномасштабный конструктор отчетов для конечного пользователя и позволяет обеспечить их возможностью редактировать отчеты, которые уже существуют, и создавать совершенно новые отчетные документы с самого начала [7].

Конструктор отчетов XtraReports обладает достаточно гибкой архитектурой и набором различных специальных инструментов, позволяющих грамотно создавать отчетные документы, а также поддерживается интегрированной средой разработки Visual Studio .NET. Инструмент содержит в себе такие возможности, как форматирование текста, изменение масштаба отчета, управление файлами в конструкторе, привязку элементов управления к данным и многие другие.

Для работы непосредственно с самими данными, хранящимися в БД, необходимо подключиться к существующей базе данных. Для подключения

используется MySql .NET Connector — драйвер, который позволяет клиентским приложениям подключаться к серверам MySQL.

2.3 Архитектура модуля отчетности

Исходя из анализа структуры программы и требований к разработке модуля отчетности, была определена архитектура модуля отчетности путем создания наглядного рисунка.

Архитектура модуля отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов, представлена на рисунке 11.

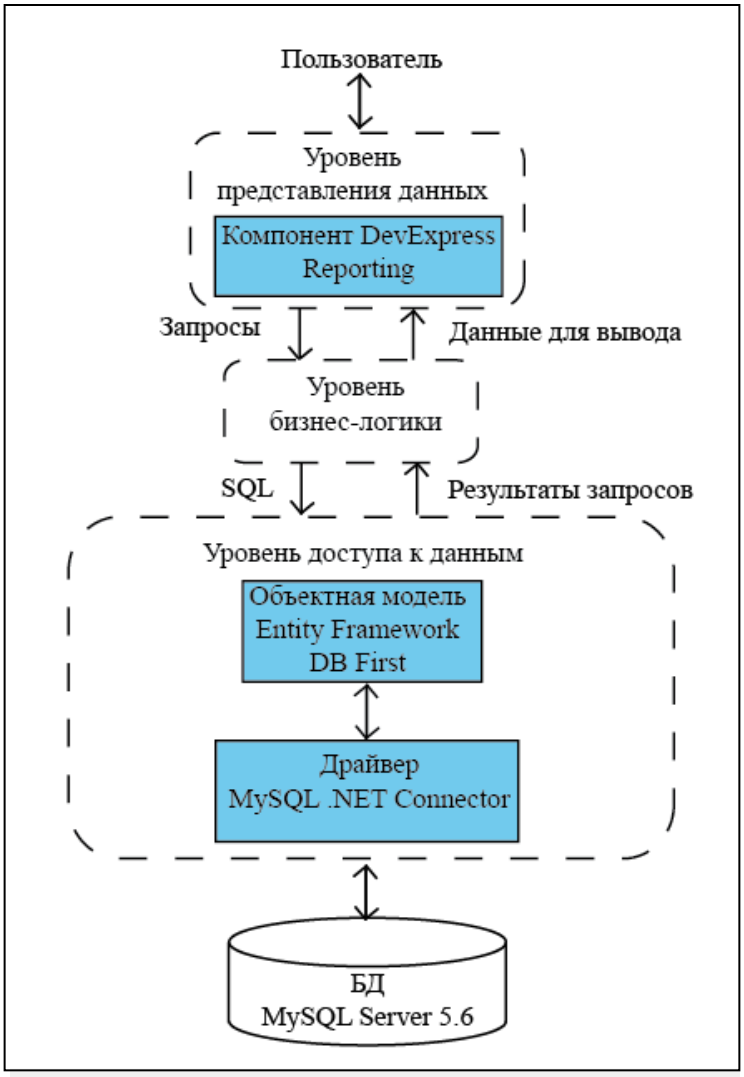


Рисунок 11 — Архитектура модуля отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов

Архитектура данного модуля является трехуровневой: уровень представления данных, уровень бизнес-логики и уровень доступа к данным. Пользователь взаимодействует с системой, реализующей бизнес-логику, посредством интерфейса программы, который создан с помощью компонентов компании DevExpress. Для работы с отчетными документами на уровне представления данных используется компонент Reporting.

На уровень доступа к данным, где используются объектная модель Entity Framework DB First и драйвер MySQL .NET Connector, при помощи которых модуль отчетности взаимодействует с базой данных, поступают запросы на языке SQL и после выборки запрашиваемых данных из БД пользователю возвращаются результаты этих запросов в удобочитаемом виде, то есть в виде отчета.

2.4 Процесс разработки модуля отчетности

Процесс разработки спроектированного модуля отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов, состоит из нескольких этапов.

Первым этапом разработки является настройка подключения к БД и создание модели EDMX в интегрированной среде Visual Studio при помощи объектно-ориентированной технологии Entity Framework. На данном этапе требуется установить необходимые компоненты и добавить их в созданный проект для того, чтобы получить доступ к данным из БД. Используя подход Database-First, требуется создать графическую EDMX-модель, отражающую имеющуюся модель базы данных системы учета электронных компонентов, предназначенных для ремонта электровозов. Благодаря мастеру обновления моделей ADO.NET модель EDMX также обладает способностью обновляться, если в подключенную базу данных внести какие-либо изменения. В созданную модель EDMX можно добавить новые объекты БД, отобразить внесенные изменения, а также удалить какие-либо объекты модели БД.

Сформированная модель добавляется в проект, после чего открывается для просмотра в конструкторе Entity Framework. Также, после создания EDMX-модели, в проект добавляется конфигурационный файл App.config, в котором указываются подробные сведения о подключении к БД.

Результат данного этапа наглядно представлен на рисунке 12.

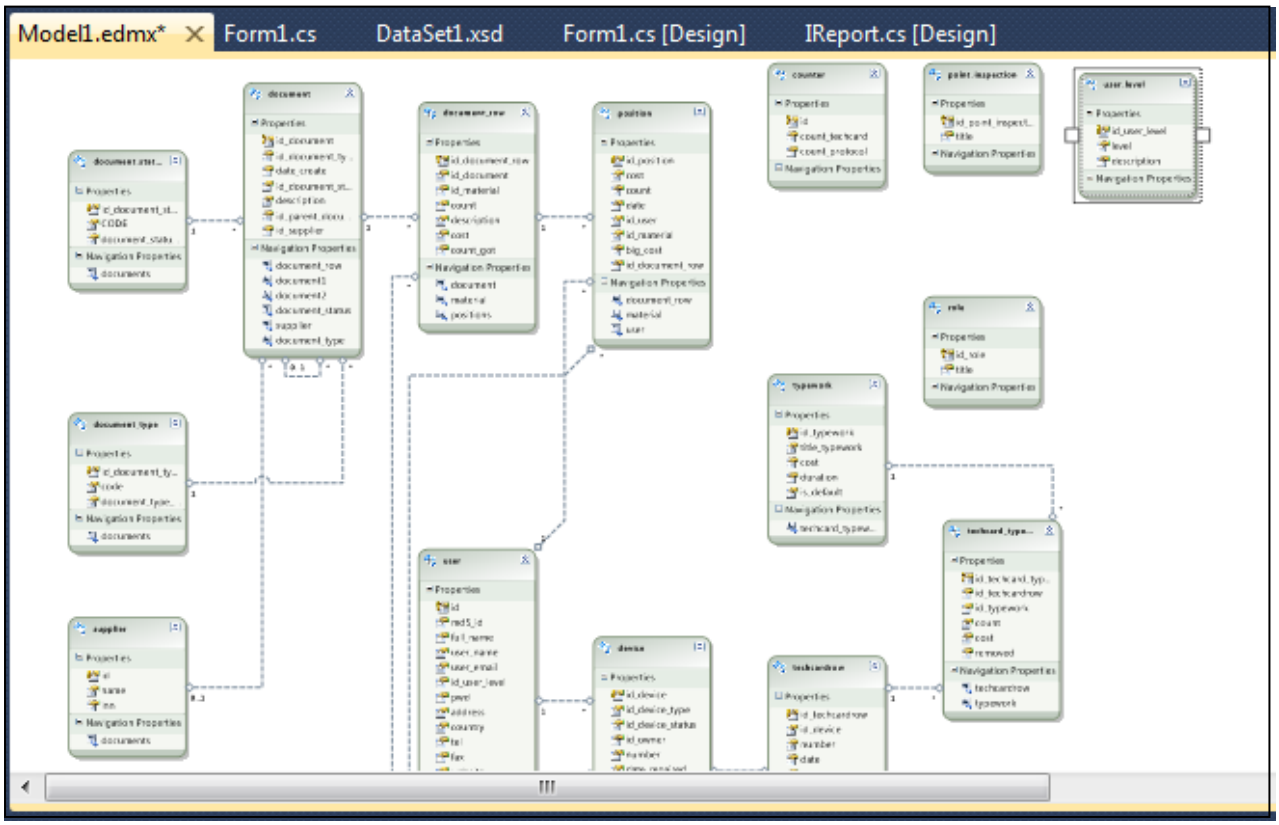


Рисунок 12 — Подключение БД к проекту и создание EDMX-модели имеющейся БД

Вторым этапом разработки модуля отчетности для системы учета электронных компонентов является генерация классов, которые необходимы для взаимодействия с БД и получения из нее различных данных. Технология Entity Framework создает набор сущностных классов, которые отражают модель имеющейся БД. Также создается класс ObjectContext, позволяющий взаимодействовать с БД, используя сущностную модель классов. Класс контекста позволяет создавать запросы и выполнять их, отслеживать различные изменения в объектах и отображать эти изменения на БД.

