

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ»**
(Н И У « Б е л Г У »)

ИНСТИТУТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ И ЕСТЕСТВЕННЫХ НАУК

Кафедра прикладной информатики и информационных технологий

**Разработка системы хранения корпоративного контента на основе
использования открытого программного обеспечения для
ООО «Касталламаре»**

Выпускная квалификационная работа

**студентки очной формы обучения
направления подготовки 09.03.03. Прикладная информатика
4 курса группы 07001205
Аушевой Натальи Сергеевны**

Научный руководитель
ст. пр. Пусная О.П.

БЕЛГОРОД 2016

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1 АНАЛИТИЧЕСКАЯ ЧАСТЬ.....	6
1.1 Технико-экономическая характеристика предметной области.....	6
1.1.1 Характеристика предприятия	6
1.1.2 Краткая характеристика видов деятельности	7
1.2 Экономическая сущность задачи	16
1.3 Обоснование необходимости и цели создания системы.....	16
1.4 Постановка задачи	17
2 ОБОСНОВАНИЕ ВЫБОРА ОСНОВНЫХ ПРОЕКТНЫХ РЕШЕНИЙ	19
2.1 Обоснование проектных решений по техническому обеспечению.....	19
2.2 Обоснование проектных решений по информационному обеспечению	22
2.3 Обоснование проектных решений по программному обеспечению	25
2.4 Обоснование выбора программных средств	29
3 ПРОЕКТНАЯ ЧАСТЬ.....	33
3.1 Разработка информационных моделей.....	33
3.2 Информационное обеспечение задачи.....	35
3.3 Программное обеспечение задачи.....	38
3.3.1 Структура программного приложения	38
3.3.2 Описание интерфейса.....	40
3.4 Описание контрольного примера реализации проекта.....	44
3.5 Целесообразность разработки с экономической точки зрения	51
3.6 SWOT-анализ разработки.....	58
ЗАКЛЮЧЕНИЕ	62
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	64
ПРИЛОЖЕНИЯ	68
ПРИЛОЖЕНИЕ А.....	69
ПРИЛОЖЕНИЕ Б.....	70

ВВЕДЕНИЕ

Аналитики и ученые все чаще выдвигают предположения о том, что объем разного рода информации в наше время увеличивается едва ли не в геометрической прогрессии. Это напрямую связано с прорывами во многих областях знаний. Научно технический прогресс играет весьма важную роль в экономическом развитии, благодаря чему улучшаются и развиваются какие-либо смежные области жизнедеятельности, в том числе абсолютно разношерстные организации, компании или учреждения.

Стремительное увеличение объемов неструктурированной информации – веское препятствие на пути развития современных предприятий любого типа. Численность различных документов, таблиц, диаграмм, графических изображений, фотографий, аудио и видео файлов увеличивается с удивительной скоростью. Доля неструктурированной информации часто составляет около 80% от всех корпоративных сведений. В данной ситуации пользователям вполне могут прийти на помощь современные программные средства. Требования к программам для организации хранения и систематизации данных разнообразного типа могут в конечном итоге отличаться, но главная цель для пользователя всегда одна и та же – возможность быстро рассортировать нужные материалы, не смотря на нынешнее многообразие файловых расширений.

Целью выпускной квалификационной работы является разработка системы хранения корпоративного контента на основе открытого программного обеспечения для ООО «Касталламаре», работающего в сфере розничных продаж.

Актуальность темы обусловлена наличием больших объемов неструктурированных данных, которые представлены в различных форматах. Именно это создает трудности с обработкой информации, ее хранением и систематизацией.

Для достижения поставленной цели необходимо решить следующие задачи:

- изучение предметной области;
- обоснование необходимости разработки;
- выработка концепции по оптимизации управления неструктурированными данными, ориентированной на перспективные цели;
- проведение анализа основных проектных решений для реализации;
- выбор средств и технологий для разработки;
- создание системы обработки и распределения массива информации различных типов и расширений;
- тестирование системы хранения корпоративного контента;
- расчет эффективности разработки.

Объектом исследования является информационный отдел ООО «Касталламаре», а предметом – выполнение работ по систематизации крупных массивов данных.

В первой части работы проводится краткий обзор предприятия и его видов деятельности. Рассматриваются основные технико-экономические характеристики компании, выясняется необходимость и цели создания системы. Выделяются основные задачи.

Во второй части рассматриваются методы и средства решения поставленных задач. Принимаются решения по технологическому, информационному и программному обеспечению. Аргументируется выбор программных средств для разработки проекта.

В третьей части представляется проектное решение и виды его обеспечения. С помощью выбранных программных средств реализуется программное приложение и проводится наглядное тестирование для подтверждения корректности работы системы.

В выпускной квалификационной работе используются 28 рисунков и 10 таблиц.

1 АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1 Технико-экономическая характеристика предметной области

1.1.1 Характеристика предприятия

Общество с ограниченной ответственностью «Касталламаре» зарегистрировано в городе Белгороде. Основным видом деятельности компании является розничная торговля по заказам. Также работает еще по 9 направлениям.

Компания «Касталламаре», не смотря на свое не слишком долгое существование, уже превосходно зарекомендовала себя на рынке розничной торговли.

Предприятие имеет широкий спектр видов деятельности, но в основном занимается поставками запчастей для ремонта сотовых телефонов и КПК, а также аксессуаров непосредственно от крупнейших производителей данной продукции. Материалы и комплектующие закупаются по оптовым ценам главным офисом в столице, что обеспечивает торговлю по ценам ниже рыночных. Любая продукция перед поставкой проходит тестирование, что делает процент брака близким к нулю. Также действует доставка по всей России.

Торговля по заказам дает определенные преимущества над сервисами и магазинами. Во-первых, различное стоимостное выражение товаров при одинаковом качестве (поставщики, как правило, одни и те же). Во-вторых, ассортимент гораздо богаче, хоть и не всегда вся продукция имеется в наличии. Кроме того, на некоторые товары дается гарантия. Договоренность на официальной основе с офисом в Москве, который работает напрямую с крупными поставщиками из разных стран, дает компании неоспоримый приоритет над конкурентами, работающими в той же области.

1.1.2 Краткая характеристика видов деятельности

Целью компании ООО «Касталламаре», как и любого предприятия, является деятельность по получению финансов. Имеет девять официальных направлений деятельности для привлечения денежных средств.

ООО «Кастелламаре» предоставляет следующие услуги:

- розничная торговля по заказам;
- торговля в неспециализированных магазинах;
- розничная торговля бытовыми электротоварами, радио- и телеаппаратурой;
- техническое обслуживание и ремонт офисных машин и вычислительной техники.

Самым развитым, прибыльным и популярным видом деятельности предприятия является розничная торговля по заказам. Диаграммы для данного направления работы разработаны в среде AllFusion Process Modeler (ранее BPwin). Это инструментальное средство для визуального моделирования, детального исследования и реорганизации бизнес-процессов.

Диаграммы построены в графических нотациях IDEF0 и IDEF3. IDEF0 – методология функционального моделирования. Благодаря наглядному графическому представлению IDEF0, анализируемую область можно представить как совокупность взаимосвязанных функций. Моделирование в методологии IDEF0 зачастую становится стартовым пунктом анализа разнообразных систем.

Функциональный блок (Activity Box) является одним из базовых понятий методологии IDEF0. Визуально изображается в виде прямоугольника и представляет собой конкретно обозначенную функцию в рамках рассматриваемой системы.

Каждая из четырех сторон функционального блока имеет своё определенное значение (роль), при этом:

- Верхняя сторона имеет значение “Управление” (Control);
- Левая сторона имеет значение “Вход” (Input);
- Правая сторона имеет значение “Выход” (Output);
- Нижняя сторона имеет значение “Механизм” (Mechanism).

Каждый функциональный блок в рамках единой рассматриваемой системы должен иметь свой уникальный идентификационный номер.

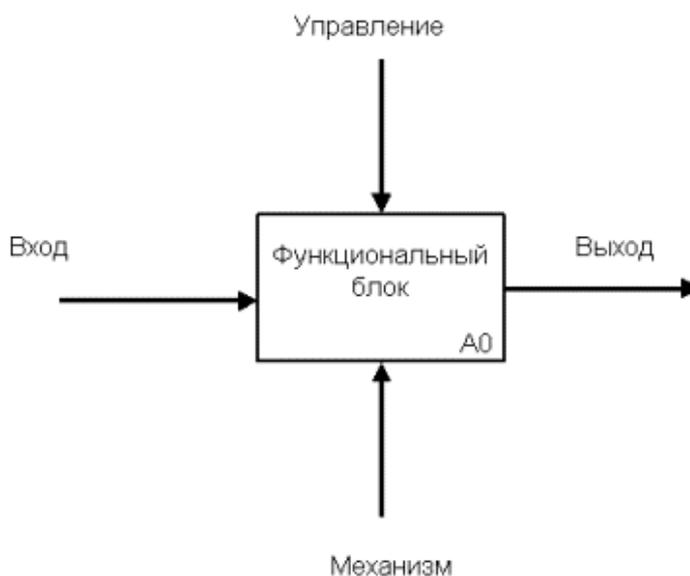


Рисунок 1.1 – Функциональный блок

Вторым важным элементом методологии IDEF0 является понятие интерфейсной дуги (Arrow). Также интерфейсные дуги часто называют потоками или стрелками. Интерфейсная дуга отображает элемент системы, который обрабатывается функциональным блоком или оказывает иное влияние на функцию, отображенную данным функциональным блоком.

Графическим отображением интерфейсной дуги является однонаправленная стрелка. Каждая интерфейсная дуга должна иметь свое уникальное наименование (ArrowLabel). По требованию стандарта, наименование должно быть оборотом существительного.

С помощью интерфейсных дуг отображают различные объекты, в той или иной степени определяющие процессы, происходящие в системе. Такими

объектами могут быть элементы реального мира или потоки данных и информации. [3]

На рисунке 1.2 представлена контекстная диаграмма Управление работой предприятия.

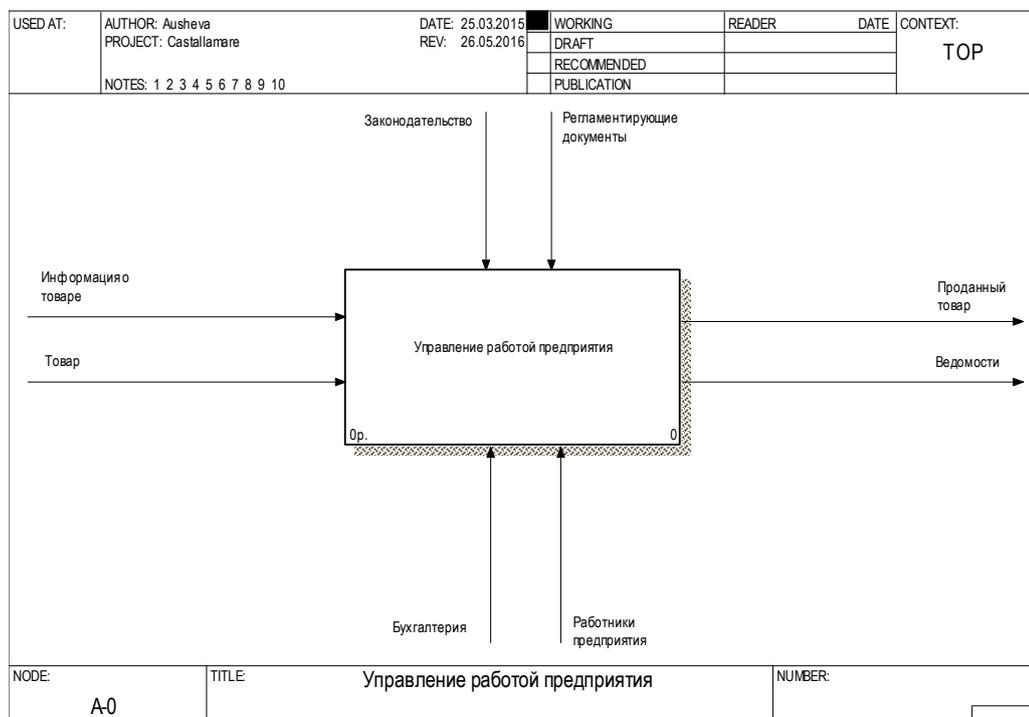


Рисунок 1.2 – Управление работой предприятия

Входной информацией выступает:

- информация о товаре;
- товар.

Все сотрудники компании являются непосредственным механизмом работы предприятия, так как они в той или иной степени оказывают влияние на функционирование организации и делают ее единой конструкцией для достижения поставленных целей. Отдельно следует выделить бухгалтерию. Используя различные программные средства, данное подразделение собирает, регистрируется и обобщается информация о различных действиях и операциях предприятия.

Как управление в представленной диаграмме представляются:

- законодательство;

- регламентирующие документы.

На выходе из функционального блока как итог получаем:

- проданный товар;
- ведомости.

На данной контекстной диаграмме проиллюстрирован общий процесс работы компании. Для более детального рассмотрения необходимо осуществить декомпозицию – метод, позволяющий расчлнить функции одного большого процессана более простые части.

Рисунок 1.3 представляет собой декомпозицию контекстной диаграммы, представленной выше контекстной диаграммы.

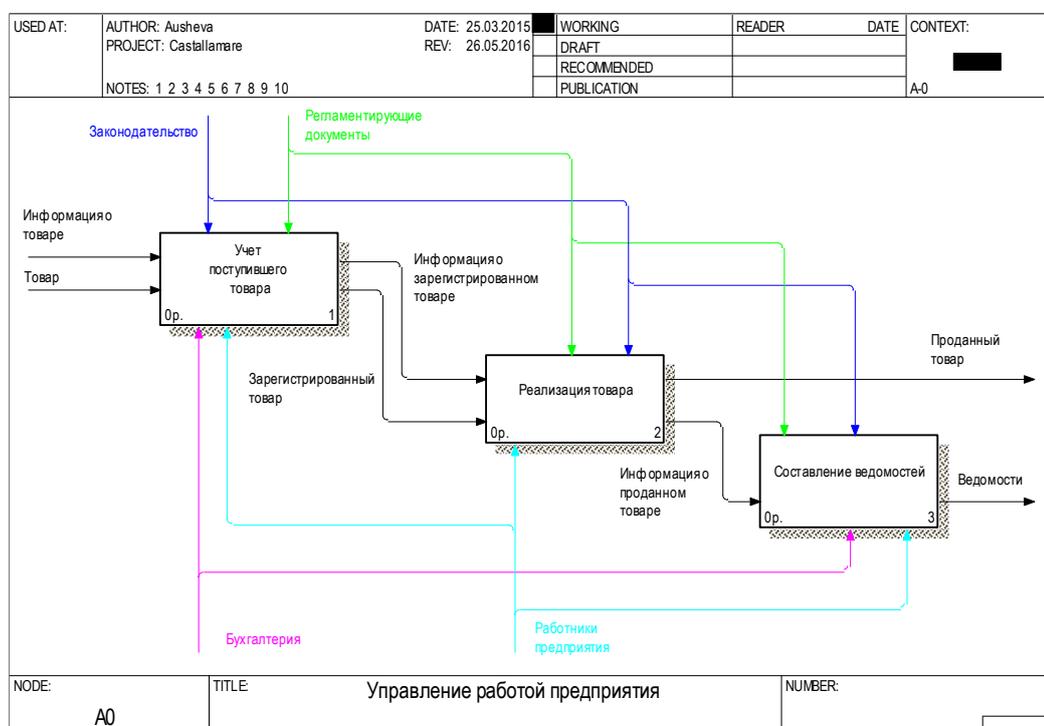


Рисунок 1.3 – Диаграмма декомпозиции

Данная диаграмма имеет уже 3 функциональных блока:

- учет поступившего товара;
- реализация товара;
- составление ведомостей.

Механизм и управление остаются неизменными.

Далее рассмотрим декомпозицию каждого функционального блока с рисунка 1.3.