Пример одного из сгенерированных классов таблицы device представлен на рисунке 13.

```
reporting.device                                   device()
namespace reporting
{
    using System;
    using System.Collections.Generic;

    public partial class device
    {
        public device()
        {
            this.techcardrows = new HashSet<techcardrow>();
        }

        public int id_device { get; set; }
        public int id_device_type { get; set; }
        public int id_device_status { get; set; }
        public int id_owner { get; set; }
        public string number { get; set; }
        public System.DateTime date_repaired { get; set; }
        public System.DateTime date_receive { get; set; }
        public System.DateTime date_send { get; set; }
        public string description { get; set; }

        public virtual ICollection<techcardrow> techcardrows { get; set; }
        public virtual device_status device_status { get; set; }
        public virtual device_type device_type { get; set; }
        public virtual user user { get; set; }
    }
}
100 %
```

Рисунок 13 — Пример сгенерированного класса таблицы device из имеющейся базы данных

Третьим этапом работы с модулем для создания отчетных документов является разработка графического интерфейса данного модуля. Интерфейс разработан на основе платформы Windows Forms с помощью стандартных элементов управления и компонентов, а также дополнительных элементов, входящих в состав компонентов от компании DevExpress. При создании приложений элементы управления можно перетаскивать на поверхность визуального конструктора WinForms и посредством этих элементов, выстраивая их в нужном порядке и помещая в определенном месте конструктора, создавать графический интерфейс. Управлять свойствами данных компонентов можно в специальном окне Properties (Свойства).

Окно интерфейса для выбора отчетного документа модуля отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов, представлено на рисунке 14.

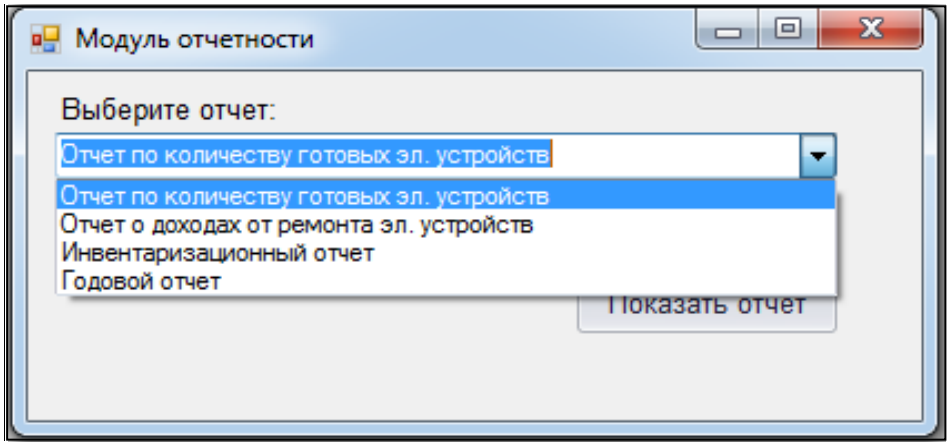


Рисунок 14 — Окно интерфейса для выбора отчетного документа модуля отчетности для системы учета электронных компонентов

Четвертым этапом разработки модуля является работа с компонентом Reporting от компании DevExpress. Для работы с данным программным продуктом к проекту требуется добавить инструмент XtraReport версии v11.1. После подключения необходимых таблиц из уже созданной и подключенной модели данных в будущий отчет можно добавлять столбцы и другие составляющие выходного документа. Одной из особенностей компонента Reporting является окно предпросмотра, где данные будут отображаться после написания необходимого кода.

Для того чтобы осуществить привязку элементов управления к данным из БД, которые в дальнейшем нужно перенести в конструктор отчетов XtraReports, необходимо создать таблицу с набором данных. Для того чтобы данные в конечном итоге отобразились на отчете, в созданном проекте необходимо прописать код для подключения и вывода данных на отчет.

Пример создания таблицы с набором данных из БД представлен ниже на рисунке 15.

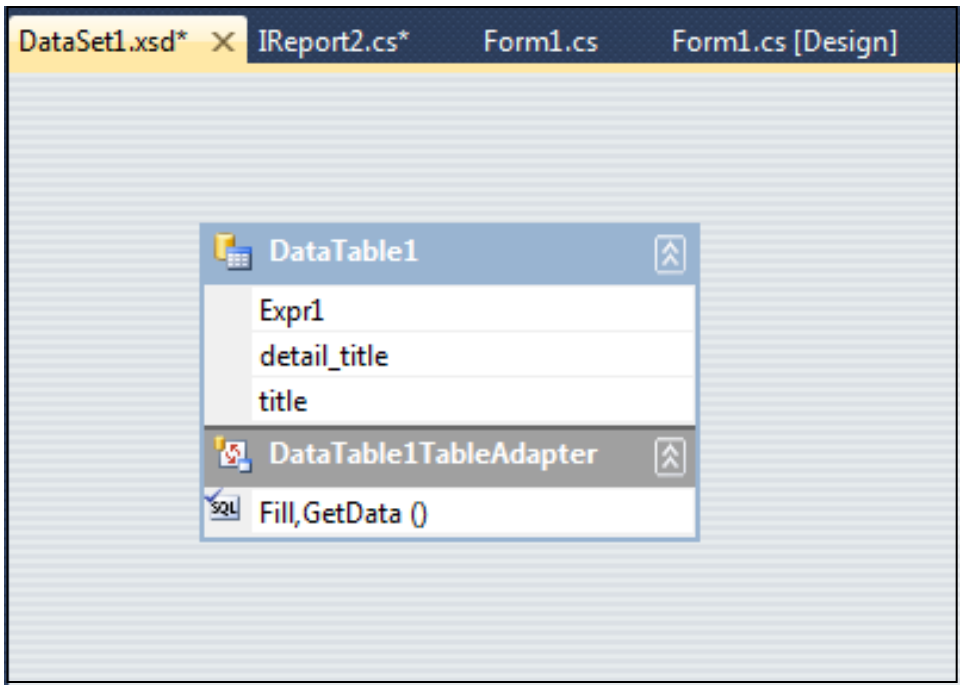


Рисунок 15 — Создание таблицы с набором данных из БД

Привязка данных из базы данных к элементам управления конструктора для создания отчетов XtraReports представлена на рисунке 16.

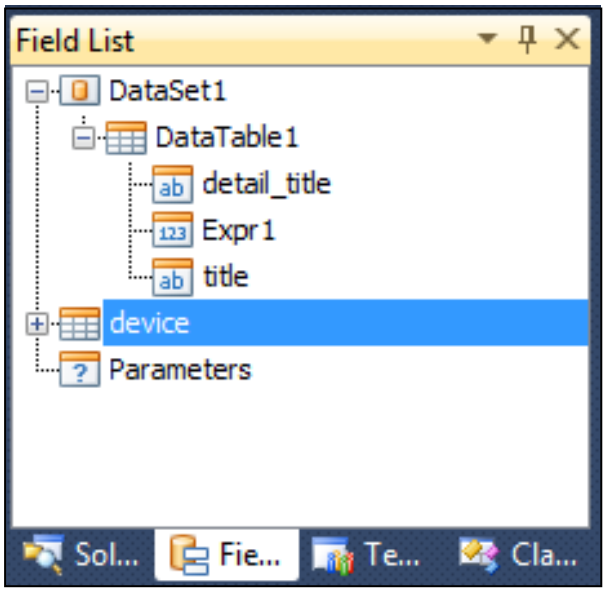


Рисунок 16 — Привязка данных из БД к элементам управления конструктора для создания отчетов XtraReports

Из рисунков, представленных выше, видно, что после создания таблицы с набором данных в специальном окне Field List (Список полей) появились необходимые поля для создания отчета.

Выборка данных, необходимых для создания отчета, из БД производится посредством SQL-запросов, которые можно писать вручную либо при помощи специального конструктора.

Пример SQL-запроса для отчета по количеству готовых электронных устройств представлен на рисунке 17.

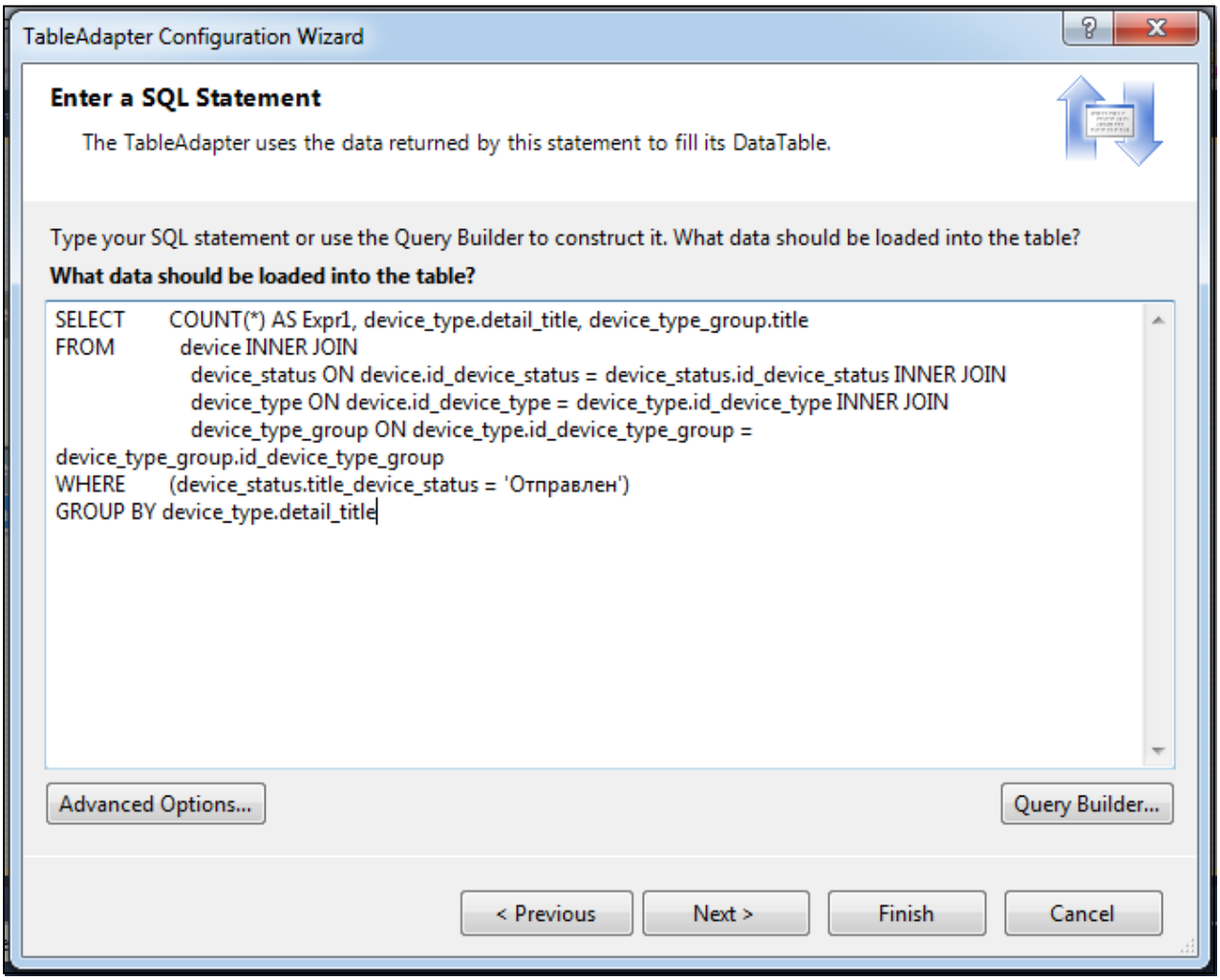


Рисунок 17 — Пример SQL-запроса для отчета по количеству готовых электронных устройств

Данный SQL-запрос позволяет подсчитывать по каждому отдельно взятому устройству количество отремонтированных, готовых к эксплуатации

электронных устройств. Данная выборка сгруппирована по названию электронного устройства МСУ.

Конструктор отчетов XtraReports предоставляет конечному пользователю возможность работать в режиме дизайнера отчетов, предлагая набор инструментов, который позволяет выстраивать положение отображающихся данных из базы данных, форматировать текст, изменять масштаб, создавать таблицы, суммировать необходимые данные, подсчитывать их количество или находить среднее, максимальное, минимальное значения. Данный инструмент также предоставляет и другие возможности.

Пример создания отчета по количеству готовых электронных устройств в режиме конструктора XtraReports представлен на рисунке 18.

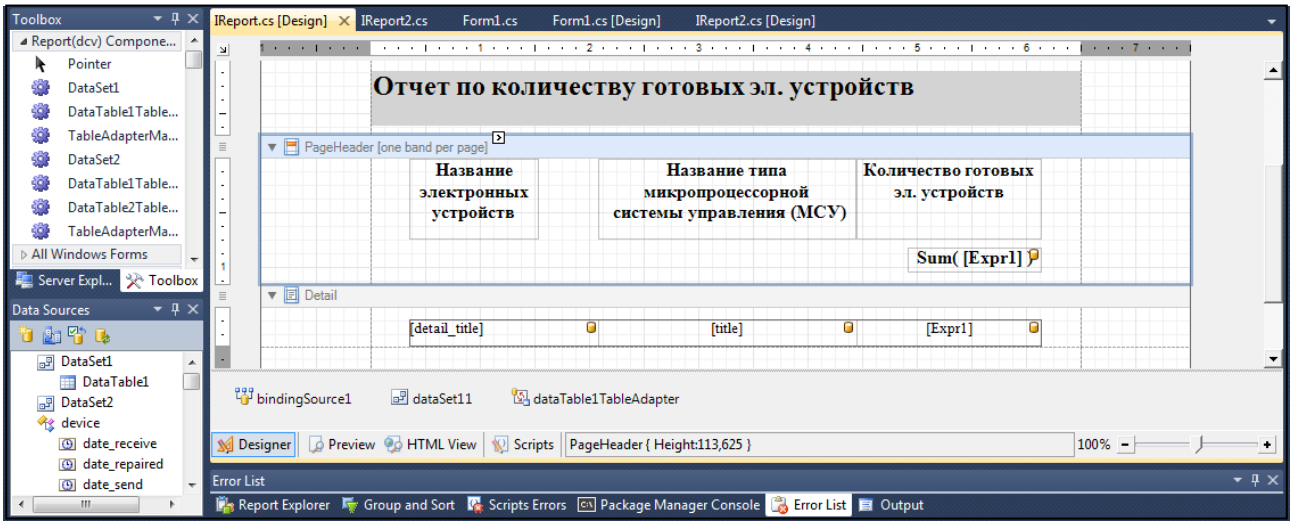


Рисунок 18 — Создание отчета по количеству готовых электронных устройств в режиме конструктора XtraReports

В отчете, представленном выше, также применяется инструмент SUM, который позволяет автоматически просуммировать количества подсчитанных данных по каждому устройству из SQL-запроса. Таким образом, на данном готовом отчете можно увидеть общее количество готовых электронных устройств МСУ.

Пример создания отчета о доходах от ремонта электронных устройств в режиме конструктора XtraReports представлен на рисунке 19.