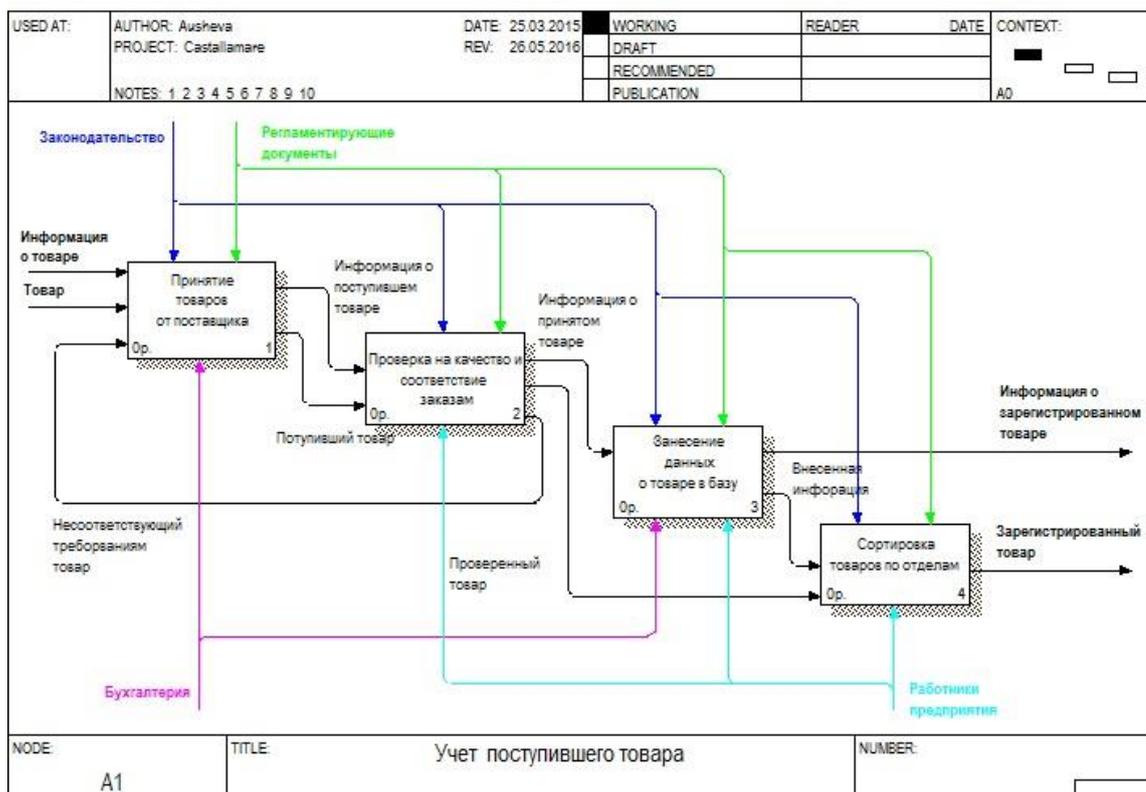


Рисунок 1.4 – Диаграмма декомпозиции «Учет поступившего товара»

Для более наглядного представления блока «Принятие товара от производителя» была выбрана методология моделирования IDEF3. Данный стандарт предназначен для описания бизнес-процессов и содержит объекты – логические операторы, с помощью которых показывают альтернативы и места принятия решений и в бизнес-процессе, а также объекты – стрелки, с помощью которых показывают временную последовательность работ в бизнес-процессе. Помимо наличия нескольких типов связей между работами, в стандарте IDEF3 имеются логические операторы, которые в данном случае называются перекрестками и делятся на несколько типов: [4]

Название перекрестков		Обозначение перекрестков	Смысл перекрестков	
			Схема расхождения	Схема схождения
"Исключающий ИЛИ"			Только одна последующая работа запускается	Только одна предшествующая работа должна быть завершена
"И"	Асинхронный		Все последующие работы запускаются	Все предшествующие работы должны быть завершены
	Синхронный		Все последующие работы запускаются одновременно	Все предшествующие работы должны быть завершены одновременно
"или"	Асинхронный		Одна или несколько последующих работ запускаются	Одна или несколько предшествующих работ должны быть завершены
	Синхронный		Одна или несколько последующих работ запускаются одновременно	Одна или несколько предшествующих работ должны быть завершены одновременно

Рисунок 1.5 – Типы перекрестков

Итогом применения данного метода моделирования стала декомпозиция, приведенная на рисунке 1.6:

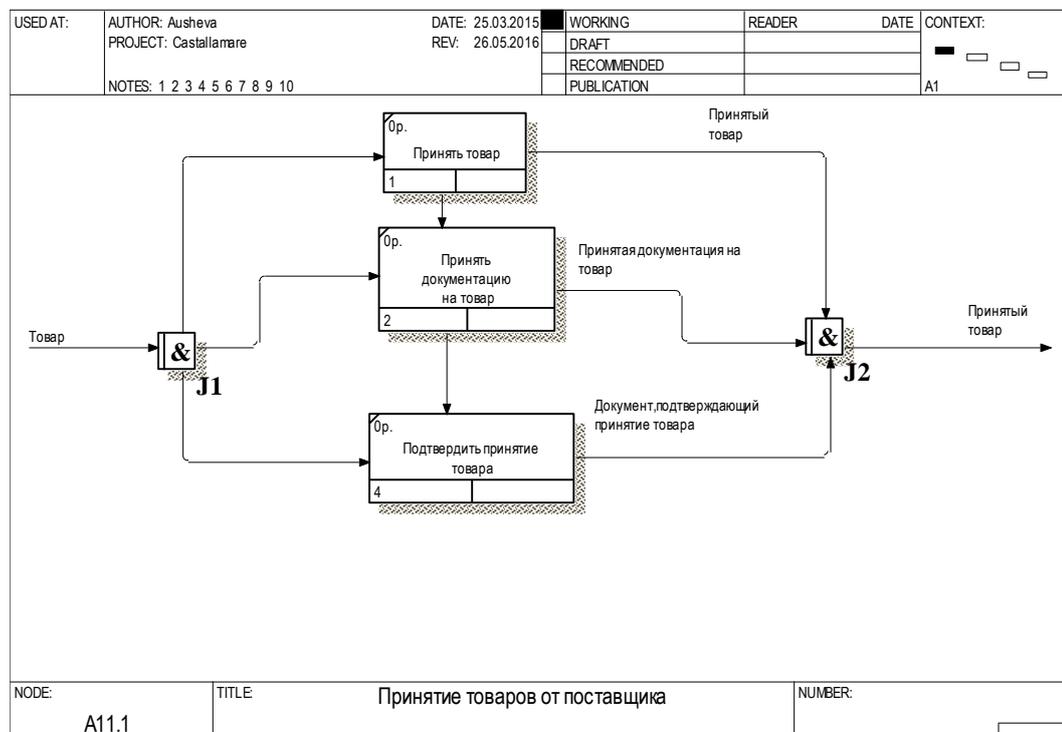


Рисунок 1.6 – Декомпозиция «Принятие товара от производителя»

В диаграммах этого типа мы используем перекресток под названием асинхронный «И», схема расхождения которого обозначает запуск всех последующих работ, а схождение говорит о том, что предшествующие действия должны быть завершены. В данном случае целесообразно отдельно зафиксировать и отразить последовательность прохождения приведенных процессов. Это помогает максимально приблизить графическое представление к реально существующей картине функционирования предприятия.

Используя тот же тип диаграмм и перекрестков, декомпозирован блок «Занесение данных о товаре в базу»:

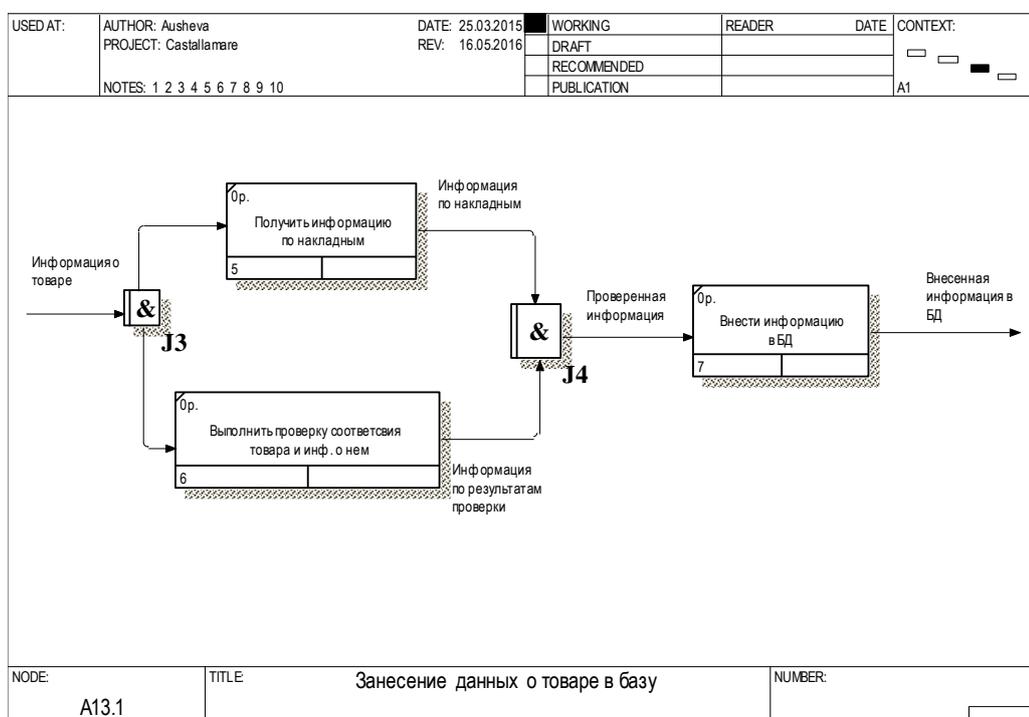


Рисунок 1.7 – Декомпозиция «Занесение данных о товаре в базу»

На данной диаграмме демонстрируется процесс ввода корректной информации о поступившей продукции для дальнейшего ведения ведомостей. Достоверность данных – важный пункт в осуществлении торговли, оформлении отчетов и накладных, а так же ведении базы данных и бухгалтерского учета.

Методом, идентичным предыдущему, выполнена декомпозиция блока «Сортировка товара по отделам»:

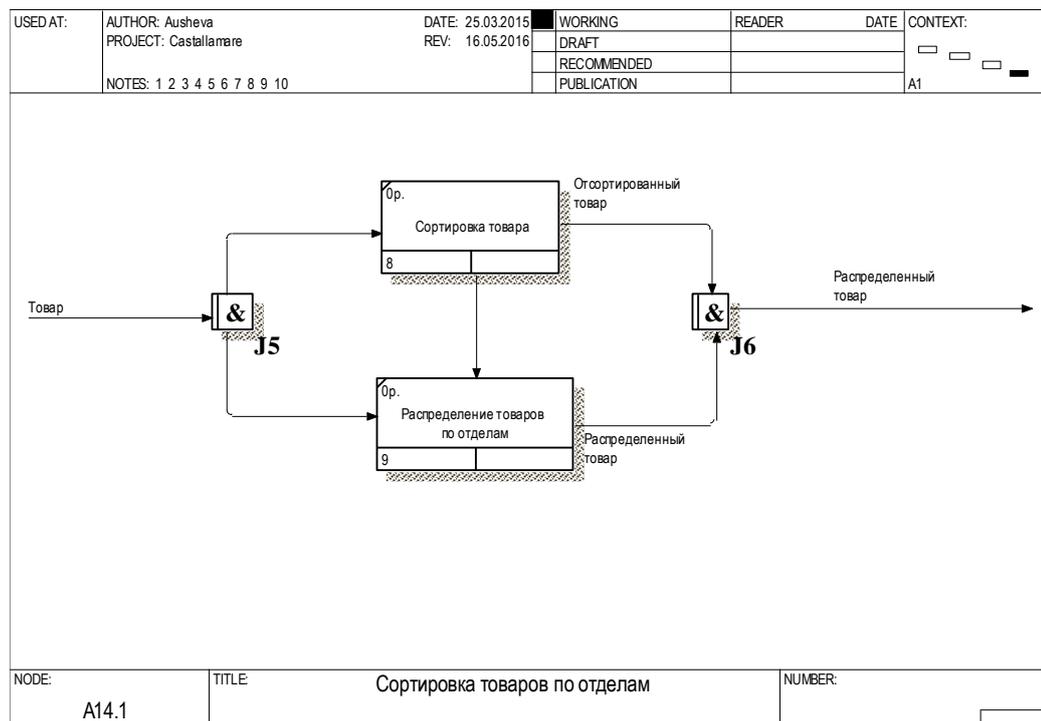


Рисунок 1.8 – Декомпозиция «Сортировка товара по отделам»

Возвращаемся к декомпозициям методологией IDEF0, дабы детально описать второй функциональный блок диаграммы, представленной на рисунке 1.3. На этот раз иллюстрируется процесс осуществления сделки по продаже продукции, сопровождаемый сбором информации для оформления различного рода отчетов. Товар и вся соответствующая информация поступают в отдел сбыта, где выбранная продукция поступает в продажу, данные о реализованном товаре фиксируются и в дальнейшем используются для ведения отчетностей разного рода.

На рисунке 1.9 представлена соответствующая описанию декомпозиция, под названием «Реализация товара»:

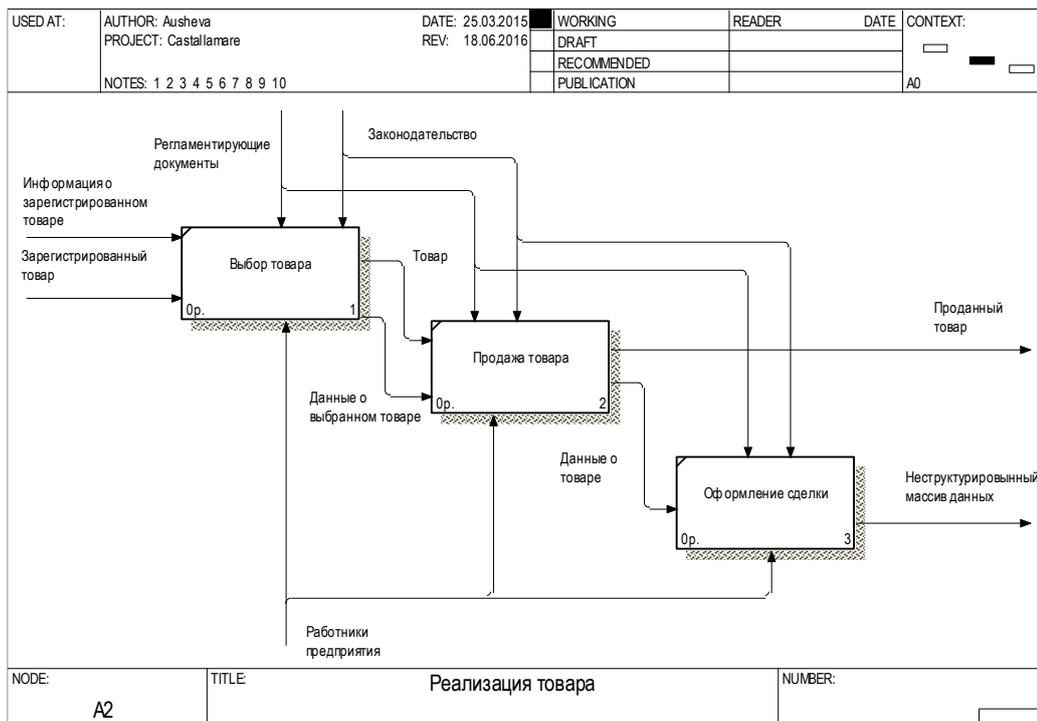


Рисунок 1.9 – Декомпозиция «Реализация товара»

На рисунке 1.10 приведена декомпозиция последнего блока контекстной диаграммы «Составление ведомостей»:

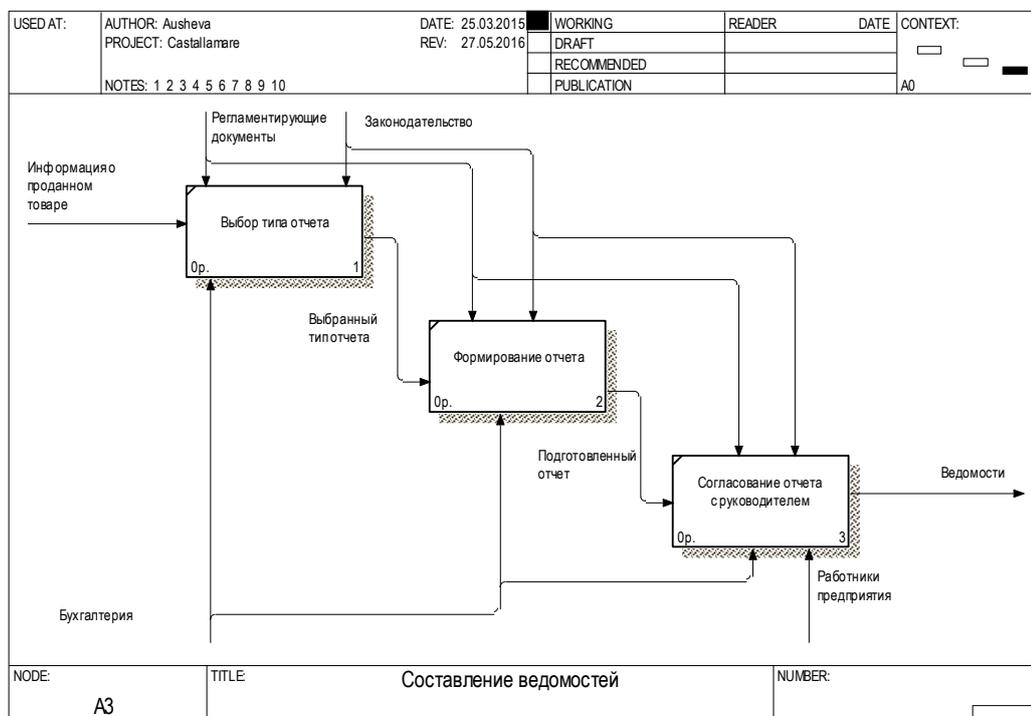


Рисунок 1.10 – Декомпозиция «Составление ведомостей»

1.2 Экономическая сущность задачи

Целью выпускной квалификационной работы является разработка системы хранения корпоративного контента на основе открытого программного обеспечения для ООО «Кастелламаре», работающих в сфере розничных продаж. Компания осуществляет торговлю по заказам, что делает поступление различного типа данных постоянным.

Разрабатываемая система должна обеспечивать частичную автоматизацию работы менеджера по контенту. Также при помощи данного программного продукта будет реализовано составление отчетности об имеющихся данных.

Экономическая сущность задачи состоит в улучшении финансового состояния предприятия посредством возросшего качества, плодотворности и результативности деятельности компании. Этого можно достичь вследствие автоматизации рутинных операций разработанным программным приложением.

В итоге целью разработки является увеличение эффективности работы организации. Результатом внедрения проекта станет колоссальная экономия затрат времени и, как следствие, повышение производительности труда.

1.3 Обоснование необходимости и цели создания системы

Ежедневно в компанию поступает огромное количество различного рода информации: чертежи, заказы, документы, фотографии, инструкции, аудио и видеозаписи совещаний или переговоров. Задача быстрого получения доступа к нужным файлам значительно усложняется по причине отсутствия структурирования крупных массивов данных. На сортировку поступающей информации уходит масса рабочего времени и сил, что еще и не исключает ошибок в процессе систематизации. Для решения подобного рода проблем целесообразно прибегнуть к автоматизации процесса

сортировки. Количество выходной информации не позволит достигнуть нужных результатов без использования специальных программных средств.

Итак, главная цель – организовать хранение и систематизацию информации, представленной в различных форматах.

Внедрение данной системы хранения корпоративного контента в ООО «Касталламаре» ориентировано на повышение результативности и корректности работы информационного отдела, что в итоге увеличит эффективность деятельности всего предприятия.

Цели создания программного приложения следующие:

- сокращение количества операций, особенно производимых вручную;
- контроль качественной сортировки информации;
- снижение затраченного времени на внушительное количество обрабатываемых файлов;
- улучшение условий труда и, как следствие, повышение производительности.

1.4 Постановка задачи

Целью разработки является увеличение общей эффективности работы ООО «Касталламаре» за счет систематизации неструктурированного массива данных.

Результатом является программное приложение, предназначенное для обработки неструктурированного массива корпоративной информации, разделения ее на категории по файловым расширениям с последующим перемещением в определенную папку, соответствующую типу загруженного файла, а так же предоставление различных отчетов о проделанной работе, с возможностью самостоятельного подбора нужных данных и выведения на печать.

Основные функции программной реализации это:

- загрузка информации в промежуточную папку;
- обработка и идентификация расширения файлов;
- загрузка копий файлов в отведенное для данного расширения хранилище и удаление файлов из промежуточной папки;
- внесение завершенной сессии в журнал;
- дальнейшее представление рассортированного контента пользователю;
- составление отчетов.

В процессе реализации проекта необходимо учитывать следующие аспекты:

- обеспечение достоверности обрабатываемой информации;
- выполнение задач в установленные временные рамки;
- обработка данных с минимальными трудовыми и ценовыми затратами;
- возможность обработки информации на ЭВМ;
- предоставление информации об обработанных файлах.

Разработанная система будет решать следующий комплекс задач:

- систематизация данных;
- группировка контента по файловым расширениям;
- упразднение рутинных операций по сортировке;
- экономия временных и трудовых ресурсов;
- обеспечение хранения данных непосредственно на локальном диске компьютера;
- устранение возможности ошибки при сортировке информации или утери файлов в результате уменьшения действия человеческого фактора;
- оперативное получение информации при помощи встроенных отчетов;
- ведение журнала по датам и времени загрузки информации.

2 ОБОСНОВАНИЕ ВЫБОРА ОСНОВНЫХ ПРОЕКТНЫХ РЕШЕНИЙ

2.1 Обоснование проектных решений по техническому обеспечению

Техническое обеспечение – комплекс технических средств, предназначенных для работы информационной системы, а также соответствующая документация на эти средства и технологические процессы.[9]

Немаловажным аспектом увеличения производительности работы предприятия является оптимальное обеспечение информационными средствами, которые имеют достаточный уровень гибкости и адаптируемости к внешним условиям. Информационные технологии берут за основу умение грамотно взаимодействовать с разного рода информацией и ЭВМ.

Техническое обеспечение в первую очередь предполагает под собой средства вычислительной техники. В современном мире это в основном персональные компьютеры.

Комплекс технических средств состоит из:

- компьютеров;
- разнообразных устройств сбора, хранения, обработки, ввода и вывода информации;
- устройств передачи данных и линий связи;
- эксплуатационные материалы и др.

Структура технического обеспечения в виде схемы представлена на рисунке 2.1:

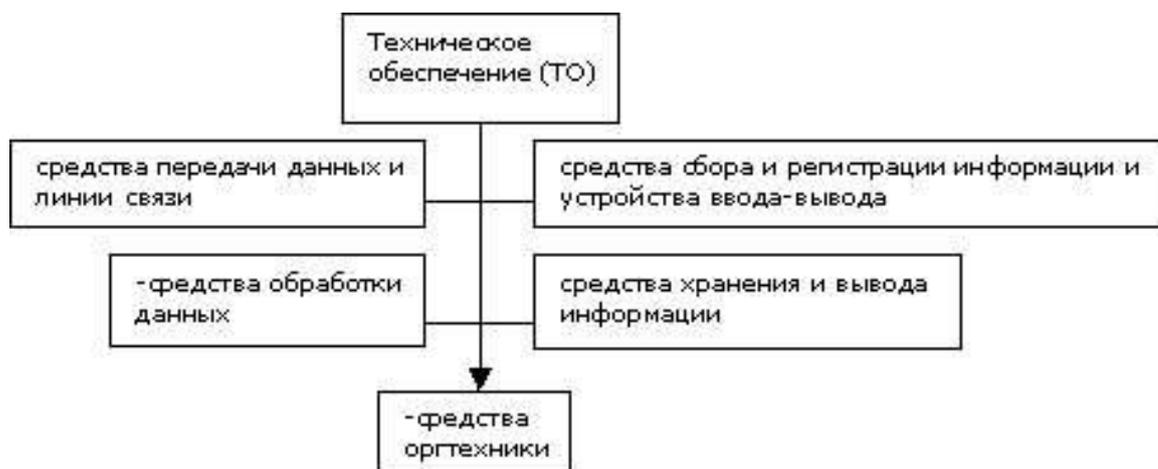


Рисунок 2.1 – Структура технического обеспечения

Основами технического обеспечения выступают:

- автоматизированные рабочие места работников предприятия;
- сеть интернет и локальная вычислительная сеть, которая может состоять из сетевых устройств (маршрутизаторов, коммутаторов и т.д.), и соединяющего их кабеля.

В качестве удаленных автоматизированных рабочих мест будут использоваться персональные компьютеры под управлением ОС Windows 7 со следующими характеристиками:

- Базовые технические конфигурации:

1) процессор AMD Athlon 64 X2 5000+ (ADA5000*/ADO5000*)
Socket AM2 BOX.

Выбор данного процессора обусловлен низкой стоимостью по сравнению с устройствами Intel с аналогичными характеристиками.

2) Материнская плата Asus M2N-SLI Deluxe, S AM2, NVIDIA nForce 570 SLI.

3) Оперативная память DDR2 2048Mb PC2-6400 (800Mhz).

4) Жесткий диск 200,0 Gb HDD Western Digital (WD5000AACS) Caviar GP.

Объем оперативной памяти и жесткого диска стандартны для

функционирования на должном уровне в современном предприятии.

5) Видеокарта 256Mb PCI-E ATI Radeon 3650 DDR3, HDMI, DVI, HIS IceQ Turbo.

6) Источник бесперебойного питания APC.

ИБП является важным дополнением при периодических сбоях в энергоснабжении.

– Устройства вывода:

1) монитор 19" Dell;

2) принтер HP LaserJet 1018.

– Устройства ввода:

1) клавиатура Genius;

2) мышь Genius;

3) привод DVD±RW ASUS DRW-2014L1T, SATA;

4) картридер внутренний.

Помимо элементов, приведенных выше, необходимо наличие сетевой карты подключения к локальной сети организации.

Комплекс технического обеспечения локальной компьютерной сети подразумевает наличие:

– клиентских компьютеров для рабочих мест персонала организации;

– комплекта сетевого оборудования;

– комплекта кабелей;

– устройств ввода и вывода – сканер и принтер;

– устройств согласования и сетевых устройствах – модемов, сетевых адаптеров и др.

Данная конфигурация позволит осуществить разработку программного продукта, а также обеспечит его дальнейшую надежную и качественную работу.

2.2 Обоснование проектных решений по информационному обеспечению

Значение и принципы информационного обеспечения берут свое начало с появления автоматизированных систем управления.

Информационное обеспечение (ИО) – это создание информационных условий функционирования системы, обеспечение необходимой информацией, включение в систему средств поиска, получения, хранения, накопления, передачи, обработки информации. [8]

Иначе говоря, информационное обеспечение – это совокупность нормативных документов базы и реализованных решений по количеству, качеству, размещению и видам существования различного рода данных, которые задействованы в процессе функционирования информационной системы.

Информационное обеспечение включает в себя внутримашинное обеспечение, которое содержит массивы данных (входные, промежуточные, выходные), а так же программы для достижения заданных целей и немашинного обеспечения, что включает систему сортировки и кодирования информации, нормативно–справочной документов.

Структура информационного обеспечения: методические инструктивные материалы (совокупность государственных стандартов), система классификации и кодирования информации, информационная база (внешняя – нормативно–справочные документы, информационные сообщения; внутренняя – информационные массивы). [15]

Схема структуры ИО представлена на рисунке 2.1:



Рисунок 2.2 – Структура информационного обеспечения

Одним из важных требований к информационному обеспечению является достоверность данных информационной базы.

Необходимый уровень достоверность информации обеспечивается высокой степенью контроля имеющихся данных на всех стадиях взаимодействия с ними.

Особенности процесса обработки информации зависят от таких условий, как: функционирование в диалоговом режиме, наличие накопителей нужных сведений, отсутствие бумажных источников данных для обработки данных.

Посредством режима диалога с пользователем исключается предварительно установленная последовательность действий по обработке информации.

В состав технологических операций входят:

- запуск системы;
- загрузка данных;
- проверка информации и возможность внесения поправок;
- информационно-справочное сопровождение;

- формирование массивов структурированного массива данных;
- вывод информации в выбранном удобном виде.

Внесение, обработка и выдача данных должны протекать в режиме диалога.

Основу диалогового режима составляет динамическое взаимодействие ЭВМ и пользователем в процессе получения и прохождения информации через устройства ввода и вывода. В этом режиме работы обеспечивается выдача необходимой информации по запросу, быстрая обработка команд, непосредственное влияние пользователя на ход обработки информации при необходимости.

Информационное обеспечение – совокупность единой системы классификации и кодирования информации, унифицированных систем документации, схем информационных потоков, циркулирующих в организации, а также методология построения баз данных.

К информационному обеспечению предъявляются следующие общие требования:

- информационное обеспечение должно быть достаточным для поддержания всех автоматизируемых функций объекта;
- для кодирования информации должны использоваться принятые классификаторы;
- для кодирования входной и выходной информации, которая используется на высшем уровне управления, должны быть использованы классификаторы этого уровня;
- должна быть обеспечена совместимость с информационным обеспечением систем, взаимодействующих с разрабатываемой системой;
- формы документов должны отвечать требованиям унифицированной системы документации;
- структура документов и экранных форм должна соответствовать характеристиками терминалов на рабочих местах конечных пользователей;
- графики формирования и содержание информационных

сообщений, а также используемые аббревиатуры должны быть общеприняты в этой предметной области;

- в ИС должны быть предусмотрены средства контроля входной и результатной информации, обновления данных в информационных массивах, контроля целостности информационной базы, защиты от несанкционированного доступа.[15]

В состав информационного обеспечения должны входить:

- потоки входной информации;
- потоки выходной информации.

2.3 Обоснование проектных решений по программному обеспечению

Программное обеспечение (ПО) – совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ. [8]

Под программным обеспечением подразумевается комплекс программных средств,обеспечивающих коректную работу системы (системное ПО), а также программ, предназначенных для достижение конкретных пользовательских целей (прикладное ПО). Структура программного обеспечения приведена на рисунке 2.3:

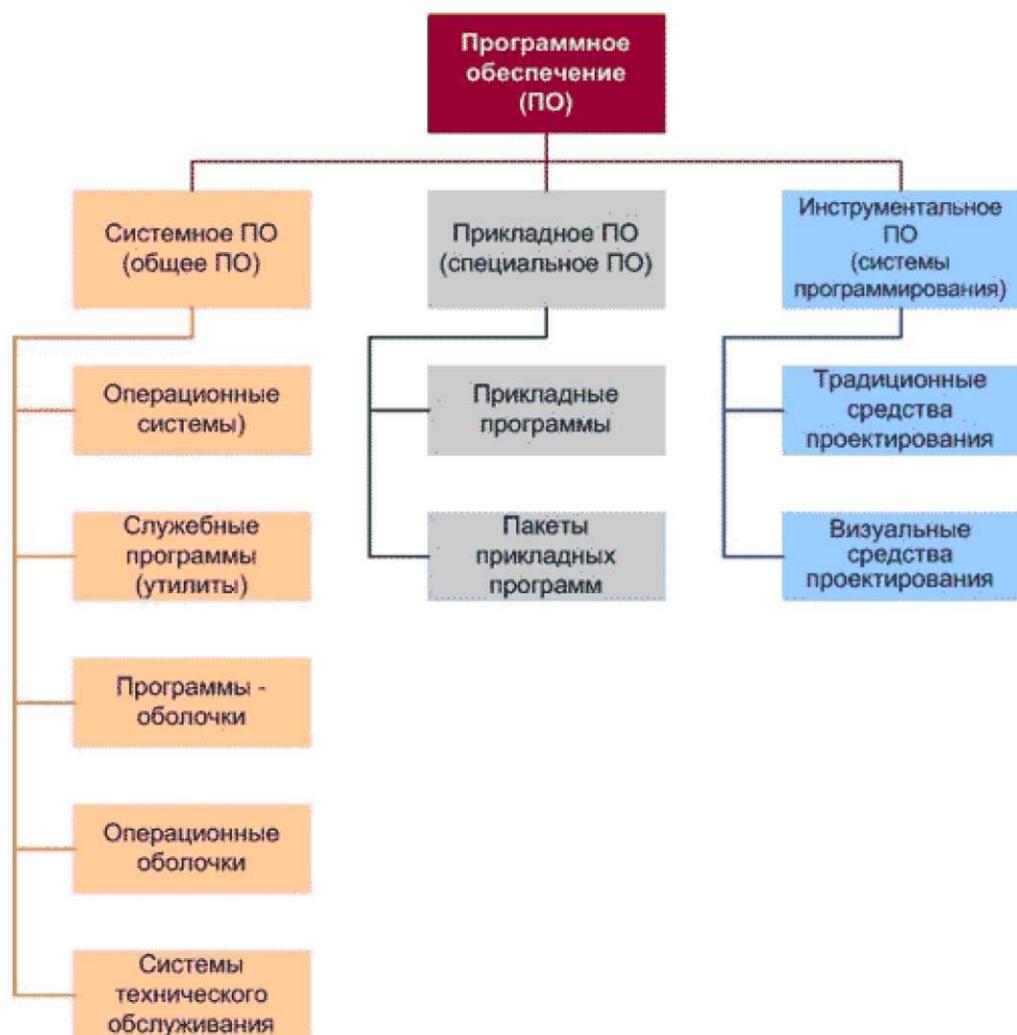


Рисунок 2.3 – Структура программного обеспечения

Для реализации проекта необходимым программным обеспечением выступают операционная система (ОС) и среда программирования.

Операционная система, сокр. ОС (англ. operating system, OS) – комплекс управляющих и обрабатывающих программ, которые, с одной стороны, выступают как интерфейс между устройствами вычислительной системы и прикладными программами, а с другой стороны — предназначены для управления устройствами, управления вычислительными процессами, эффективного распределения вычислительных ресурсов между вычислительными процессами и организации надёжных вычислений. Это определение применимо к большинству современных операционных систем общего назначения. [23]

Операционные системы делятся на:

- однопользовательские и многопользовательские;
- однозадачные и многозадачные.

В настоящее время наиболее популярной операционной системой является Windows XP, не смотря на то, что в 2014 году корпорация Microsoft, (создатель приведенной ОС) и объявила об окончании её поддержки. Современные компании вынуждены переходить на Windows 7 и Windows 8. ООО «Касталламаре» не является исключением – компания ориентирована на инновационное развитие. В связи с этим, разрабатываемое программное обеспечение должно работать под управлением ОС Windows 7,8.

Современные средства разработки ПО имеют обширный диапазон возможностей, используя которые разработчик имеет осуществляет автоматизацию процесса реализации различного рода приложений. Так, на данный момент инструментальные средства позволяют:

- разрабатывать интерфейсы при помощи стандартных компонентов;
- реализовывать управление различным процессам, в зависимости от состояния системы;
- создавать более надежное ПО за счёт возможности обработки исключительных ситуаций, возникающих при сбоях в работе.

Для создания системы было выбрано открытое программное обеспечение, что подразумевает открытый исходный код реализуемого программного приложения. Исходный код создаваемой системы будет доступен для просмотра, изучения, исправления ошибок и внесения изменений, что позволит сотрудникам компании, обладающим нужным набором знаний и средств, доработать программную реализацию проекта, использовать уже имеющиеся строки для создания новых систем и ввести собственные коррективы – посредством заимствования готовых частей.

Современные средства разработки из выбранной категории

характеризуются следующими свойствами:

- полная поддержка объектно-ориентированного стиля программирования;
- использование визуальных компонент для наглядного проектирования интерфейса;
- возможность дальнейшей модернизации и совершенствования на базе уже существующей разработки;
- оперативное вмешательство и устранение ошибок и сбоев в работе системы;
- возможность синхронизации составных частей проекта (предоставляется при разработке больших программных комплексов).

Были выбраны наиболее распространенные решения в этой области – системы разработки программных приложений, которые обладают вышеперечисленными параметрами:

- Delphi;
- Visual C++;
- Borland C++ Builder;
- Visual FoxPro.

В каждом из этих средств имеется в наличии весь набор свойств и спектр современного инструментария, приведенный выше. Различия между ними заключается в области применения рассматриваемых языков.

Visual C++ чаще всего используют при разработке приложений, предназначенных для работы с ОС Windows и в основном выполняющих вычислительные операции. Неоспоримыми недостатками данного средства являются высокие требования к системным и аппаратным ресурсам, а также низкая скорость компиляции программного кода при разработке конечного продукта. Использование Visual C++ для создания приложения требует больший объем дискового пространства, чем при реализации аналогичного ПО другими программными продуктами.

Borland C++ Builder обладает почти идентичным набором недостатков

что и Visual C++, но имеет ряд положительных качеств, дающих преимущество этой системе разработки.

Весомым доводом в пользу выбора Visual FoxPro является то, что она довольствуется наименьшим количеством системных ресурсов, но ее недостатком является визуальное неудобство при создании интерфейса разрабатываемого приложения.

Delphi является добротным визуальным конструктором программ, но не дает достаточного доступа к функциям ОС, что отрицательно сказывается на частоте выбора в пользу данного средства программирования.

2.4 Обоснование выбора программных средств

Основной базой для становления на путь оптимизации и автоматизации деятельности организации является реинжиниринг бизнес-процессов, что подразумевает фундаментальное переосмысление и радикальное перепроектирование бизнес-процессов для достижения максимального эффекта производственно-хозяйственной и финансово-экономической деятельности, оформленное соответствующими организационно-распорядительными и нормативными документами. Реинжиниринг использует специфические средства представления и обработки проблемной информации. [27] Для достижения качественной оптимизации структуры бизнес-процессов были выбраны средства системного анализа AllFusion Process Modeler r7.

AllFusion Process Modeler r7 – средство моделирования, используемое для анализа, документирования и реорганизации бизнес-процессов, сложных по своей структуре. Бизнес-процессы, созданные с помощью средств AllFusion Process Modeler r7, документируют разные стороны деятельности: действия, необходимые для предприятия, способы их осуществления, необходимые ресурсы. [19]

Уникальность разрабатываемого программного приложения

заключается в том, что система не нуждается в использовании промежуточной базы данных. Она связывается и взаимодействует напрямую с Проводником, что позволяет хранить отсортированный массив данных на локальном диске компьютера. Это положительно повлияет на производительность и результативность труда после внедрения разработки, а также будет существенно экономить время.