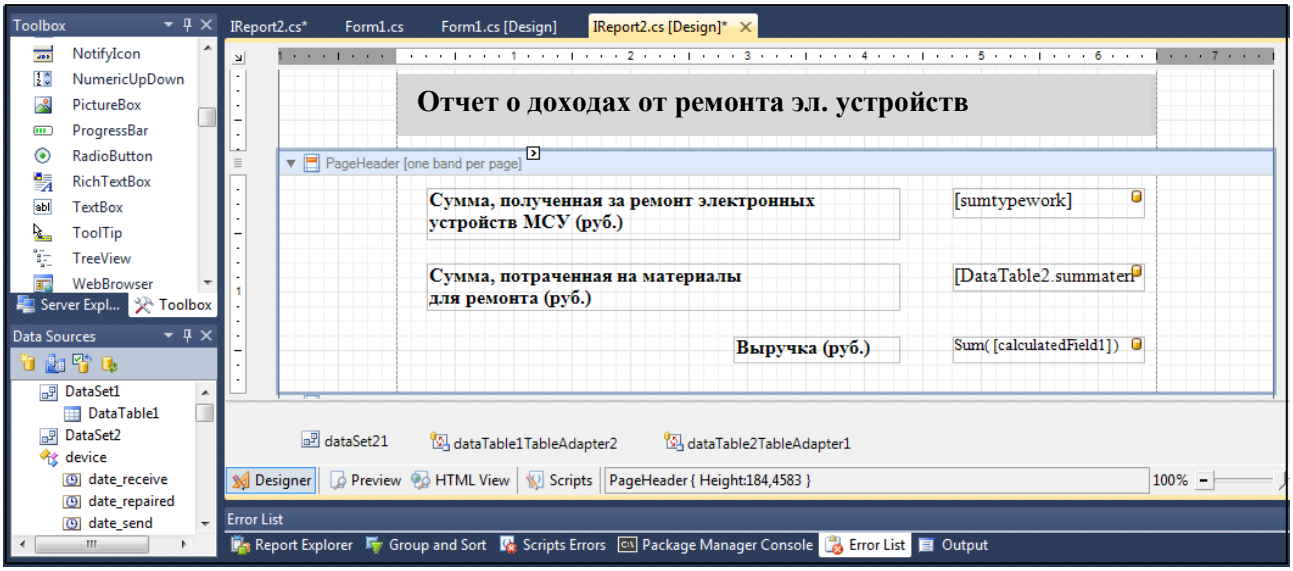


Рисунок 19 — Создание отчета о доходах от ремонта электронных устройств в режиме конструктора XtraReports

В отчете о доходах от ремонта электронных устройств используется подключение двух таблиц с наборами данных, которые содержат в себе SQL-запросы: для отображения суммы, полученной за ремонт электронных устройств МСУ, и суммы, потраченной на материалы для ремонта этих устройств. А также подсчитывается денежная выручка посредством алгебраических операций, произведенных над найденными значениями, с помощью инструмента calculatedField.

Итоговым результатом данной работы являются разработанные макеты отчетов, которые позволят сотрудникам «ДЦВ Красноярской железной дороги» вести контроль над общим количеством отремонтированных электронных устройств, а также над количеством по каждому такому устройству, принадлежащему определенному типу МСУ, в отдельности и над доходами от ремонта электронных устройств.

Готовый макет отчета по количеству готовых электронных устройств представлен на рисунке 20.

22.06.2016 13:25

Отчет по количеству готовых эл. устройств

Название электронных устройств	Название типа микропроцессорной системы управления (МСУ)	Количество готовых эл. устройств
		664
	МСУД1.1(3)	324
Блок ввода - вывода дискретных сигналов	БАУВИП-208	59
блок ввода-вывода дискретных сигналов	МСУД-Н02	1
Блок входных сигналов	БУ-199	17
Блок выходных усилителей	БУ-199	35
Блок индикации	БИ1	2

Рисунок 20 — Готовый макет отчета по количеству готовых электронных устройств

Пример готового отчета о доходах от ремонта электронных устройств представлен на рисунке 21.

22.06.2016 9:49

Отчет о доходах от ремонта эл. устройств

Сумма, полученная за ремонт электронных устройств МСУ (руб.)	89543,30
Сумма, потраченная на материалы для ремонта (руб.)	76143,90
Выручка (руб.)	13399,40

Рисунок 21 — Пример готового отчета о доходах от ремонта электронных устройств МСУ

Данный модуль также позволяет сохранять готовые отчеты в любом из предложенных форматов, производить поиск по определенным параметрам. Благодаря окну редактирования конечный пользователь имеет возможность изменить оформление сформированного отчета. Возможности компонента Reporting представлены на рисунках ниже 22, 32 и 24.