Проводник Windows – это приложение, реализующее графический интерфейс доступа пользователя к файлам в операционной системе Microsoft Windows. В Windows одна из программ является оболочкой (англ. Windows shell). Оболочка запускается вместе с Windows, в ранних версиях Windows выход из оболочки приводил к выходу из операционной системы. Современные версии Windows, если оболочка завершит работу крахом, могут загружать новую копию. [23]

Иными словами Проводник – это специализированная программа, созданная для взаимодействия с файловой системой. Она позволяет пользователю осуществить доступ к дискам и папкам, находящимся на компьютере.

На проводник ОС возложены функции по созданию, копированию, перенесению, открытию, поиску, удалению и переименованию файлов и папок.

На данный момент Проводник является основой графической оболочки пользователя Windows.

Далее необходимо определиться с выбором системы разработки программного приложения. При создании системы главными критериями выбора являлись:

- скорость создания программного продукта;
- возможность оперативного внесения изменений в реализуемое приложение;
- осуществимость привязки среды к оболочке Проводник ОС Windows;

- наличие средств создания, редактирования и выведения отчетов при использовании выбранной системы разработки.

Как дополнение к перечисленному, можно указать, что время разработки зависит от: поддержки выбранным инструментарием ОС, аппаратной поддержки, необходимой для их оптимального функционирования; наличия предварительного опыта у разработчиков в использовании соответствующих программных средств. Обеспечить минимальное время разработки можно только при выполнении этих условий.

Принимая в учет выделенные критерии, назначим следующие характеристики средств разработки программных продуктов:

- наличие навыков использования программного обеспечения;
- условия по системным и аппаратным ресурсам;
- поддержка операционной системы;
- визуализированность создания интерфейса;
- спектр возможностей работы с оболочкой ОС;
- доступность;
- скорость функционирования реализованного приложения;
- обработка исключительных ситуаций;
- время, затраченное на создание готовой системы;
- удобство эксплуатации.

Для выбора наилучшего варианта из нескольких предложенных будет задействован метод вариантных обоснований. Он нацелен на подведение окончательных итогов о наиболее подходящем программном средстве из представленных. В данном случае метод будет строиться из следующих аспектов:

- утверждение критериев для сравнения и определение степени их весомости;
- каждое средство разработки оценивается по выбранному перечню характеризующих критериев. Полученное значение – оценка в числовом выражении;

- вычисление общей суммы баллов для каждого из программных средств;
- самым подходящим считается средство разработки, которое получило наибольшую итоговую оценку.

В результате выполненной сравнительной характеристики нескольких средств разработки, приведенного в Приложении А, было установлено, что в качестве инструментального средства реализации программного приложения будет использован Borland C++ Builder, как наиболее оптимальный вариант с точки зрения разработчика.

Borland C++ Builder это мощные и надежные среды разработки высокоэффективных web-служб и приложений для электронного бизнеса, позволяющие эффективно работать с корпоративными базами данных и имеющие широкий выбор управляемых данными элементов интерфейса, что даёт возможность быстро создавать прототипы приложений.

Данное программное обеспечение было выпущено компанией Borland как средство для быстрой разработки приложений, позволяющее создавать программные реализации на языке C++, имея возможность задействовать среду и библиотеку компонентов Delphi. Borland C++ Builder обладает уникальной поддержкой визуальной разработки в сочетании с высокой степенью соответствия стандарту ANSI C++. Это позволило C++ Builder занять лидирующие позиции в области средств разработки на C++.

Значительным плюсом выбранного программного средства является технология быстрой разработки приложений. Данного преимущества удалось добиться за счёт включения широкого спектра средств, которые повышают производительность труда разработчиков и сокращают продолжительность цикла создания программных продуктов.

3 ПРОЕКТНАЯ ЧАСТЬ

3.1 Разработка информационных моделей

Информационная модель – модель объекта, представленная в виде информации, описывающей существенные для данного рассмотрения параметры и переменные величины объекта, связи между ними, входы и выходы объекта и позволяющая путём подачи на модель информации об изменениях входных величин моделировать возможные состояния объекта. Информационные модели нельзя потрогать или увидеть, они не имеют материального воплощения, потому что строятся только на информации. Информационная модель — совокупность информации, характеризующая существенные свойства и состояния объекта, процесса, явления, а также взаимосвязь с внешним миром. [28]

Разработка функциональных моделей предназначена для описания процесса работы отдельно взятого процесса. На этих моделях наглядно представлены основные моменты работы системы, а так же процесс преобразования входных данных в выходные.

Задача выпускной квалификационной работы заключается в разработке системы хранения неструктурированного массива данных.

Наглядное представление нынешнего процесса сортировки файлов реализуется при помощи выбранного ранее инструмента визуального моделирования AllFusion Process Modeler. Средствами именно этой среды была построена диаграмма «КАК ЕСТЬ».

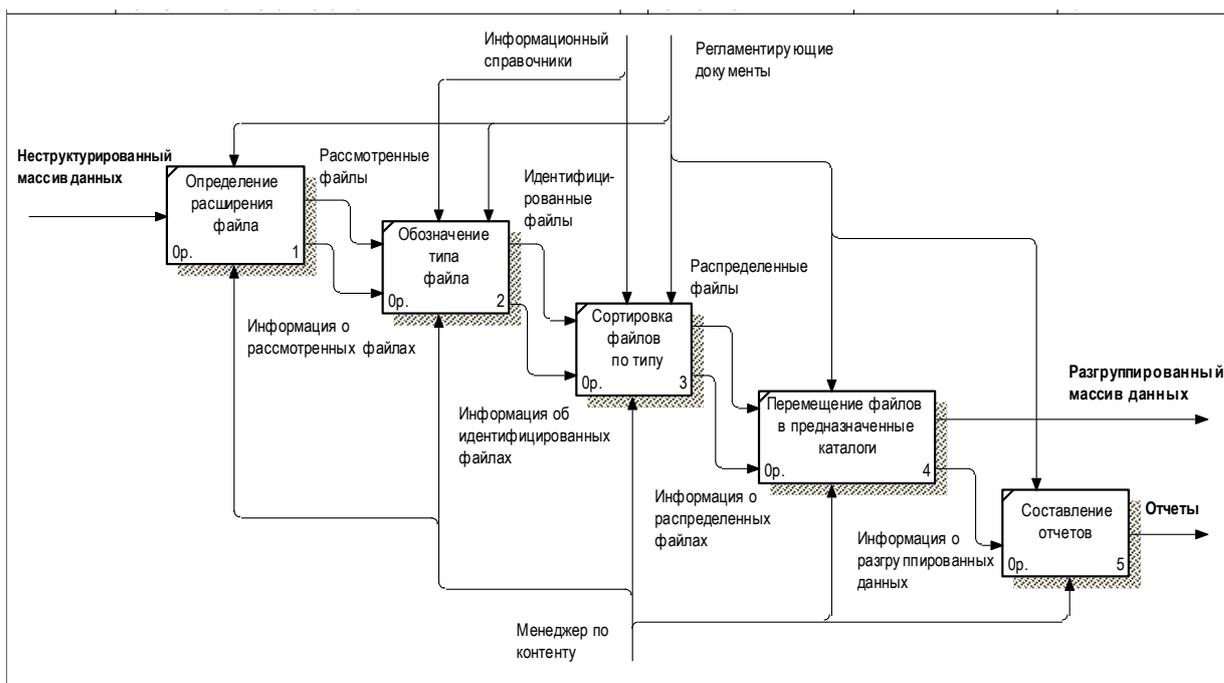


Рисунок 3.1 – Диаграмма процесса сортировки данных «КАК ЕСТЬ»

Изучив и проанализировав данную область, была составлена схема процесса с теми же исходными данными, но учитывающая внедрение разработанного программного приложения. По сравнению с моделью «КАК ЕСТЬ» в качестве механизма управления для функционирования добавлено Приложение «Касталламаре». Разрабатываемая система предназначена для автоматизации процесса сортировки, а также хранения файлов на локальном диске, а так же фиксирование результатов распределения в электронном виде.

В результате внедрения программного приложения существенно упростится работа сотрудников организации за счет того, что она возьмет на себя функции рутинной механической работы и позволит избежать ошибок при систематизации.

Процесс сортировки данных «КАК ДОЛЖНО БЫТЬ» в виде модели в нотации IDEF0 представлен на рисунке 3.2.

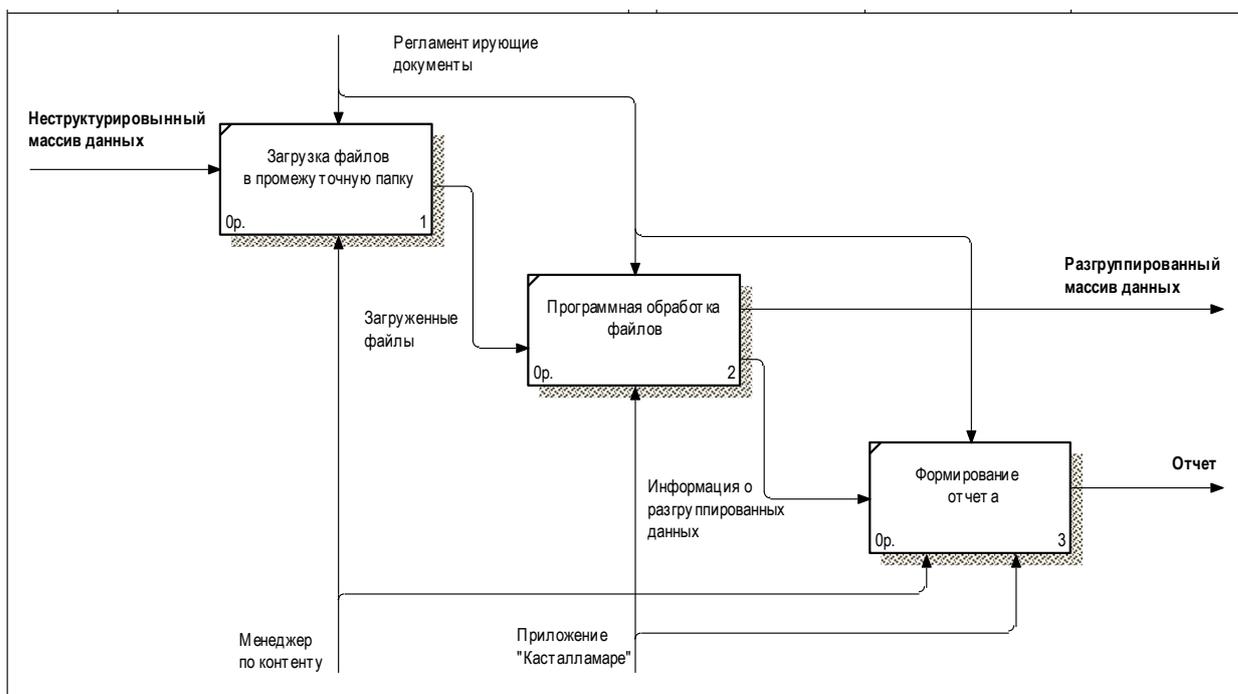


Рисунок 3.2 – Диаграмма процесса сортировки данных «КАК ДОЛЖНО БЫТЬ»

В итоге на данной схеме можно наглядно пронаблюдать существенное упрощение процесса сортировки при совершенно идентичных входных условиях.

3.2 Информационное обеспечение задачи

Информационное обеспечение в данном случае – оснащение программного приложения актуальными и самыми распространенными файловыми расширениями, для качественной реализации функций внедряемой системы.

Расширение имени файла (англ. file name extension, часто говорят просто расширение файла или расширение) – последовательность символов, добавляемых к имени файла и предназначенных для идентификации типа (формата) файла. Это один из распространённых способов, с помощью которых пользователь или программное обеспечение компьютера может определить тип данных, хранящихся в файле.

Расширение обычно отделяется от основной части имени файла точкой. В операционных системах CP/M и MS-DOS длина расширения была ограничена тремя символами, в современных операционных системах это ограничение отсутствует. Иногда могут использоваться несколько расширений, следующих друг за другом, например, «.tar.gz».

В файловой системе FAT16 имя файла и расширение являлись отдельными сущностями, а точка, разделявшая их, реально не являлась частью полного имени файла и служила лишь для визуального отделения имени файла от расширения. В файловых системах FAT32 и NTFS точка стала обычным разрешённым символом в имени файла, поэтому ограничения на количество точек в имени файла в этих системах и их местоположения были сняты (за некоторыми исключениями, например, все конечные точки в именах файлов просто отбрасываются).

Некоторые операционные системы или менеджеры файлов могут устанавливать соответствия между расширениями файлов и приложениями. Когда пользователь открывает файл с зарегистрированным расширением, автоматически запускается соответствующая этому расширению программа. Некоторые расширения показывают, что файл сам является программой. [33]

Реализованная система будет распознавать:

– архивы с расширениями:

1) 7z;

2) rar;

3) zip.

– Аудиофайлы с расширениями:

1) mp3;

2) wav;

3) wma;

4) dat;

5) aif.

- Видеофайлы с расширениями:
 - 1) avi;
 - 2) wmv;
 - 3) mkv;
 - 4) mov;
 - 5) mp4.
- Документы с расширениями:
 - 1) doc;
 - 2) docx;
 - 3) docm;
 - 4) dot;
 - 5) dotx;
 - 6) txt;
 - 7) odt;
 - 8) fodt;
 - 9) odm;
 - 10) xls;
 - 11)xlsx;
 - 12) xml;
 - 13) pdf;
 - 14) djvu.
- Изображения с расширениями:
 - 1) jpg;
 - 2) jpeg;
 - 3) png;
 - 4) gif;
 - 5) psd;
 - 6) raw.
- Презентации с расширениями:
 - 1) ppt;

2) pptx;

3) pps;

4) ppsx.

Также необходимо наличие папки «Прочее», для всех файлов, тип которых не удалось определить.

На данный момент программа идентифицирует самые часто встречающиеся расширения. Спектр распознаваемых типов файлов может быть существенно расширен любым сотрудником компании, обладающим необходимыми знаниями в области разработки подобного рода систем.

3.3 Программное обеспечение задачи

3.3.1 Структура программного приложения

В данном случае под структурой программного приложения подразумеваются файлы и папки, задействованные в процессе систематизации.

Для выполнения поставленных задач программным приложением необходимо следовать определенной последовательности действий, с которыми вполне может справиться рядовой пользователь ПК. Все компоненты, необходимые для запуска и функционирования системы находятся в одной папке, которую принципиально важно помещать на Локальный диск (C:) компьютера. Там же будет храниться отсортированный массив данных.

Структура программного приложения представлена на рисунке 3.3. Здесь собран весь функционал для использования программного приложения, а также дальнейшего взаимодействия со всеми типами данных, полученных в результате работы системы: начиная корпоративной информацией и заканчивая электронным журналом операций.

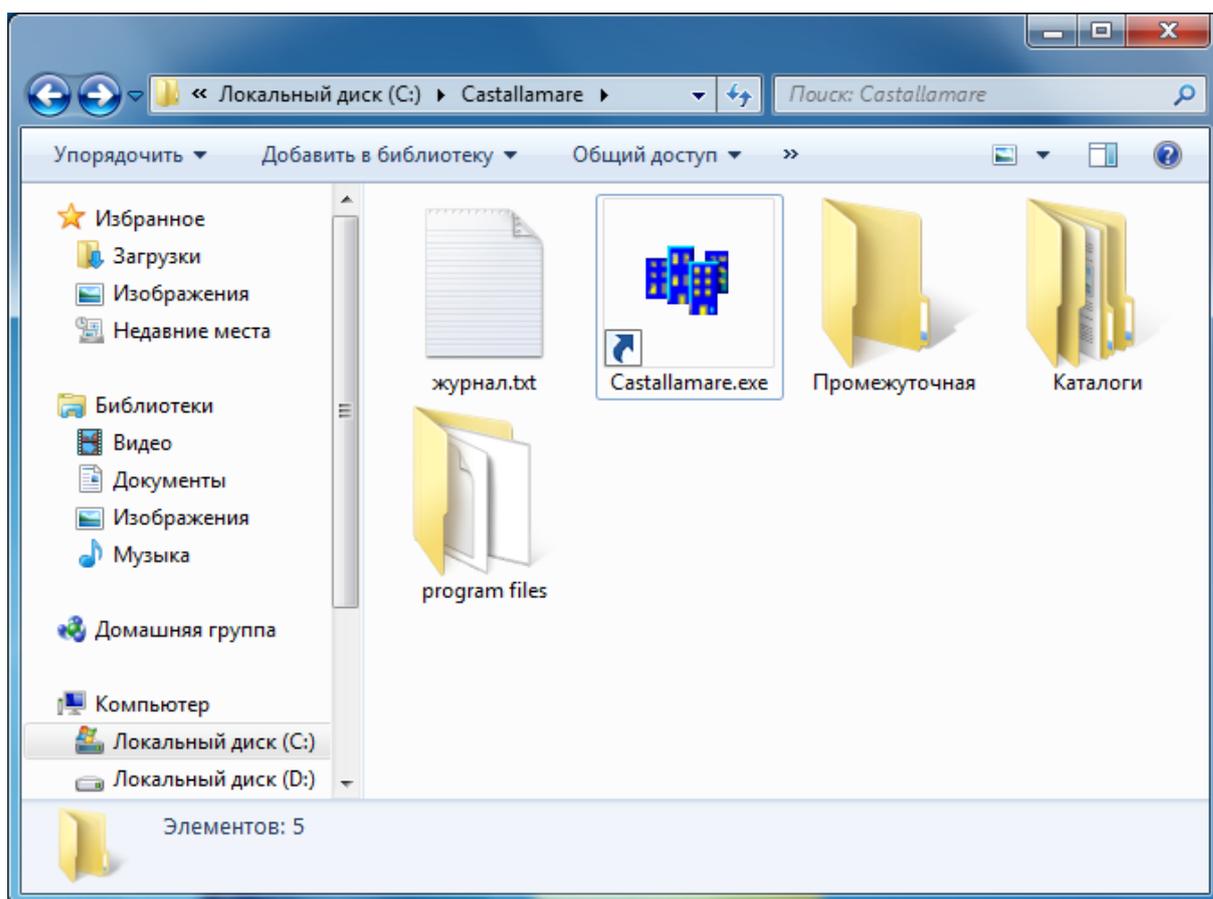


Рисунок 3.3 – Содержимое рабочей папки программного приложения

Папка под названием «Промежуточная» предназначена для загрузки данных, требующих сортировки. Ярлык `Castallamare.exe` запускает процесс идентификации и сортировки, который обрабатывает весь контент, находящийся в папке «Промежуточная».

Копии файлов помещаются в соответствующий расширению и типу раздел папки «Каталоги», в которой и будет предоставляться рассортированный контент пользователю. Исходные сведения удаляются.

Каждая завершённая сессия по сортировке фиксируется в файле «Журнал».

В папке «program files» находится перечень объектов, реализующих работу системы. Содержимое представлено на рисунке 3.4.

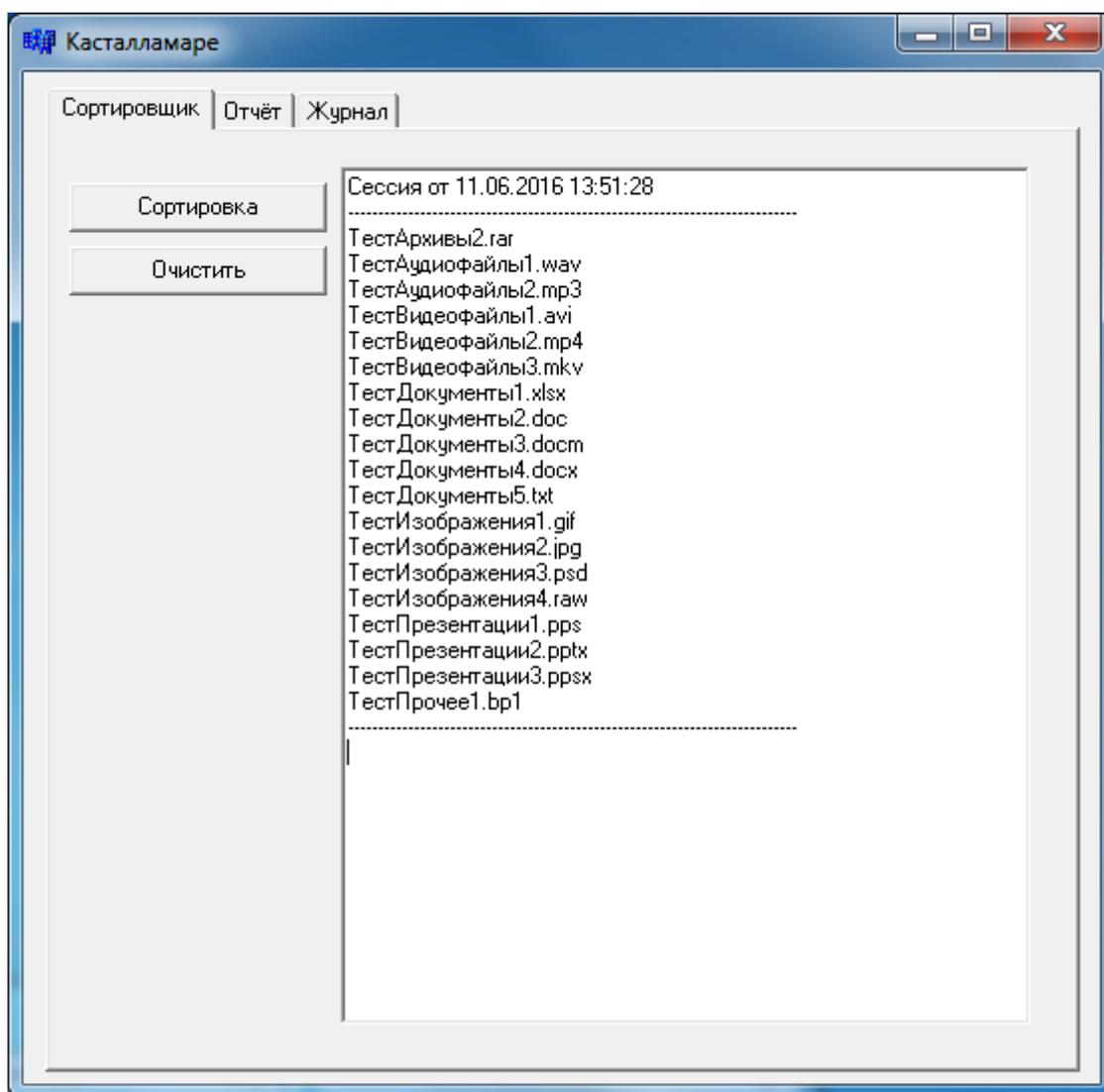


Рисунок 3.5 – Вкладка «Сортировщик»

В данной части программного приложения реализуется распределение неструктурированного массива данных по предназначенным для их расширения папкам. Это происходит по команде кнопки «Сортировка». В окне выводится названия всех файлов, проходящих распределение, а так же фиксируется дата и время. Кнопка «Очистить» стирает все данные, находящиеся в окне, но только на этой вкладке. Вся представленная информация автоматически заносится в вкладку «Журнал» и в файл txt с идентичным названием, который находится в рабочей папке программного приложения.

Далее на рисунке 3.6 приведен вид вкладки под названием «Отчет»:

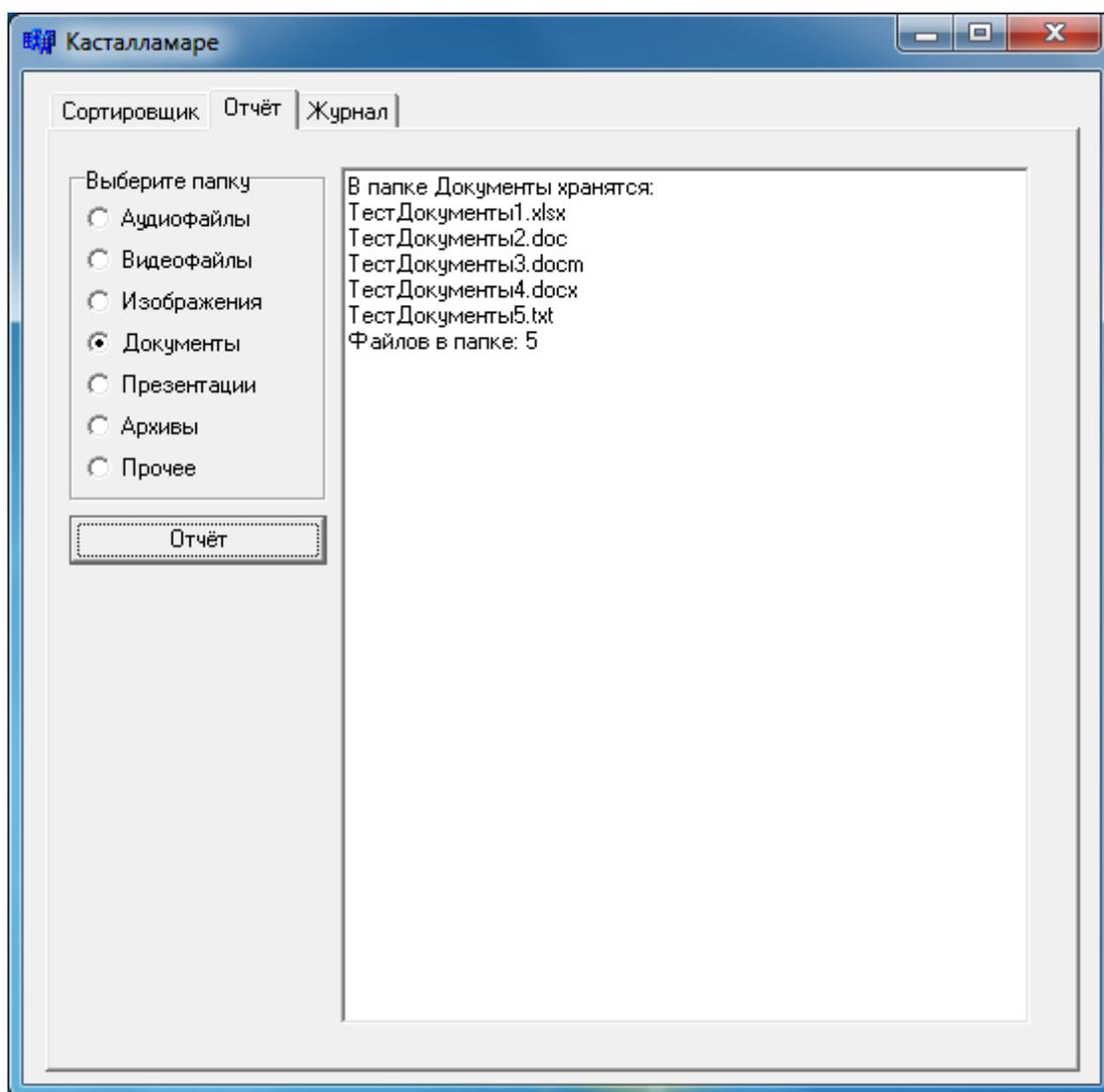


Рисунок 3.6 – Вкладка «Отчет»

Данный раздел программного приложения предназначен для внутрипрограммного доступа к информации о содержащихся в папках файлах. Приведен перечень имеющихся вариантов составления отчета и, выбрав один, щелкнув по кнопке «Отчет» мы получим соответствующую выбранному критерию информацию. Также реализован подсчет количества всех находящихся файлов. Важно отметить, что с выводимой в окне информацией можно взаимодействовать. Это позволяет быстро сформировать отчетную ведомость по количеству рассортированных файлов в каждой папке.

На рисунке 3.7 продемонстрирована последняя вкладка:

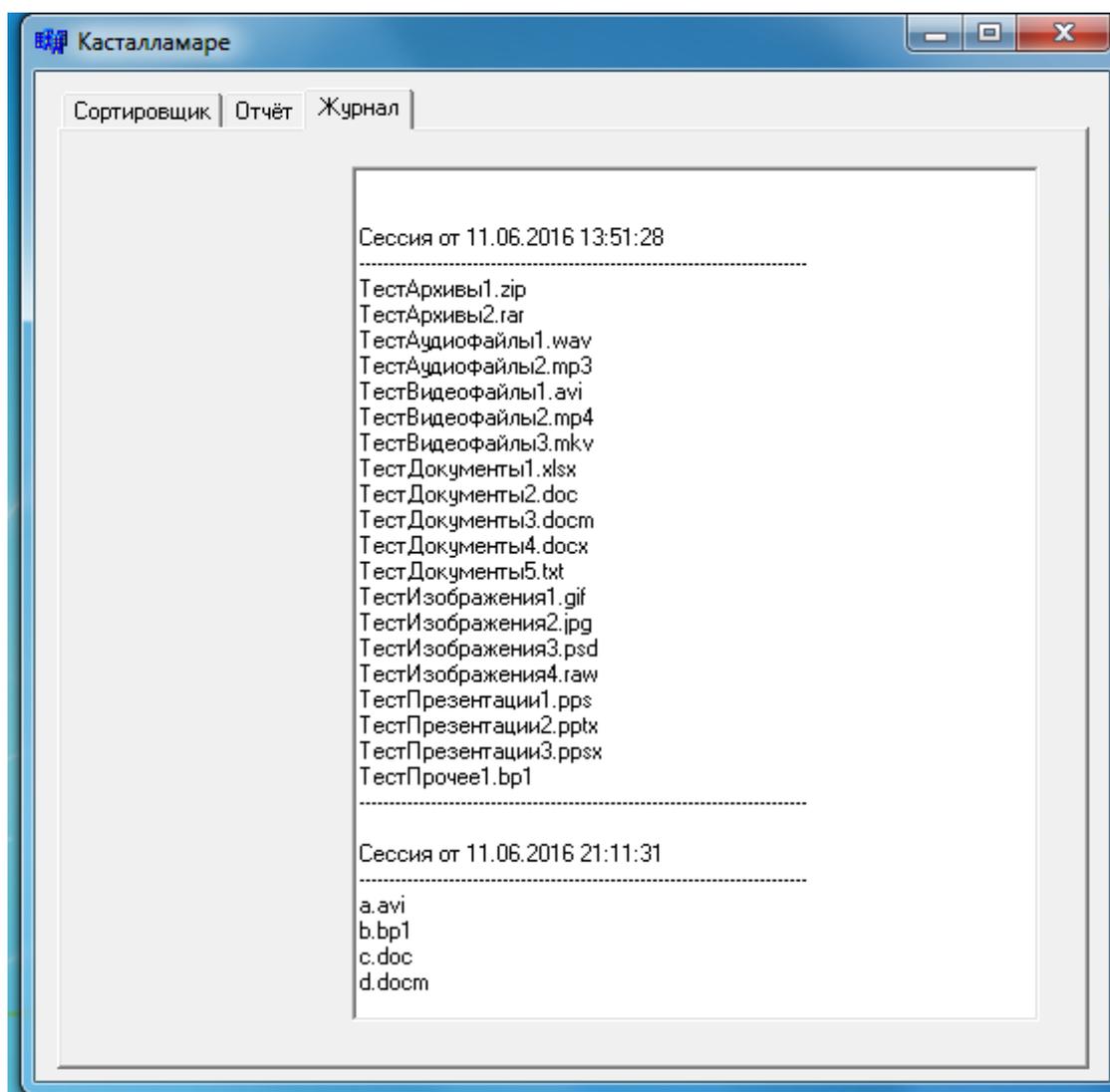


Рисунок 3.7 – Вкладка «Журнал»

Представленный сегмент системы предназначен для ведения учета производимых операций. Здесь хранится информация о дате и времени каждой сортировки, а так же список файлов, предоставленных программному приложению для сортировки. Данная вкладка связана с файлом форматаtxt, который хранится в папке программы. Туда заносятся все данные, представленные в журнале для дальнейшего использования для составления отчетных ведомостей.

При завершении каждой программно реализуемой операции, приложение дает знать об этом при помощи выведения сообщения об окончании выполняемой процедуры. Визуальное отображение данного свойства представлено на рисунке 3.8:

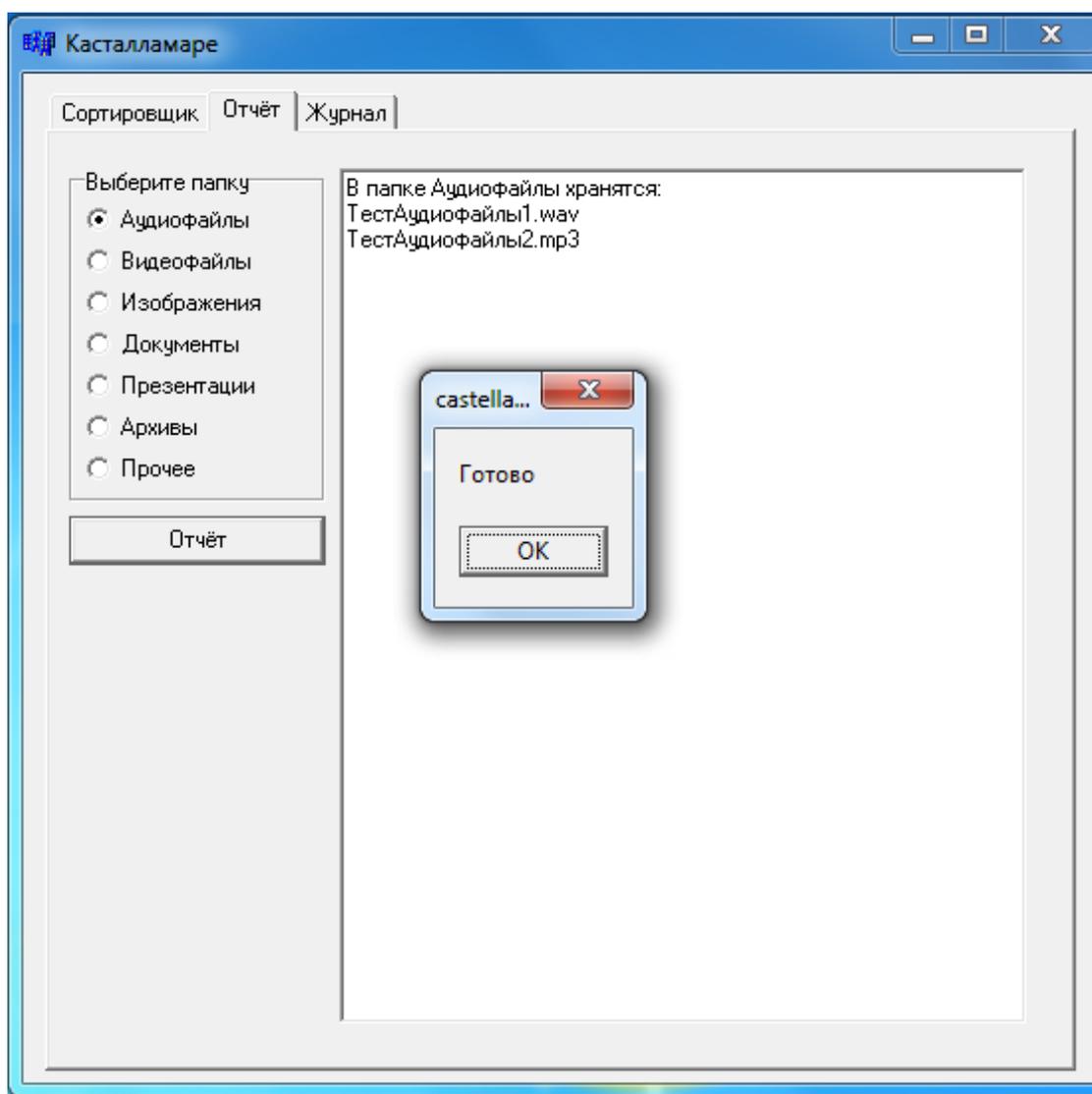


Рисунок 3.8 – Окно, сообщающее о завершении процесса

3.4 Описание контрольного примера реализации проекта

В качестве контрольного примера приведем подробное и наглядное описание процесса работы системы. Будут задействованы программное приложение и файлы различных типов и расширений.