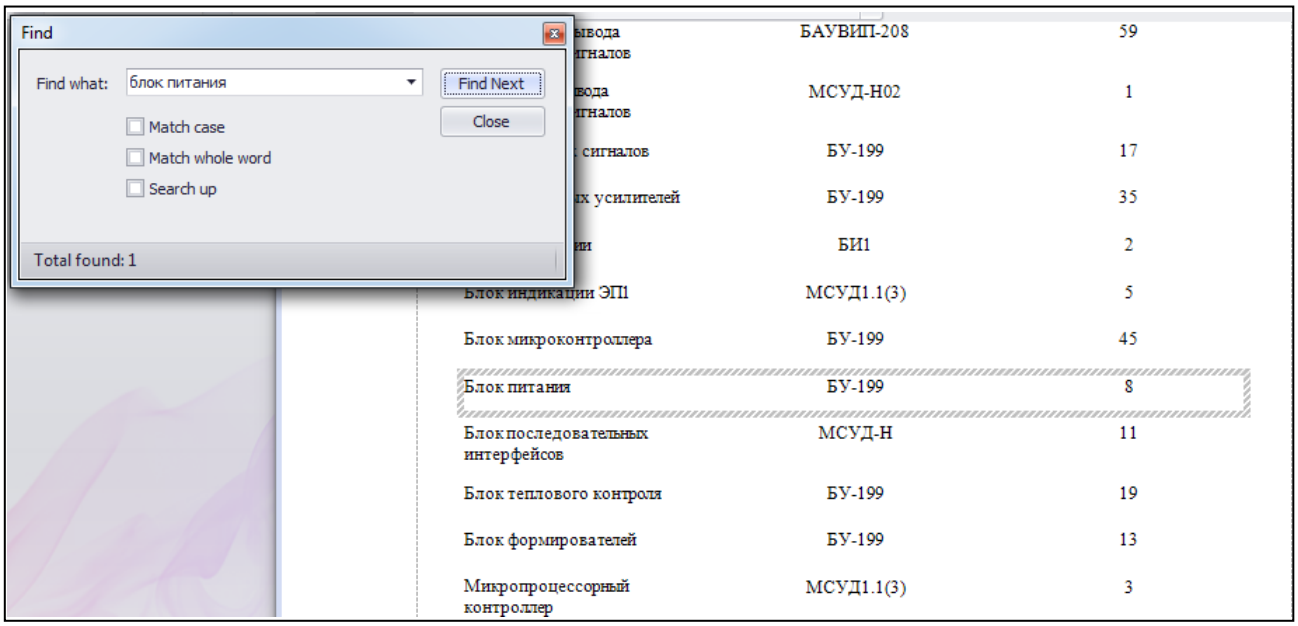


Рисунок 22 — Поиск в отчете по названию

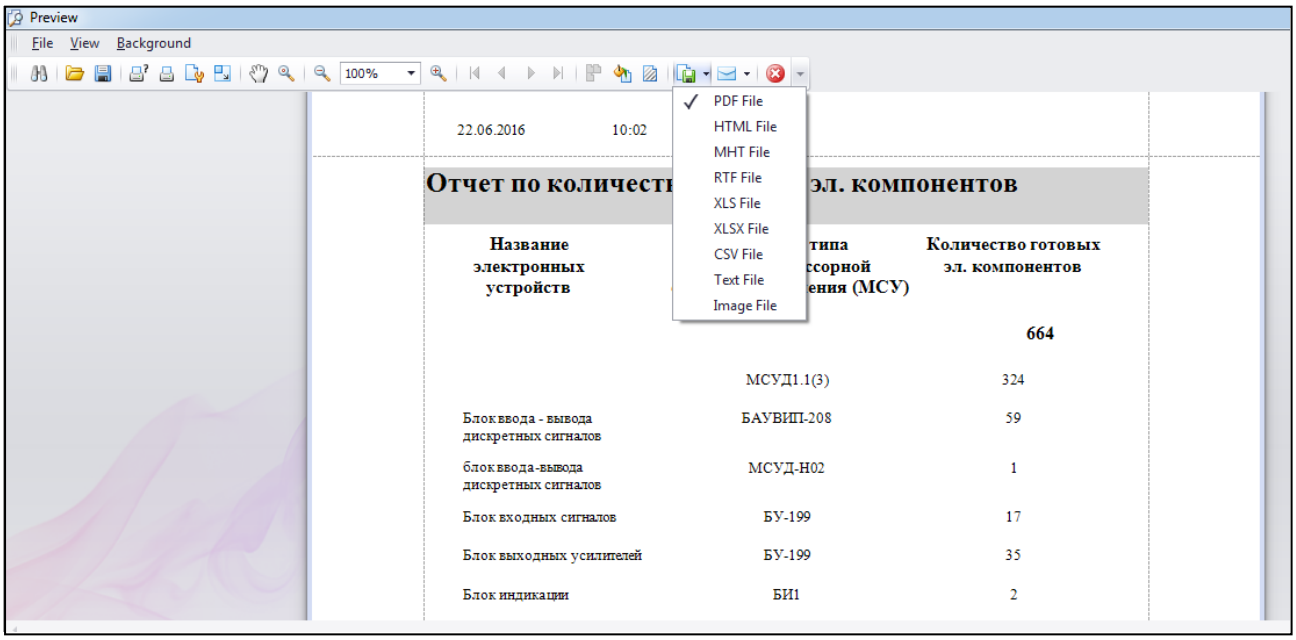


Рисунок 23 — Сохранение отчета в предложенных форматах

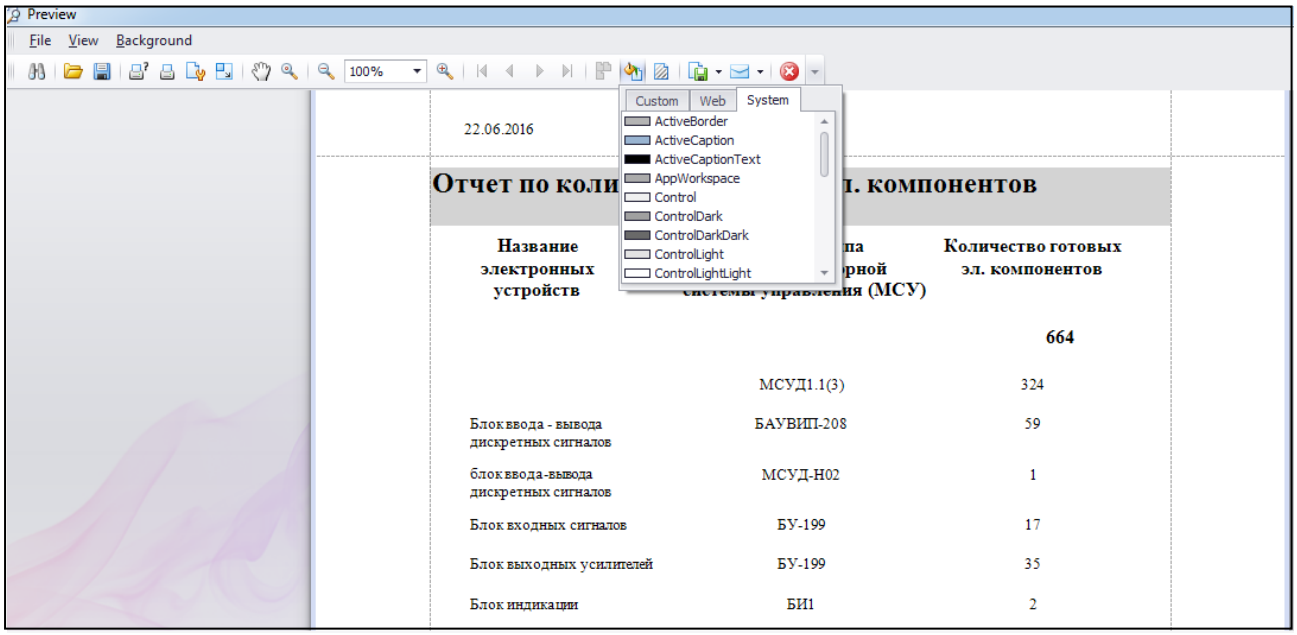


Рисунок 24 — Редактирование оформления отчета конечным пользователем

2.5 Вывод к главе 2

В ходе анализа проведенного интервью сформулированы определенные требования к разрабатываемому модулю. На этапе проектирования построена архитектура, наглядно демонстрирующая концепцию, структуру, выполняемые функции и взаимосвязь компонентов модуля отчетности.

В процессе разработки было создано подключение к базе данных программы при помощи современной объектно-ориентированной технологии Entity Framework, по модели БД сгенерированы необходимые классы. На основе платформы WinForms был создан графический интерфейс модуля отчетности. При помощи компонента Reporting в конструкторе XtraReports созданы готовые макеты отчетов, позволяющие осуществлять контроль над отремонтированными электронными устройствами, предназначенными для электровозов, и над доходами от ремонта данных электронных устройств. Выборка необходимых данных из базы данных осуществлялась посредством SQL-запросов.

Данный модуль позволяет создавать новые отчеты, меняя запросы к БД и дизайны макетов.

ЗАКЛЮЧЕНИЕ

В ходе данной работы проведен анализ существующей программы учета электронных компонентов. АРМ ремонта МСУ является одним из проектов ОАО «РЖД», который осуществляется организацией «ДЦВ Красноярской железной дороги» под руководством генерального директора Семченко В. В. Представлены диаграммы, наглядно описывающие процесс движения поступающих в заказ электронных устройств для микропроцессорных систем управления локомотивов. А также представлена архитектура системы.

Исходя из выявленных проблем в области представления данных, сделан вывод, что крупные предприятия и организации, имеющие АИС, больше делают акцент на хранение данных, а не на их представление в готовом виде для конечного пользователя. Информация, представленная некорректно, трудна для восприятия и делает анализ и обработку отчетной информации неточными, увеличивая риск возникновения ошибок в итоговых результатах. Для решения этой проблемы «ДЦВ Красноярской железной дороги» было принято решение создать модуль отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов.

Проведен обзор существующих программных продуктов для создания отчетов, которые используются подавляющим большинством разработчиков. Учитывая функционал, который позволяет решить поставленные задачи по разработке модуля отчетности для АРМ ремонта МСУ, в процессе выявления требований заказчика был выбран компонент Reporting от компании DevExpress как наиболее подходящий.

Разработан модуль отчетности, который впоследствии будет изменяться и дорабатываться в соответствии с дополнительными требованиями заказчиков.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1) DevExpress [Электронный ресурс]: Reporting for the .NET Framework – Режим доступа: https://www.devexpress.com

2) Fast Reports [Электронный ресурс]: FastReport.Net – Режим доступа: https://www.fastreport.ru

3) Ixbt.com [Электронный ресурс]: Определение отчета – Режим доступа: http://www.ixbt.com

4) Metanit.com – Сайт о программировании [Электронный ресурс]: Руководство по Entity Framework – Режим доступа: http://metanit.com

5) MySQL [Электронный ресурс]: Справочное руководство по MySQL – Режим доступа: http://www.mysql.ru

6) Odant [Электронный ресурс]: Работа с XtraReports – Режим доступа: http://wiki.odant.ru

7) Professorweb [Электронный ресурс]: Entity Framework 6 – Режим доступа: http://professorweb.ru

8) Алан Бьюли Изучаем SQL : учебное пособие / Алан Бьюли – Санкт-Петербург : Символ-Плюс, 2007. – 312 с.

9) Андерс Хейлсберг Язык программирования C# : учебное пособие / Андерс Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд — Санкт-Петербург : Питер, 2012. — 784 с.

10) Герберт Шилдт Полный справочник по C# : учебное пособие / Герберт Шилдт – Москва : Вильямс, 2011. – 748 с.

11) Документация DevExpress [Электронный ресурс]: Reporting – Режим доступа: https://documentation.devexpress.com

12) Дорожный центр внедрения Красноярской железной дороги [Электронный ресурс] Микропроцесорная система управления электровозов переменного тока — Режим доступа: http://dcv.ru

13) Дэвид Мак-Амис Профессиональная работа с Crystal Reports для Microsoft Visual Studio .NET : учебное пособие / Дэвид Мак-Амис – Москва : Вильямс, 2003. – 352 с.

14) Мартин Фаулер Архитектура корпоративных программных приложений : учебное пособие/ Мартин Фаулер – Москва : Издательский дом «Вильяме», 2006. – 541 с.

15) Мэтью Мак-Дональд Отчеты Access : Мэтью Мак-Дональд – Санкт-Петербург : БХВ-Питербург, 2009. – 50 с.

16) Общие сведения об отчетах [Электронный ресурс]: Microfoft Access 2010. – Режим доступа: https://support.office.com

17) Отраслевой центр внедрения новой техники и технологии [Электронный ресурс] – Режим доступа: http://new.ocv.ru

18) Прохоренок Н. А. HTML, JavaScript, PHP и MySQL. Джентельменский набор Web-мастера: учебное пособие / Н. А Прохоренок – Санкт-Петербург : БХВ-Петербург, 2010. – 900 с.

19) Руководство программиста АРМ МСУ : руководство пользователя. – Красноярск, 2014. – 48 с.

20) Сайт для программистов [Электронный ресурс]: – Режим доступа: http://www.realcoding.net

21) СТО СФУ 4.2-07-2014 Стандарт организации. Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности : стандарт организации – Красноярск, 2014. – 60 с.

22) Хабрхабр [Электронный ресурс]: DXperience v2011 vol 1 – Новая версия .NET компонентов от DevExpress – Режим доступа: http://habrahabr.ru

23) Энтони Молинаро SQL. Сборник рецептов : учебное пособие / Энтони Молинаро – Санкт-Петербург : Символ-Плюс, 2009. – 672 с.

ПРИЛОЖЕНИЕ А

Плакаты презентации

ФГАУ ВО «СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра систем искусственного интеллекта

БАКАЛАВРСКАЯ РАБОТА
Разработка модуля отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов

Руководитель: Е.А. Мальцев
Выпускник: Н.А. Юматова

г. Красноярск 2016

Рисунок А.1 — Слайд презентации №1

Цель:

- Разработать модуль отчетности для системы учета электронных компонентов, предназначенных для ремонта электровозов

Задачи:

- анализ имеющейся программы и обзор аналогов компонентов для создания отчетов;
- проектирование модуля отчетности для системы учета электронных компонентов;
- разработка модуля для данной системы

2/17

Рисунок А.2 — Слайд презентации №2

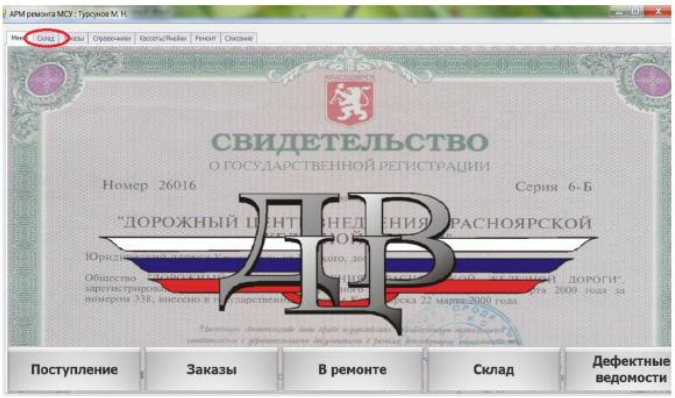
Актуальность

С увеличением количества печатных документов, а также усложнением их форм для ЗАО «ДЦВ Красноярской ЖД» необходимо решить задачу создания модуля, который позволит методами визуального программирования создавать отчеты, интегрируя необходимую информацию из БД

3/17

Рисунок А.3 — Слайд презентации №3

АРМ Ремонта МСУ



Основное назначение разработанной системы – выполнение учета ремонта МСУ и отдельных электронных устройств, из которых состоят МСУ

4/17

Рисунок А.4 — Слайд презентации №4

Рисунок А.5 — Слайд презентации №5

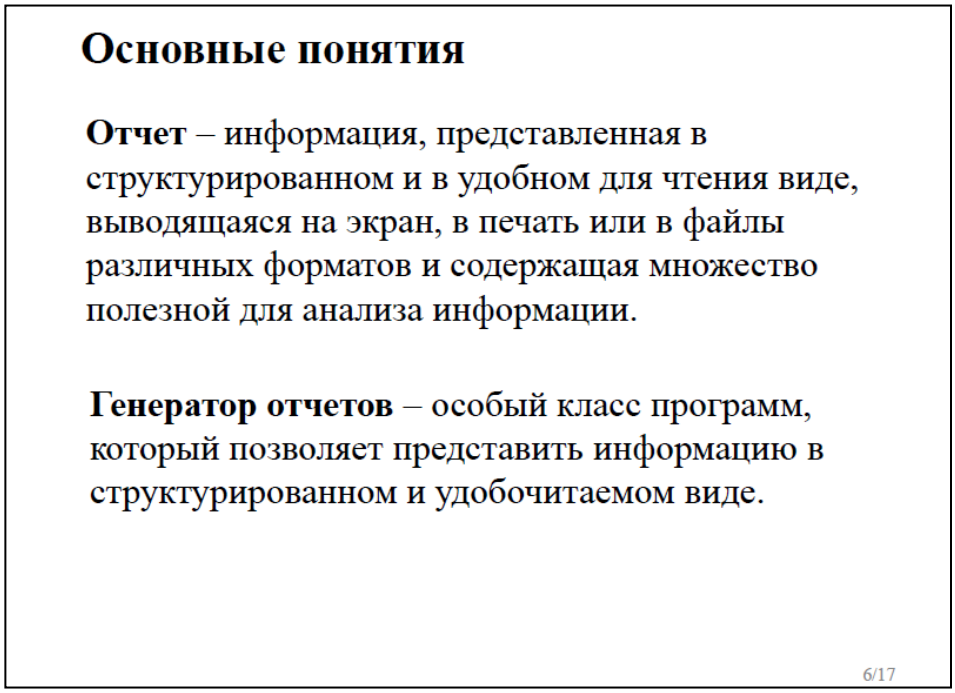


Рисунок А.6 — Слайд презентации №6

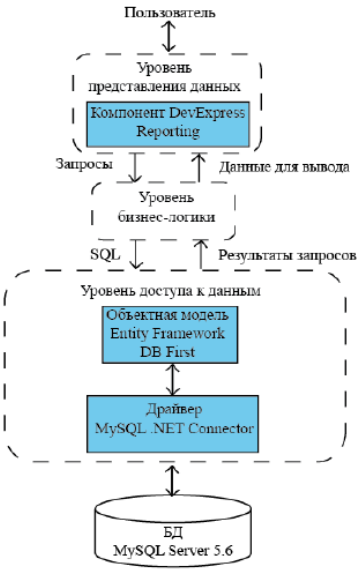
Сравнительная таблица аналогов компонентов для построения отчетов

Возможности	Название компонента отчетности			
	Crystal Reports.NET (For Visual Studio(VS))	Access (Microsoft Corp.)	FastReport.Net (Fast Reports, Inc)	Reporting (Developer Express Inc.)
Поддержка СУБД	MS SQL Server, MS Access, Oracle, MySql	Access	MS SQL Server, MS Access, Oracle, DB2, PostgreSQL, MySQL и др.	MS SQL Server, PostgreSQL, MS Access, Oracle, DB2, MySQL и др.
Представление данных в виде графических изображений	Графики, диаграммы, гистограммы	Графики, диаграммы	Диаграммы, гистограммы, графики, линии, фигуры	Графики, диаграммы, гистограммы, линии, фигуры
Совместимость с Visual Studio	Совместим	Не предназначен для работы с VS	Совместим	Совместим
Автоматический подсчет итогов	Сумма, среднее значение, max, min	Отсутствует	Сумма, среднее значение, max, min	Сумма, среднее значение, max, min
Содержание стандартных файлов кода	Не содержит	Не содержит файлы кода	Содержит только в версии Professional	Содержит
Форматы составления отчетов	PDF, Word, Excel, RTF, HTML, Plain Text и др.	Word, Excel	Word, Excel, PDF, HTML, Plain Text, RTF и др.	Word, PDF, Excel, Plain Text, HTML, RTF и др.