Начальным этапом работ по систематизации является перенесение неструктурированного массива данных, нуждающегося в обработке, в промежуточную папку. В приведенном конкретном случае использовались 20 файлов общим весом 193 Мб. Эта операция продемонстрирована на рисунке 3.9:

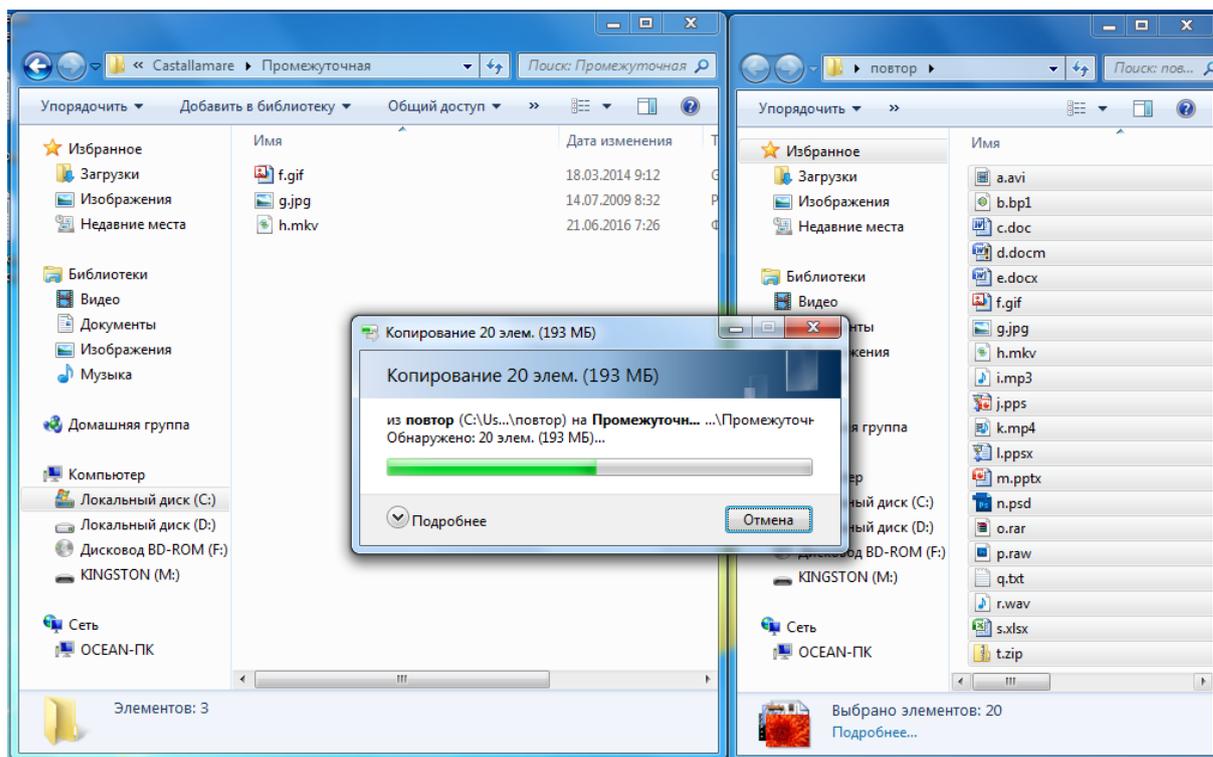


Рисунок 3.9 – Загрузка информации в промежуточную папку

Дальнейшим действием является запуск приложения `Castallamare.exe`. При нажатии кнопки под названием `Сортировка` запускается процесс обработки и идентификации расширений файлов, предоставленных пользователем, которые находятся в папке «Промежуточная».

Название каждого обработанного файла выводится в окно вместе с расширением. Опытным путем было выяснено, что программное приложение способно обработать до 3 Гб неструктурированных данных за минуту.

Дождемся подтверждения окончания операции по сортировке и нажимаем «ОК». На рисунке 3.10 показано визуальное отображение результата описанных выше действий:

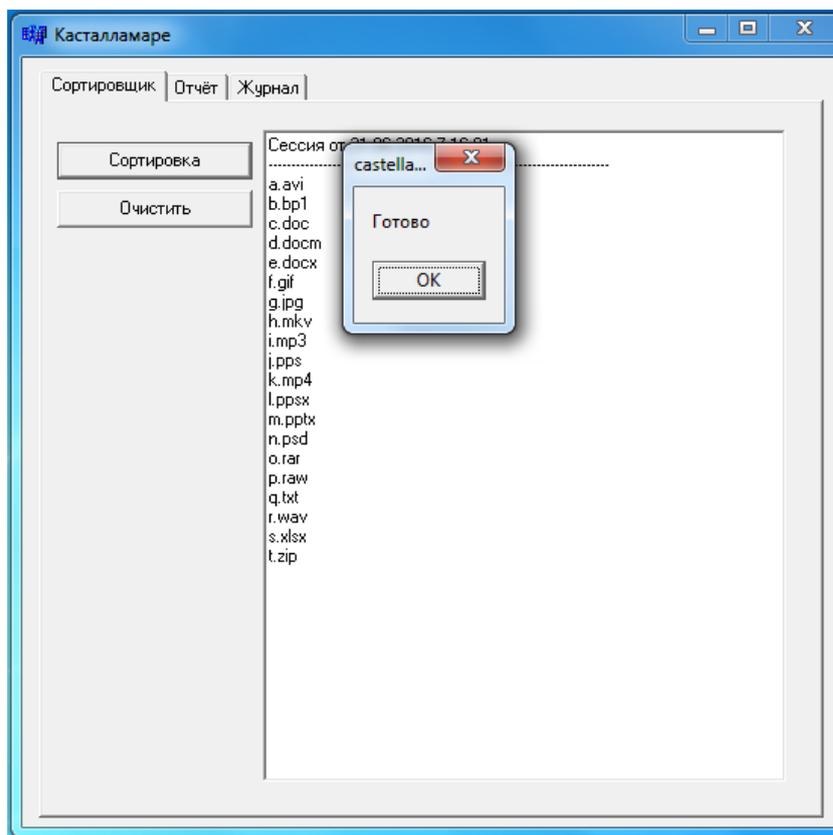


Рисунок 3.10 – Завершение сортировки предоставленных файлов

Удаляем содержимое рабочего окна с помощью кнопки «Очистить»:

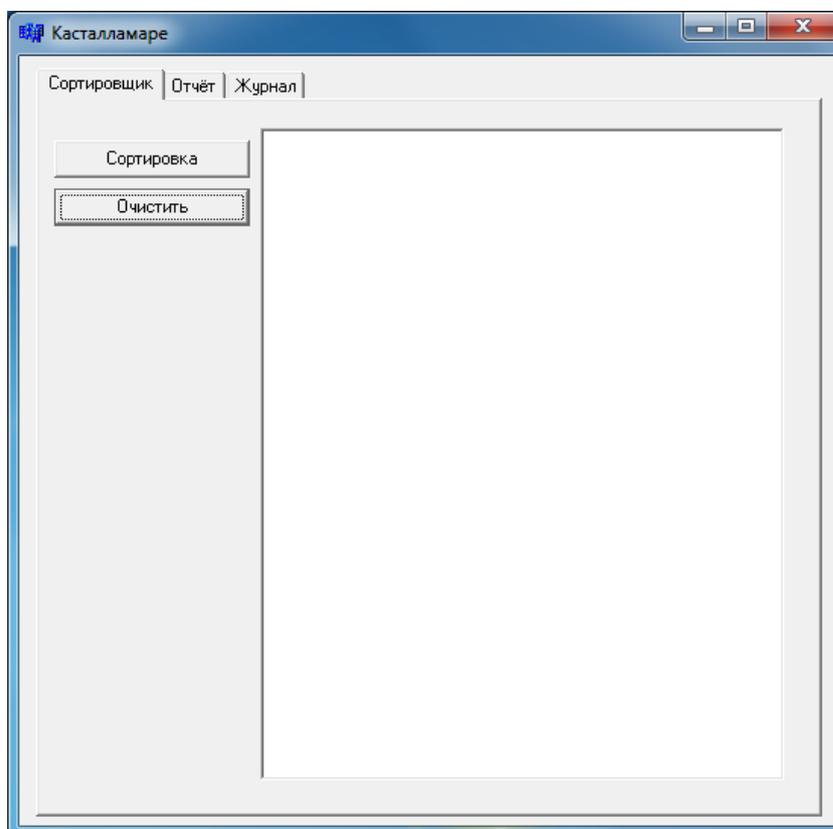


Рисунок 3.11 – Результат работы кнопки «Очистить»

Для подтверждения корректности работы системы проверяем содержимое папок в Каталоге. На рисунке 3.12 приведен конечный итог функционирования программного приложения:

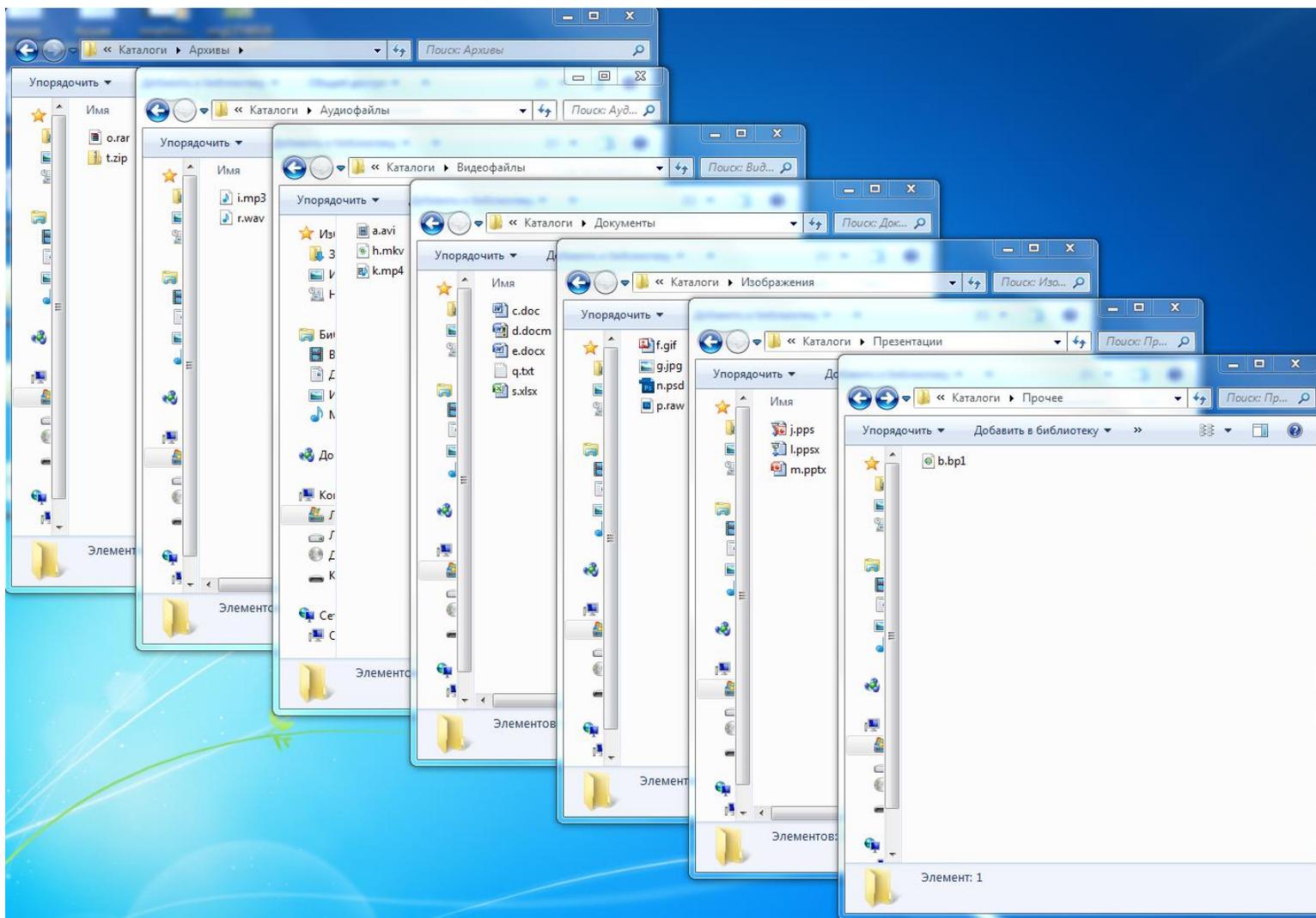


Рисунок 3.12 – Итоги сортировки по папкам

Проверим отображение нескольких последних процессов сортировки файлов как в программной реализации (вкладка «Журнал»), так и в файле Журнал.txt.

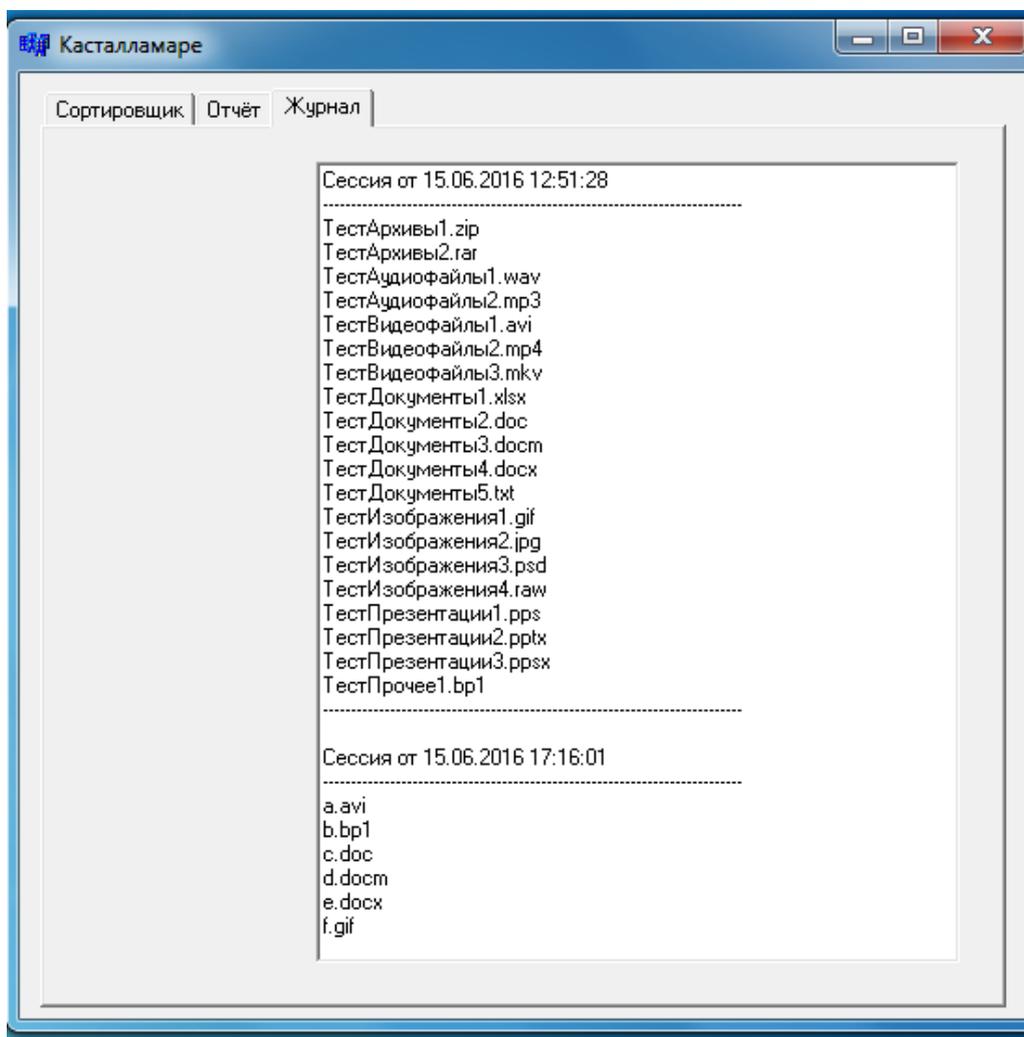


Рисунок 3.13 – Содержимое вкладки «Журнал»

Все сортировки должны фиксироваться не только в приведенном выше средстве, хоть и имеется возможность выполнять некоторые операции, свойственные текстовым документам. Это окно напрямую связано с файлом, находящимся в рабочей папке.

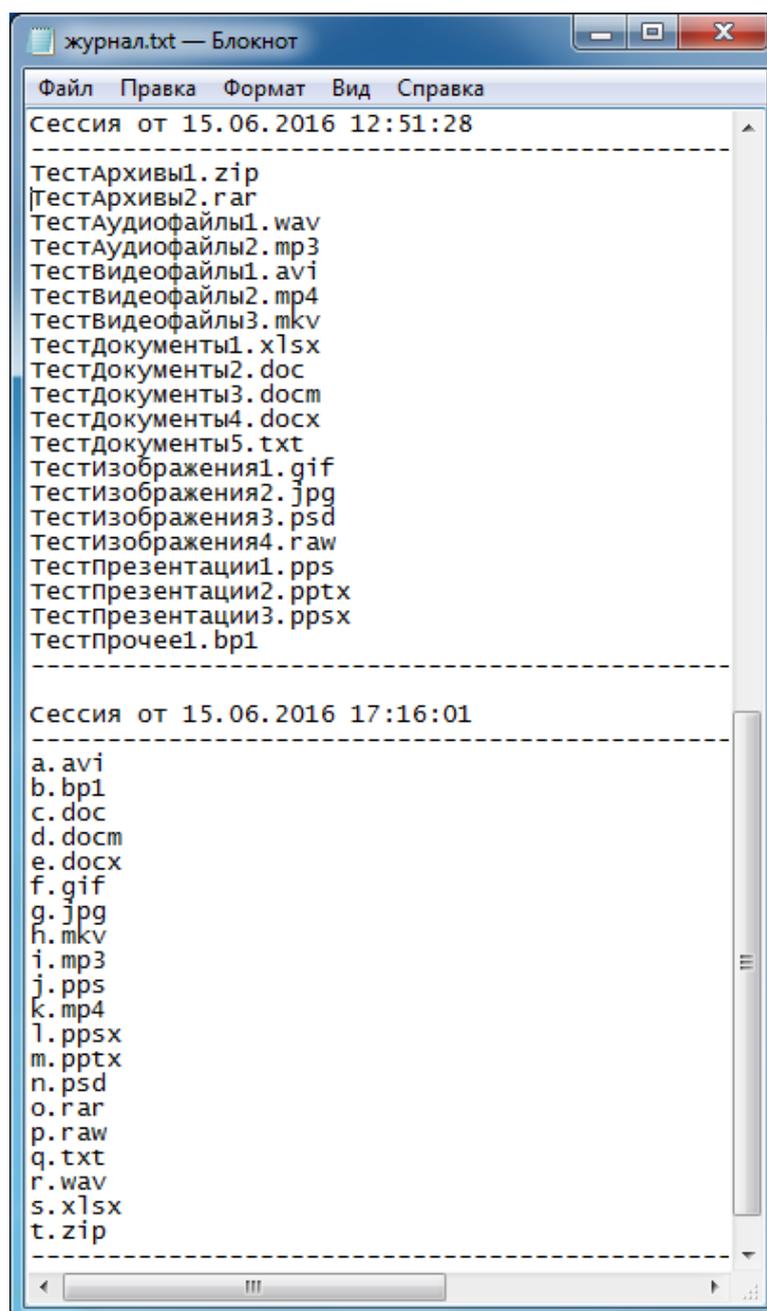


Рисунок 3.14 – Содержимое файла журнал.txt

Оба варианта призваны оказывать помощь при составлении отчетных ведомостей, взаимодействовать с ними можно по выбору пользователя. Но более весомый вклад в информативность несет вкладка «Отчеты». Выбрав нужную папку, пользователь может не только просмотреть имеющиеся в типовом каталоге сведения. Программа также выдает информацию о количестве файлов в папках.

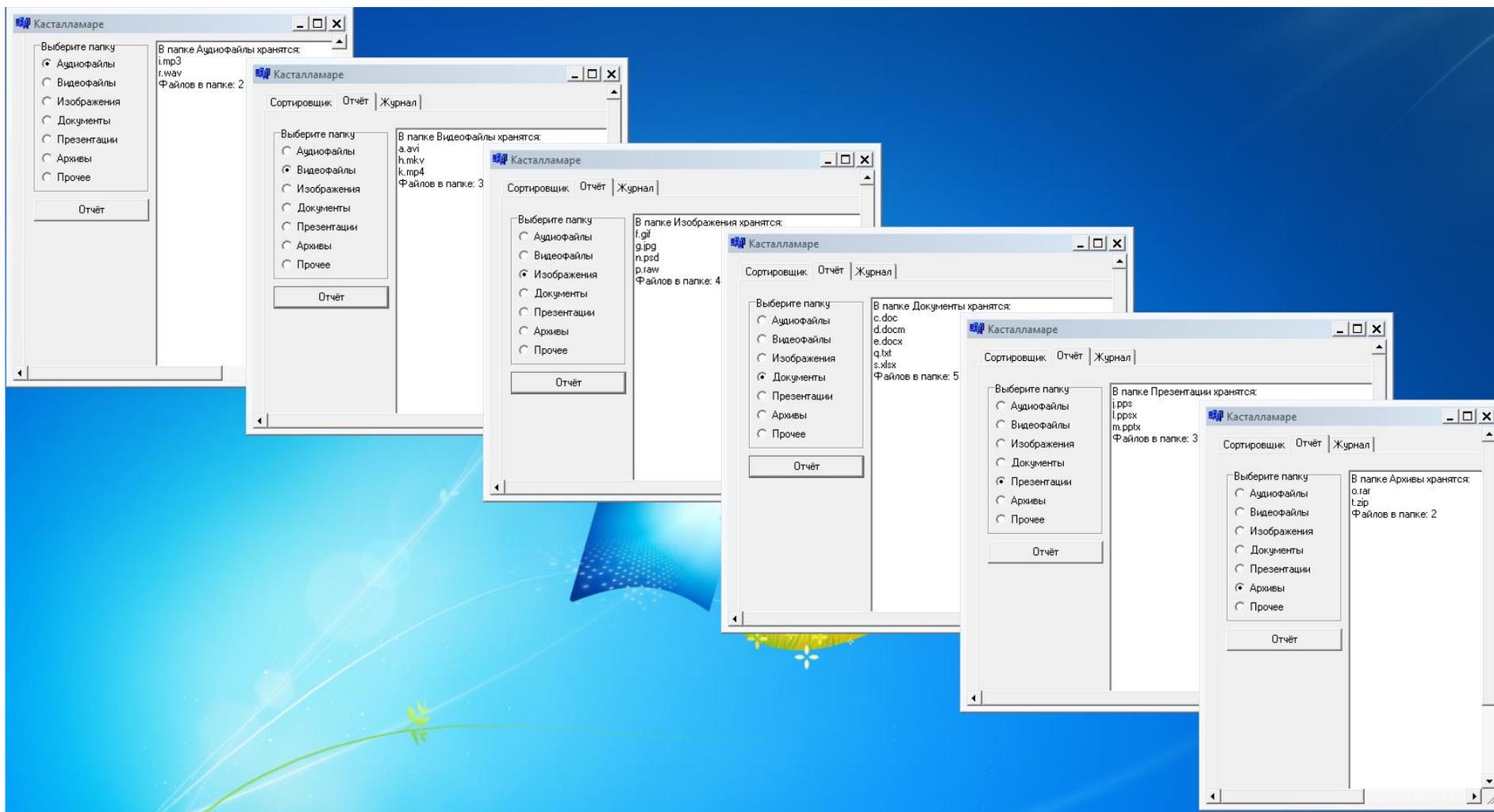


Рисунок 3.15 – Результаты функционирования вкладки «Отчеты»

Данная функция является наиболее полезной для ведения отчетности, так как ведет числовой подсчет файлов. Сотруднику предприятия остается лишь сформировать ведомость по заданным критериям и с требуемыми данными.

При тестировании системы хранения корпоративного контента на основе открытого программного обеспечения для ООО «Касталламаре» ошибок и сбоев выявлено не было. Программное приложение исправно функционирует и выполняет поставленные перед ним задачи. Также был наглядно продемонстрирован процесс работы системы, ее особенности и спецификации.

Листинг программного приложения для ООО «Касталламаре» представлен в приложении Б.

3.5 Целесообразность разработки с экономической точки зрения

Экономическая целесообразность программной разработки оценивается как разница между расходами на разработку и её эффективностью. Под эффективностью понимается результативность экономической системы, выражающаяся в отношении полезных конечных результатов её функционирования к затраченным ресурсам.

Разработанная система хранения корпоративного контента позволит уменьшить время и силы, затрачиваемые на сортировку массивов неструктурированных данных. Упростится работа сотрудников организации за счет того, что она возьмет на себя функции рутинной механической работы и позволит избежать ошибок при систематизации. Это улучшит условия труда и, как следствие, повысит производительность.

В таблице 3.1 приводятся временные затраты на разработку проектного решения.

Таблица 3.1 – Оценка трудоёмкости разработки

№ п/п	Наименование этапа разработки	Трудоёмкость этапа, час
1	Изучение предметной области	8
2	Изучение нормативных документов, литературы и деятельности организации	14
3	Постановка задачи, разработка алгоритма и структуры приложения	16
4	Проектирование и разработка программного продукта	27
5	Тестирование разработки	5

Для работ по каждому этапу определен специалист: разработчик.

Затраты на реализацию системы подразделяем на капитальные или единовременные, а также на эксплуатационные или текущие. Расчет проводится по отдельным статьям:

- прямые материальные расходы;
- фонд оплаты труда;
- отчисления на социальные нужды;
- амортизационные отчисления;
- накладные затраты;
- прочие расходы.

В таблице 3.2 представлен расчет материальных расходов.

Таблица 3.2 – Расчет прямых материальных расходов

Материалы	Ед. измер.	Кол- во	Цена за ед. (руб.)	Стоимость (руб.)
Бумага	лист	250	1	250
Картридж для принтера	шт.	1	1000	1000
Диск CD	шт.	5	25	125
Расходы на электроэнергию	кВ/ч	50	3	176,5
Итого				1551,5

К прямым затратам были отнесены:

- бумага;
- картридж для принтера;
- диск CD;
- расходы на электроэнергию.

Прямые материальные затраты составили 1551,5 рубля.

Для подсчетов фонда оплаты труда надлежит провести расчет заработной платы разработчика системы и оформить баланс рабочего времени. Полученные данные приведены в таблице 3.3.

Таблица 3.3 – Баланс рабочего времени.

П/п	Наименование показателей	ИТР
1	Число календарных дней в году	365
2	Число выходных и нерабочих дней в году	118
3	Число рабочих дней в году	247
4	Невыходы на работу:	
	А) по болезни	0
	Б) очередной отпуск	0
5	Фактическое число рабочих дней в году	253
6	Продолжительность рабочего дня	8
7	Годовой фонд рабочего времени (час.)	1976

Часовую ставку заработной платы (Чс) определяем по формуле:

$$\text{Чс} = (З * п * к) / \Phi, \quad (3.1)$$

где Чс - месячная зарплата, руб.;

п - число месяцев в году, исключая отпуск;

к - коэффициент, учитывающий премии из фонда зарплаты;

Φ - фактический годовой фонд рабочего времени, час.

Среднемесячная зарплата разработчика программных приложений составляет 18000 руб.

При $З = 18000$ руб., $n = 11$, а $\Phi = 1976$ час, $k = 1,3$ получаем $Чс = 130,26$ руб./час.

Размер основной заработной платы определяется временными затратами на реализацию проекта и стоимости часа работ специалиста.

Основную заработную плату рассчитываем по графику основных этапов работ. Дополнительная заработная плата имеет произвольный разброс до 15% от основной.

Расчет фонда оплаты труда представлен в таблице 3.4.

Таблица 3.4 – Расчет фонда оплаты труда

Этапы разработки	Время (час)	Часовая ставка (руб.)	Сумма (руб.)
Изучение предметной области	8	130,26	1042
Изучение нормативных документов, литературы и деятельности организации	14	130,26	1823,64
Постановка задачи, разработка алгоритма и структуры приложения	16	130,26	2084,16
Проектирование и разработка программного продукта	20	130,26	2605,2
Тестирование разработки	12	130,26	1563,12
Основная заработная плата(итого)			9118,12
Дополнительная заработная плата			1185,35
Итого			10303,5

Имея итоговый расчет фонда оплаты труда, определяем размер отчислений на социальные нужды.

В таблице 3.5 представим расчет этих отчислений.

Таблица 3.5 – Отчисления на социальные нужды

Отчисления	Доля от фонда оплаты труда (%)	Сумма (руб.)
В пенсионный фонд	22	2005,98
Страховые тарифы	0,2	18,23
Медицинское страхование	5,1	465,02
Социальное страхование	2,9	264,42
Итого	30,2	2753,65

В таблице 3.6 приведена полная смета на разработку программного приложения:

Таблица 3.6 – Расчет затрат на разработку ИС

Статья расхода	Сумма (руб.)
Фонд оплаты труда	10303,5
Отчисления на социальные нужды	2753,65
Материальные затраты	1551,5
Амортизационные отчисления	-
Прочие расходы	-
Итого	14582,15

В таблице 3.7 представлен расчет затрат на эксплуатацию системы в месяц:

Таблица 3.7 – Расчет затрат на эксплуатацию системы

Статья расхода	Сумма (руб.)
Заработная плата менеджера по контенту	5000
Отчисления на социальные нужды	1510
Затраты на электроэнергию	68,34
Итого	6578,34

Расчет планируемые поступления денежных средств от реализации системы хранения по месяцам приводится для оценки инвестиционного проекта. Эти данные представлены в таблице 3.8.

Таблица 3.8 – Расчет планируемых поступлений.

	Увеличение притока данных, %	Сумма, руб.	Затраты, руб.	CF, руб.	$cF_t^{(\Sigma)}$ руб.
1	0	0	6578,34	-6578,34	-6578,34
2	105	8400	6578,34	1821,66	-4756,68
3	110	8800	6578,34	2221,66	-2535,02
4	115	9200	6578,34	2621,66	86,64
5	120	9600	6578,34	3021,66	3108,3
6	130	10400	6578,34	3821,66	6929,96
7	135	10800	6578,34	4221,66	11151,62
8	140	11200	6578,34	4621,66	15773,28
9	140	11200	6578,34	4621,66	20394,94
10	150	12000	6578,34	5421,66	25816,6
11	150	12000	6578,34	5421,66	31238,26
12	160	12800	6578,34	6221,66	37459,92

Далее необходимо провести оценку инвестиционного проекта подсистемы учёта.

Первым этапом является расчет NPV (чистой текущей стоимости).

$$NPV = \sum_{t=1}^n \frac{cF_t}{k + q} - I_0, \text{ где} \quad (3.2)$$

NPV – чистая текущая стоимость инвестиций;

CF – поступление денежных средств в конце t- ого периода;

q – банковская ставка;

I – стоимость реализации инвестиционного проекта (инвестиции).

$$NPV = 37459,92 / (1 + 0,11) - 14582,15 = 19165,52$$

Для реализации проекта NPV составляет 19165,52 руб.

Положительное значение чистой текущей стоимости инвестиций означает, что вложение капитала в реализуемую систему эффективно.

Дальнейшим этапом является расчет рентабельности инвестиций. Он проводится по формуле:

$$PI = \left[\sum_{t=1}^n \frac{cF_t}{(1 + q)^t} \right] / \left[\sum_{t=1}^n \frac{I_t}{(1 + q)^t} \right], \text{ где} \quad (3.3)$$

PI – рентабельность инвестиций.

$$PI = (37459,92 / (1 + 0,11)) / 14582,15 = 224 = 224\%$$

Этот показатель определяет, в какой мере возрастает ценность фирмы в расчете на 1 руб. Для реализации проекта по разработке системы хранения рентабельность составляет 224%.

После рассчитаем период окупаемости проекта. Расчет окупаемости осуществляется по формуле:

$$PP = \frac{I_0}{cF_t^{(\Sigma)}}, \quad (3.4)$$

где PP – период окупаемости (лет);

I_0 – первоначальные инвестиции;

$cF_t^{(\Sigma)}$ – годовая сумма денежных поступлений от реализации инвестированного проекта.

$$PP = 14582,15 / (37459,92 / (1 + 0,11)) = 0,4 = 22 \text{ нед.}$$

Осуществив расчет периода окупаемости, было определено, что этот показатель составит 22 недели.

Подытожив полученные результаты можно сделать вывод, что затраты на разработку и запуск системы окупятся спустя 22 недели со дня внедрения программного приложения в эксплуатацию, расчет чистой текущей стоимости инвестиций дал положительное значение, рентабельность составила 224 %.

Внедрение системы увеличит эффективность работы сотрудников и существенно уменьшит временные затраты на рутинные операции. Кроме этого программное приложение позволит избежать найма еще одного сотрудника на идентичную должность, а также уменьшить вероятность возникновения ошибок при систематизации данных.

3.6 SWOT-анализ разработки

SWOT-анализ (с англ. SWOT Analysis) – вид ситуационного анализа, позволяющий оценить текущую и будущую конкурентоспособность товара компании на рынке с помощью анализа внутренней и внешней среды.

Расшифровка SWOT: S=Strengths, сильные стороны товара; W=Weaknesses, слабые стороны товара; O=Opportunities, возможности компании; T=Threats, угрозы компании.

Задача SWOT-анализа – дать структурированное описание ситуации, относительно которой нужно принять какое-либо решение. Выводы, сделанные на его основе, носят описательный характер без рекомендаций и расстановки приоритетов. [34]

Для деятельности целых организаций анализ выходит достаточно общим и содержит не так много полезных сведений, но анализ для определенных продуктов дает большой объем информации для дальнейших размышлений. Методика проведения SWOT-анализа довольно проста, благодаря чему можно сосредоточиться на конкретных фактах, без затрат времени на различные функциональные аспекты.

Использование SWOT-анализа позволит систематизировать всю имеющуюся информацию и, ввиду всех обстоятельств, принимать взвешенные решения, касающиеся дальнейшей разработки программного средства.

Основные этапы SWOT–анализа:

- определение сильных и слабых сторон разработки, ее возможностей и угроз;
- формирование и анализ SWOT–матрицы;
- формулировка заключения перспективы разработки.

В таблице 3.9 приведена SWOT – матрица системы хранения корпоративного контента на основе открытого программного обеспечения. Возможности – это то, что не реализовано на данный момент, но может быть притворено в жизнь в будущем.

Угрозы, которые существуют, или могут возникнуть в будущем. Угрозы могут привести к тому, что разработанная система окажется неконкурентоспособной, в худшем же случае никому не нужной совсем.

Для подробного обоснования эффективности разработанной подсистемы проведем swot анализ. Этот анализ позволит выявить сильные и слабые стороны подсистемы и дать ей оценку в целом. Ниже представлена матрица анализа для проекта в таблице 3.9:

Таблица 3.9– матрица SWOT - анализа

Сильные стороны	Возможности		Угрозы.		Итого
	1. Создание новых методов и способов анализа	2. Совершенствование разработки	1. Появление новых конкурентов	2. Быстрое моральное устаревание	
1. Низкая стоимость разработки	++	0	++	0	+4
2. Низкая стоимость сопровождения	++	++	+	+	+6
3. Ускоренный процесс сортировки	++	++	+	++	+7
Итого	+6	+4	+4	+3	+17
Слабые стороны					
1. Недостаточное количество запрограммированных форматов	-	-	-	-	-4
2. Нехватка квалифицированных кадров	-	--	-	--	-5
3. Трудность разработки сложных методов и способов анализа	--	-	-	0	-5
Итого	-4	-4	-3	-3	-14
Общий итог	+1	0	+2	0	+3

Проанализировав полученную матрицу, можно сделать следующие выводы:

1) Самой важной сильной стороной разработки является уникальность программного приложения, возможность расширения базы опознаваемых форматов файлов, а также возможность получать необходимые сведения о процессе работы программы. В перспективе при развитии разработки необходимо увеличить спектр обрабатываемых типов и расширений файлов.