7/17

Рисунок А.7 — Слайд презентации №7

Архитектура модуля отчетности



8/17

Рисунок А.8 — Слайд презентации №8

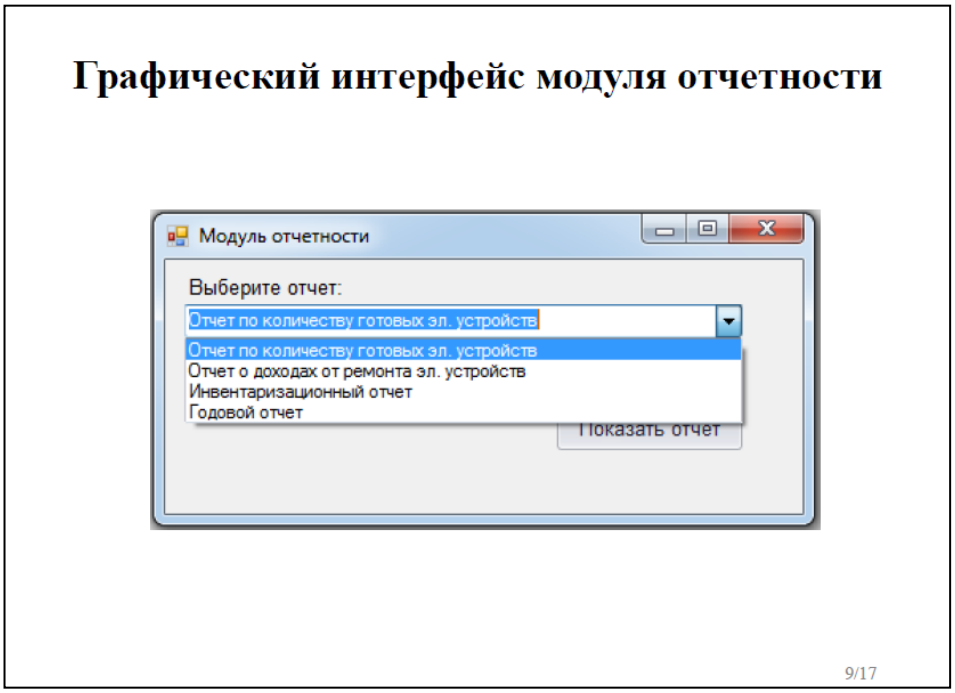


Рисунок А.9 — Слайд презентации №9

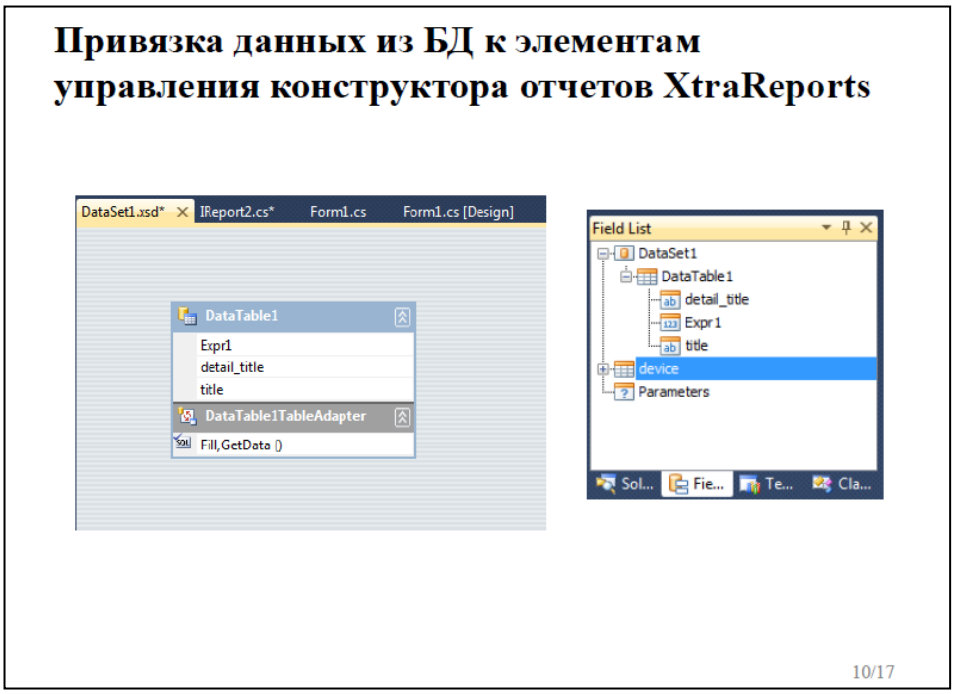


Рисунок А.10 — Слайд презентации №10

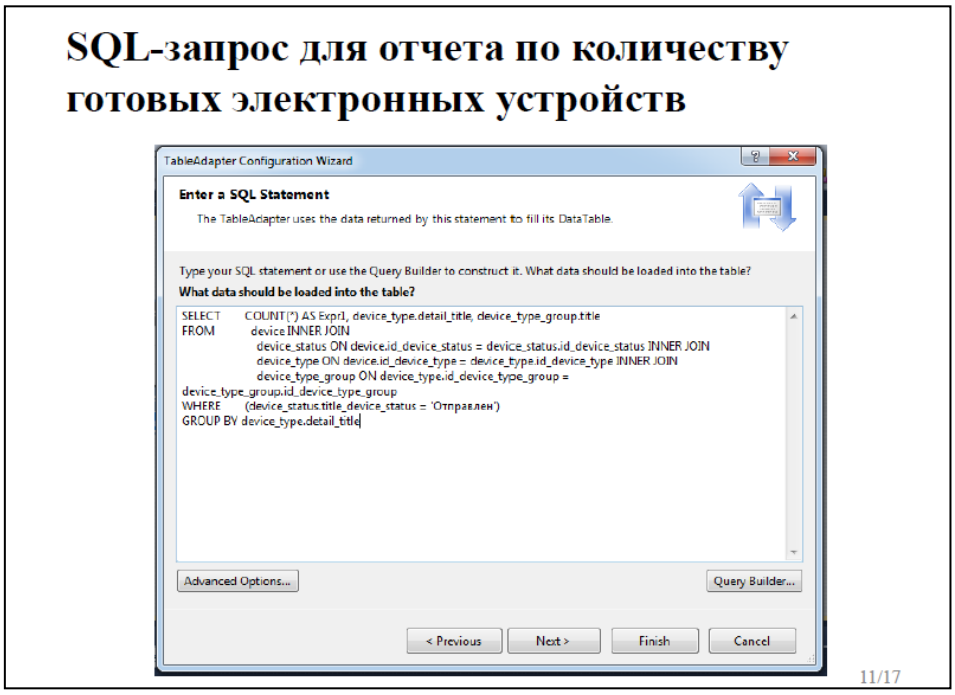


Рисунок А.11 — Слайд презентации №11

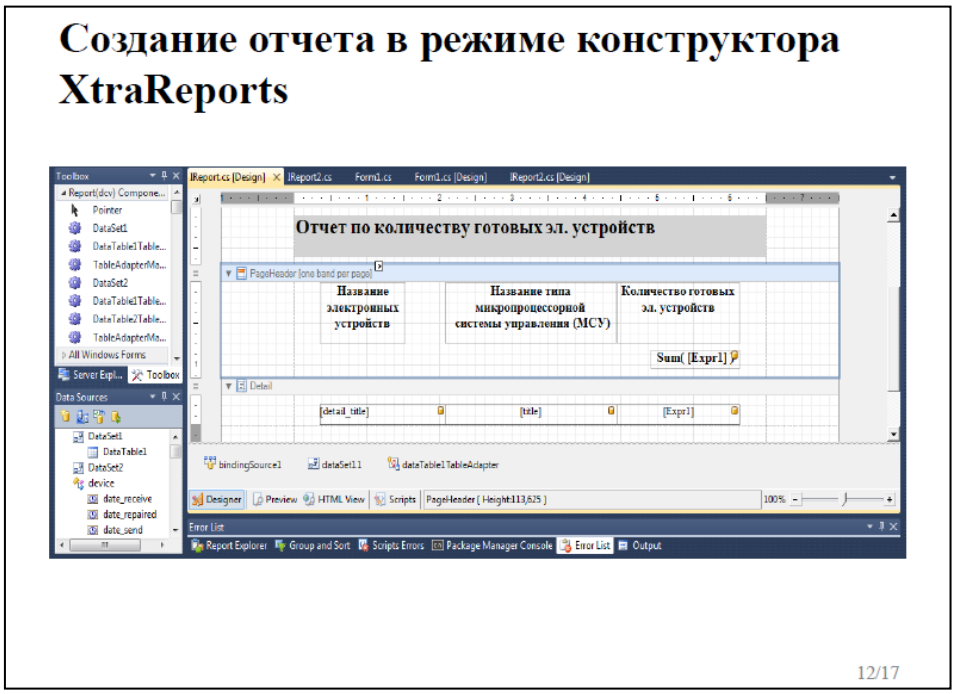


Рисунок А.12 — Слайд презентации №12

Итоговые результаты

22.06.2016 13:25

Отчет по количеству готовых эл. устройств

Название электронных устройств	Название типа микропроцессорной системы управления (МСУ)	Количество готовых эл. устройств
		664
	МСУД1.1(3)	324
Блок ввода - вывода дискретных сигналов	БАУВИП-208	59
блок ввода-вывода дискретных сигналов	МСУД-Н02	1
Блок входных сигналов	БУ-199	17
Блок выходных усилителей	БУ-199	35
Блок индикации	БИ1	2