2) Из слабых сторон разработки следует выделить трудность разработки сложных методов и способов анализа, т.к. в случае разработки сложных математических методов требуется дополнительное

программирование, из-за того, что заложенных в разработку алгоритмов может быть недостаточно, а при нехватке квалифицированных кадров это может быть большой проблемой.

3) Из возможностей разработки наиболее важной является возможность создать более обширную базу распознавания и повышения качества сортировки. Но следует брать в учет, что реализация данной возможности очень сильно зависит от наличия квалифицированных работников.

4) Наиболее опасной угрозой является моральное устаревание разработки, связанное с ограниченным количеством утвержденных на начальном этапе форматов и если не продолжить их дальнейшую проработку, то развитие продукта замедлится.

5) Система не нуждается в использовании промежуточной базы данных. Она связывается и взаимодействует напрямую с Проводником, что позволяет хранить отсортированный массив данных на локальном диске компьютера. Это выделяет разработку на фоне других программных продуктов. А благодаря низкой стоимости сопровождения и открытости программного обеспечения позволяет сделать разработку конкурентоспособной, а потому появление конкурентов маловероятно.

6) Проект в ближайшее время будет перспективен, а если будет развиваться, то будет перспективен и в будущем.

ЗАКЛЮЧЕНИЕ

В данной работе была разработана система хранения корпоративного контента на основе открытого программного обеспечения ООО «Касталламаре», работающего в сфере розничных продаж.

В ходе выполнения работы предлагалось найти решение, которое позволяло бы автоматизировать деятельность ООО «Касталламаре». Целью выпускной квалификационной работы является разработка системы хранения корпоративного контента на основе открытого программного обеспечения ООО «Касталламаре», работающего в сфере розничных продаж. Актуальность темы обусловлена необходимостью организовать обработку, хранение и систематизацию информации, представленной в различных форматах.

Для достижения поставленной цели были решены следующие задачи:

- изучение предметной области;
- обоснование необходимости разработки;
- выработка концепции по оптимизации управления неструктурированными данными, ориентированной на перспективные цели;
- проведение анализа;
- выбор средств и технологий для разработки;
- создание системы обработки и распределения массива информации различных типов и расширений;
- тестирование системы хранения корпоративного контента.

В созданной информационной подсистеме автоматизации деятельности, были спроектированы работы со справочниками, документами и отчетной информацией.

В данной выпускной квалификационной работе были реализованы следующие возможности:

- систематизация данных;
- группировка контента по файловым расширениям;

- упразднение рутинных операций по сортировке;
- экономия временных и трудовых ресурсов;
- обеспечение хранения данных непосредственно на локальном диске компьютера;
- устранение возможности ошибки при сортировке информации или утери файлов в результате уменьшения действия человеческого фактора;
- оперативное получение информации при помощи встроенных отчетов;
- ведение журнала по датам и времени загрузки информации.

Благодаря простому и удобному интерфейсу, увеличилась скорость работы сотрудников и уменьшилась в разы количество рутинной работы, что оправдывает проведенный выше экономический анализ, который эффективен с экономической стороны. В дальнейшем предполагается улучшение спроектированной системы.

Цель выпускной квалификационной работы достигнута.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Мартишин, С.А. Основы теории надежности информационных систем [Текст] / С.А. Мартишин. – М.: –Инфра-М, Форум, 2015. – 256 с.
2. Криницкий, Н.А. Автоматизированные информационные системы [Текст] / Н.А. Криницкий, Г.А. Миронов, Г.Д. Фролов.–М.: – Наука, 2014.–384с.
3. Маклаков, С. В. ВРwin и ERwin. CASE – средства разработки информационных систем [Текст] / С. В. Маклаков – Изд.: "Диалог–Мифи" – 2014.–295 с.
4. Назад – к простоте! Методологии моделирования [Электронный ресурс] – Режим доступа: <http://podelise.ru/docs/75050/index-700.html>, свободный.
5. Таетр, Т. Надежность программного обеспечения [Текст] / Т. Таетр, М. Липов, Э. Нельсое. – М.: ИЛ, 2008. – 323 с.
6. Волкова В.Н. Информационные системы [Текст] / Под ред. В.Н. Волковой, Б.И. Кузина. – СПб.: СПбГТУ, – 2001. – 216 с.
7. Степанов, А. Н. Информатика: Учебник для вузов [Текст] / А.Н. Степанов – Санкт–Петербург: Питер, 2010. – 720 с.
8. Титоренко, Г.А. Информационные технологии управления [Текст] / Г.А. Титоренко – М.: ЮНИТИ–ДАНА, 2014
9. Насакин, Г.С. Три кита разработки [Текст] / Г.С. Насакин, А.М. Арапов – М., 2012 – №31–32 – с. 32–33
10. Тидвелл, Дженнифер. Разработка пользовательского интерфейса [Текст] / Дженнифер Тидвелл Питер, 2012 . – 416с.
11. Брага, И.И. Автоматизированные информационные технологии в экономике [Текст] / И.И. Брага, Н.Г. Бубнова, Л.А. Вдовенко и др. Под ред. проф. Г.А. Титоренко. — М.: Компьютер, ЮНИТИ, 2012. – 400 с.

12. Бондарь, А.Г. Interbase и Firebird. Практическое руководство для умных пользователей и начинающих разработчиков [Текст] / А.Г. Бондарь – СПб.: БХВ–Петербург, 2007. – 592 с.
13. Лапушта, М.А. Справочник директора предприятия [Текст] / М.А. Лапушта. – М.: Издательство Инфра, 2012 – 48 с.
14. Гвоздева, Т. В. Проектирование информационных систем [Текст] / Т. В. Гвоздева, Б.А. Баллод. – Ростов н/Д : Феникс, 2009. – 508 с.
15. Бычкова, С.М. Информационные технологии в бухгалтерском учете и аудите [Текст] / С.М. Бычкова, С. В. Ивахненко – М.: «Проспект», 2014. – 216 с.
16. Зайцева, Т. В. Методические указания по содержанию, выполнению и защите выпускных квалификационных работ [Текст] / Т. В. Зайцева, О.П. Пусная, Н.П. Путивцева – Белгород: Изд-во БелГУ, 2015. – 111с.
17. Грекул, В.И. Проектирование информационных систем [Текст] / В.И.Грекул – М.: Интернет– университет информационных технологий, 2005. – 304 с.
18. Баронов, В. В. Информационные технологии и управление предприятием [Текст] / В. В. Баронов, Г.Н. Калянов, Ю.Н. Попов, И.Н. Титовский. – М.: Компания АйТи, 2012. – 328 с.
19. SWOT анализ «с нуля» менее чем за час: теория и применение на практике. [Электронный ресурс] – Режим доступа: http://powerbranding.ru/marketing_lessons/swot-analisis/, свободный.
20. Национальный открытый университет Интуит |Лекция 9: Информационное обеспечение ИС [Электронный ресурс] – Режим доступа: <http://www.intuit.ru/studies/courses/2195/55/lecture/1634>, свободный.
21. Метод SWOT анализа в стратегическом управлении [Электронный ресурс] – Режим доступа: <http://powerbranding.ru/biznes-analiz/swot>, свободный.

22. Операционная система [Электронный ресурс]– Режим доступа: [http://ru.wikipedia.org/wiki/Операционная система](http://ru.wikipedia.org/wiki/Операционная_система), свободный.
23. Прикладное программное обеспечение [Электронный ресурс]– Режим доступа: <http://tpl-it.wikispaces.com> , свободный.
24. Проводник Windows [Электронный ресурс] – Режим доступа: http://ru.wikipedia.org/wiki/Проводник_Windows, свободный.
25. Программа финансового анализа [Электронный ресурс] – Режим доступа: <http://1fin.ru>, свободный.
26. Котлер, Ф. Стратегический менеджмент по Котлеру. Лучшие приемы и методы[Текст] / Ф. Колтер, Р. Бергер, Н. Бикхофф. – М.: Альпина Паблишер, 2012. – 144 с.
27. Терехов, С. А. Нейросетевые информационные модели сложных инженерных систем: Нейроинформатика [Текст] / А. Н. Горбань, В. Л. Дунин–Барковский, А. Н. Кирдин и др. – Новосибирск: Наука. Сибирское предприятие РАН, 2005. – 296 с.
28. Техническое обеспечение информационной системы – ИС элекрес [Электронный ресурс] –Режим доступа: <http://www.itstan.ru/it-i-is/tehicheskoe-obespechenie-informacionnoj-sistemy-is.html>, свободный.
29. Технологический процесс обработки информации [Электронный ресурс] – Режим доступа: http://dic.academic.ru/dic.nsf/fin_enc/30412, свободный.
30. Что такое системы программирования? [Электронный ресурс] – Режим доступа: http://book.kbsu.ru/theory/chapter6/1_6_11.html, свободный.
31. Гарсиа–Молина, Г. Системы баз данных: Полный курс [Текст] / Г. Гарсиа–Молина, Д.Д. Ульмон, Д. Уидом.–М. : Вильямс, 2011. – 1088 с.
32. Расширение имени файла [Электронный ресурс] – http://ru.wikipedia.org/wiki/Расширение_имени_файла, свободный.
33. Майсак, О. С. SWOT-анализ: объект, факторы, стратегии. Проблема поиска связей между факторами // Прикаспийский журнал: управление и высокие технологии. – 2013. – № 1 (21). – С. 151–157

34. Деверадж, С. Окупаемость информационных технологий [Текст] / С. Демередж, Р. Кохли – М.: ЗАО «Новый издательский дом», 2013. – 192 с
35. Кузнецов, С.Д. Основы современных баз данных [Текст] /С.Д.Кузнецов.– Калининград: Издательская группа ВHV, 2008. – 349 с.
36. Гордеев, А. В. Операционные системы: Учебник для вузов [Текст] / А. В. Гордеев.– СПб.: Питер, 2007. – 416 с.
37. Шаньгин, В. Ф. Информационная безопасность [Текст] / В. Ф Шаньгин. – М.: «ДМК–Пресс», 2014. – 702 с.
38. Жемчугов, М.К. Как построить эффективную организацию в России: стратегия и структура // Проблемы экономики и менеджмента. – 2012. – № 3.
39. Страуструп, Б. Программирование: принципы и практика использования С++, исправленное издание [Текст] Б.Страуструп. – М.: Вильямс, 2011. – С. 1248.
40. Павловская, Т.А. С/С++. Структурное и объектно-ориентированное программирование / Т.А. Павловская – СПб.: Питер, 2010. – 352 с.

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ А

Таблица А.1 – Сравнительная характеристика средств разработки

Средство разработки и характеристика	Delphi	Visual C++	Borland C++ Builder	Visual FoxPro
Наличие опыта разработки с использованием данного программного продукта;	1	3	5	2
Требования по ресурсам;	5	2	5	5
Поддержка операционной системы;	5	5	5	4
Наглядность разработки интерфейса;	4	5	5	4
Предоставляемые возможности работы с Проводником;	5	4	4	3
Скорость работы разработанного программного обеспечения;	4	4	5	3
Обработка исключительных ситуаций;	4	4	4	4
Время создания разработанного программного обеспечения;	3	4	5	3
Удобство эксплуатации;	3	5	5	4
Всего:	34	39	43	32

ПРИЛОЖЕНИЕ Б

```
//-----

#include <vc1.h>
#pragma hdrstop

#include "Unit1.h"
#include "Unit2.h"

//-----

#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;

//-----
__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
}

//-----

int MoveFiles( AnsiString sSourceDir, AnsiString sTargetDir, bool toCopy = true )
{
    SHFILEOPSTRUCT sh;
    sh.fFlags = FOF_NOCONFIRMATION | FOF_SILENT | FOF_FILESONLY;
    sh.wFunc = toCopy ? FO_COPY : FO_MOVE;
    sh.pFrom = sSourceDir.c_str();
    sh.pTo = sTargetDir.c_str();
    sh.hNameMappings = NULL;
    sh.lpszProgressTitle = NULL;
    return SHFileOperation ( &sh );
}

//-----

void ListFiles(AnsiString path, TStringList* List)
{
    TSearchRec sr;
    if (FindFirst(path+"*.*", faAnyFile, sr) == 0)
    {
        do
        {

```

```

if (sr.Attr & faDirectory)
{
if (sr.Name!=".")
if (sr.Name!="..")
{
ListFiles(path+sr.Name+"\\",List);
}
}
else
{
AnsiString Ext=ExtractFileExt(sr.Name).UpperCase();
if (Ext==".CPP"); {
AnsiString Ex=ExtractFileExt(path+sr.Name);
if (Ex==".mp3" || Ex==".wav" || Ex==".wma" || Ex==".dat" || Ex==".aif") {
List->Add(sr.Name);

!MoveFiles(path+sr.Name, "c:\\castellammare\\Каталоги\\Аудиофайлы");
AnsiString FileName=path+sr.Name;
DeleteFile(FileName);}
else if (Ex==".avi" || Ex==".wmv" || Ex==".mkv" || Ex==".mov" || Ex==".mp4") {
List->Add(sr.Name);

!MoveFiles(path+sr.Name, "c:\\castellammare\\Каталоги\\Видеофайлы");
AnsiString FileName=path+sr.Name;
DeleteFile(FileName);}
else if (Ex==".gif" || Ex==".jpg" || Ex==".jpeg" || Ex==".png" || Ex==".psd" || Ex==".raw") {
List->Add(sr.Name);

!MoveFiles(path+sr.Name, "c:\\castellammare\\Каталоги\\Изображения");
AnsiString FileName=path+sr.Name;
DeleteFile(FileName);}
else if (Ex==".gif" || Ex==".doc" || Ex==".docx" || Ex==".docm" || Ex==".dot" || Ex==".dotx" || Ex==".txt" || Ex==".odt" ||
Ex==".fodt" || Ex==".odm" || Ex==".xls" || Ex==".xlsx" || Ex==".xml" || Ex==".pdf" || Ex==".djvu") {
List->Add(sr.Name);

!MoveFiles(path+sr.Name, "c:\\castellammare\\Каталоги\\Документы");
AnsiString FileName=path+sr.Name;
DeleteFile(FileName);}
else if (Ex==".ppt" || Ex==".pptx" || Ex==".pps" || Ex==".ppsx") {
List->Add(sr.Name);

!MoveFiles(path+sr.Name, "c:\\castellammare\\Каталоги\\Презентации");
AnsiString FileName=path+sr.Name;

```

```

DeleteFile(fileName);}
else if (Ex==" .7z" || Ex==" .rar" || Ex==" .zip") {
    List->Add(sr.Name);

    !MoveFiles(path+sr.Name, "c:\\castellammare\\Каталоги\\Архивы");
    AnsiString FileName=path+sr.Name;
DeleteFile(fileName);}

else List->Add(sr.Name); {!MoveFiles(path+sr.Name, "c:\\castellammare\\Каталоги\\Прочее");
    AnsiString FileName=path+sr.Name;
DeleteFile(fileName);}

}
}
}
while (FindNext(sr) == 0);
FindClose(sr);
ShowMessage("Готово");
}
Application->ProcessMessages();
}
//-----

void __fastcall TForm1::Button1Click(TObject *Sender)
{
Memo1->Lines->Add("Сессия от "+Now().DateTimeString());
Memo1->Lines->Add("-----");
ListFiles("c:\\castellammare\\Промежуточная\\", Memo1->Lines);
Memo1->Lines->Add("-----");
Memo3->Lines->Add(Memo1->Lines->Text) ;
Memo3->Lines->SaveToFile("c:\\castellammare\\журнал.txt");
}
//-----

void Report(AnsiString path, TStrings* List)
{
TSearchRec sr;
if (FindFirst(path+"*.*", faAnyFile, sr) == 0)
{
do
{
if (sr.Attr & faDirectory)

```

```

{
if (sr.Name!=".")
if (sr.Name!="..")
{
ListFiles(path+sr.Name+"\\",List);
}
}
else
{
AnsiString Ext=ExtractFileExt(sr.Name).UpperCase();
if (Ext==".CPP"); {

List->Add(sr.Name);
}
}
}
while (FindNext(sr) == 0);
FindClose(sr);

ShowMessage("Готово");

}
Application->ProcessMessages();
}
//-----

void __fastcall TForm1::Button2Click(TObject *Sender)
{
Memo2->Lines->Clear();
if (RadioGroup1->ItemIndex==0){Memo2->Lines->Add("В папке Аудиофайлы хранятся.");
Report("c:\\castellammare\\Каталоги\\Аудиофайлы\\", Memo2->Lines);}
else if (RadioGroup1->ItemIndex==1){Memo2->Lines->Add("В папке Видеофайлы хранятся.");
Report("c:\\castellammare\\Каталоги\\Видеофайлы\\", Memo2->Lines);}
else if (RadioGroup1->ItemIndex==2){Memo2->Lines->Add("В папке Изображения хранятся.");
Report("c:\\castellammare\\Каталоги\\Изображения\\", Memo2->Lines);}
else if (RadioGroup1->ItemIndex==3) {Memo2->Lines->Add("В папке Документы хранятся.");
Report("c:\\castellammare\\Каталоги\\Документы\\", Memo2->Lines);}
else if (RadioGroup1->ItemIndex==4) {Memo2->Lines->Add("В папке Презентации хранятся.");
Report("c:\\castellammare\\Каталоги\\Презентации\\", Memo2->Lines);}
else if (RadioGroup1->ItemIndex==5) {Memo2->Lines->Add("В папке Архивы хранятся.");
Report("c:\\castellammare\\Каталоги\\Архивы\\", Memo2->Lines);}
else if (RadioGroup1->ItemIndex==6) {Memo2->Lines->Add("В папке Прочее хранятся.");
}
}

```

```
Report("c:\\castellammare\\Каталоги\\Прочее\\", Memo2->Lines);}
else ShowMessage("Выберите папку");
Memo2->Lines->Add("Файлов в папке: "+IntToStr(Memo2->Lines->Count-1));
}
//-----

void __fastcall TForm1::FormCreate(TObject *Sender)
{
Memo3->Lines->LoadFromFile("c:\\castellammare\\журнал.txt");
}
//-----

void __fastcall TForm1::Button3Click(TObject *Sender)
{
Memo1->Lines->Clear();
}
//-----
```