13/17

Рисунок А.13 — Слайд презентации №13

Итоговые результаты (продолжение)

22.06.2016 9:49

Отчет о доходах от ремонта эл. устройств

Сумма, полученная за ремонт электронных устройств МСУ (руб.)	89543,30
Сумма, потраченная на материалы для ремонта (руб.)	76143,90
Выручка (руб.)	13399,40

14/17

Рисунок А.14 — Слайд презентации №14

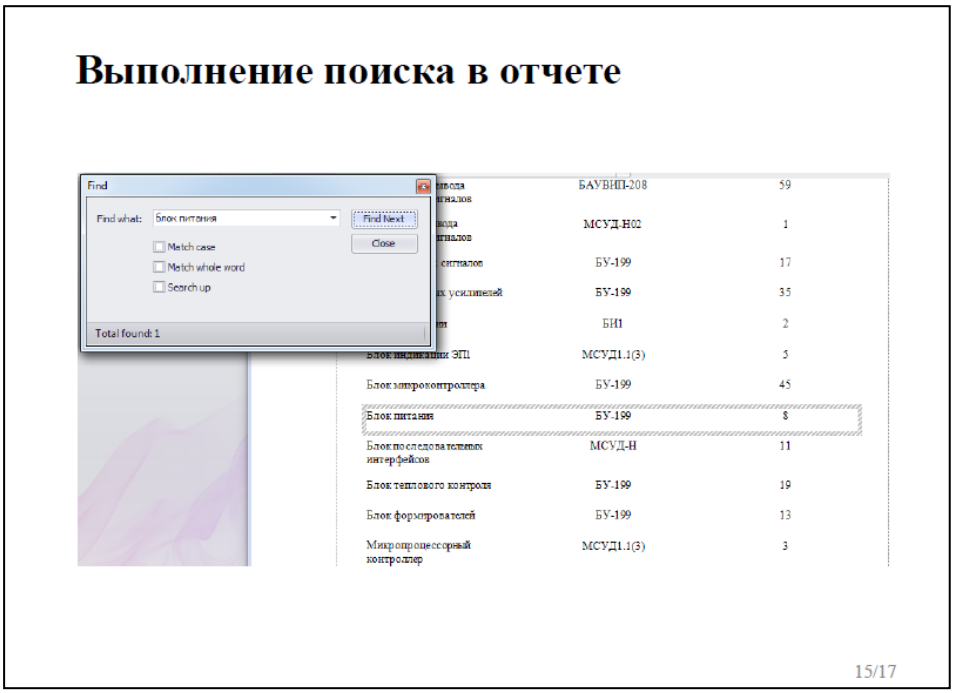


Рисунок А.15 — Слайд презентации №15

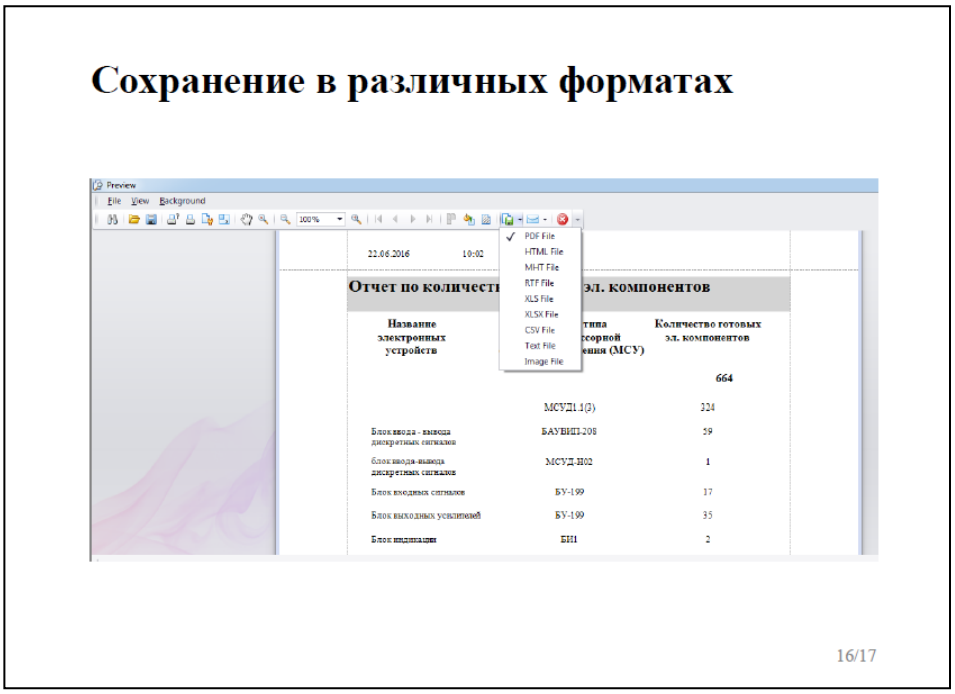


Рисунок А.16 — Слайд презентации №16

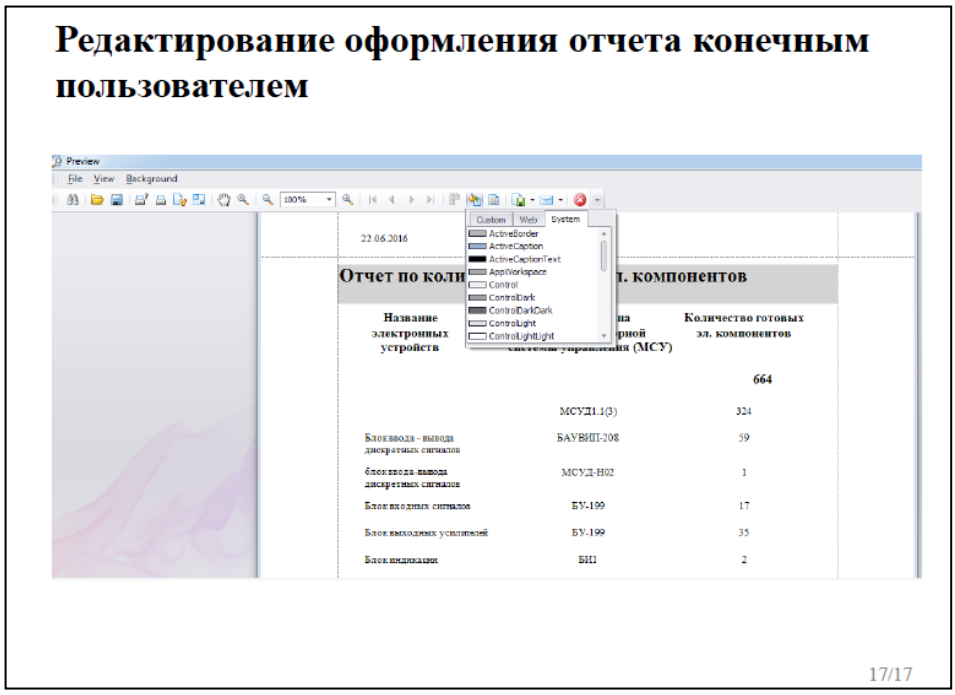


Рисунок А.17 — Слайд презентации №17