

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 ИНФОРМАЦИОННЫХ И РОБОТОТЕХНИЧЕСКИХ СИСТЕМ

**РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ ДЛЯ
ИНТЕРАКТИВНОГО КНИЖНОГО ТЕРМИНАЛА**

Выпускная квалификационная работа
обучающейся по направлению подготовки 09.03.02 Информационные
системы и технологии
очной формы обучения, группы 07001408
Ефимовой Кристины Олеговны

Научный руководитель
ст. преподаватель
Удовенко И. В.

БЕЛГОРОД 2018

РЕФЕРАТ

Разработка информационной системы для интерактивного книжного терминала. – Ефимовой Кристины Олеговны, выпускная квалификационная работа бакалавра, Белгород, ФГАОУ ВО "Белгородский государственный национальный исследовательский университет" (НИУ «БелГУ»), количество страниц 44, включая приложения 50, количество рисунков 12, количество таблиц 9, количество формул 5, количество использованных источников 30.

КЛЮЧЕВЫЕ СЛОВА: информационная система, база данных, программное обеспечение, система управления базами данных.

ОБЪЕКТ ИССЛЕДОВАНИЯ: книжные магазины.

ПРЕДМЕТ ИССЛЕДОВАНИЯ: технология организации учета и обслуживание посетителей.

ЦЕЛЬ РАБОТЫ: повышение эффективности обслуживания клиентов книжного магазина за счет разработки информационной системы для интерактивного книжного терминала.

ЗАДАЧИ ИССЛЕДОВАНИЯ: анализ предметной области; обосновать необходимость автоматизации; провести структурно-функциональный анализ бизнес-процессов организации и выявить недостатки работы; провести реинжиниринг бизнес-процессов и предоставить модель "как будет"; разработать программное обеспечение.

МЕТОДЫ ИССЛЕДОВАНИЯ: метод лексико-синтаксических шаблонов; статистические методы анализа текстов на естественном языке.

ПОЛУЧЕННЫЕ РЕЗУЛЬТАТЫ: в результате работы была спроектирована и разработана информационная система для интерактивного книжного терминала.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Деятельность книжного магазина	6
1.1 Функциональная модель бизнес-процессов книжного магазина.....	7
1.2 Обслуживание клиентов книжного магазина.....	10
1.3 Математическая бизнес модель обслуживания клиентов.....	11
2 Описание оптимизации функциональной модели.....	17
2.1 Оптимизация математической модели подбора товара	19
2.2 Описание подбора товара с использованием электронного помощника ..	24
2.3 Анализ среды для разработки базы данных	25
2.3.1 Выбор среды проектирование базы данных.....	26
2.3.2 Разработка базы данных для книжного магазина.....	29
3 Проектирование программы	34
3.1 Выбор аппаратно-программной платформы	34
3.2 Выбор языка программирования	36
3.3 Выбор среды программирования	38
3.4 Разработка программы.....	40
ЗАКЛЮЧЕНИЕ	44
ПРИЛОЖЕНИЕ А	45
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	48

ВВЕДЕНИЕ

В современном мире существует большая конкуренция между различными направлениями в сфере торговли. Это не обошло стороной и книжные магазины. Кто-то может сказать, что в век информационных технологий бумажные книги потеряли свое положение на торговом рынке. Однако это утверждение будет не верно. Так как некоторые книги нельзя скачать с интернет ресурсов бесплатно, так же кто-то все еще предпочитает бумажный формат книги, а так же не отменены бумажные книги в школьных заведениях. Из этого следует, что бумажные книги все еще популярны.

Все магазины находятся в борьбе за покупателя, пытаясь увеличить свой ассортимент, предоставляя покупателю огромное количество различных товаров, они забывают о том, сколько клиент готов потратить своего драгоценного времени выбирая из представленного ассортимента то что его действительно интересует.

Зачастую приходя в книжный магазин, покупатель теряется в предоставляемом ему выборе товара. Даже несмотря на то, что магазины стараются разложить все товары согласно каким-либо критериям, или книги по жанрам, не всегда взгляд клиента совпадает с порядком сортировки магазина. В таком случае покупатель вынужден бродить по магазину в поисках нужной ему книги. Не справляясь самостоятельно, он вынужден обратиться за помощью к сотрудникам магазина, и в этом случае покупатель сталкивается с рядом других проблем:

- в магазине не оказывается сотрудников консультантов;
- сидящий на кассе сотрудник отказывается предоставить помощь в связи с тем, что это не его обязанности;
- кассир занят обслуживанием других покупателей.

В таких случаях клиенту приходится тратить свое драгоценное время на ожидания помощи, либо если не может себе этого позволить, он вынужден покинуть магазин и остаться без нужного ему товара, либо уйти к

конкурентам, в надежде что там он найдет нужный ему товар. В любом случае магазин теряет потенциального клиента, а значит и свой доход.

Цель работы: повышение эффективности обслуживания клиентов книжного магазина за счет разработки информационной системы для интерактивного книжного терминала.

Задачи дипломной работы:

- исследование предметной области;
- проектирование информационной системы;
- реализация программного приложения;
- разработка информационной системы для интерактивного книжного терминала.

В данной дипломной работе разрабатывается программное обеспечение "Электронный помощник".

Пояснительная записка выполнена на 44 страницах без приложения, и 50 страниц с приложением, содержит 12 рисунков, 9 таблиц и 5 формул.

1 Деятельность книжного магазина

Книжный магазин занимается деятельностью по продаже и закупке книг, а значит, в магазине работают сотрудники, которые выполняют весь процесс, от закупок книг у каких либо поставщиках, с которыми они заключают договора, и продажей этих книг клиентам книжного магазина. На рисунке 1.1 представлена организационная структура книжного магазина.



Рисунок 1.1 – Организационная структура книжного магазина

На рисунке 1.1 представлены основные подразделения книжного магазина такие как: директор, бухгалтерия, отдел маркетинга, отдел кадров, магазин (торговый зал), кассир, продавец, отдел поставок, склад. Каждое подразделение выполняет определенные функции [29]:

– Директор – занимает руководящую должность, обеспечивая выполнение предприятия всех обязанностей связанных договором с поставщиком. Выполняет вопросы касающиеся финансовой стороны, контролирует деятельность и выполнение своих должностных обязанностей

остальных подразделений, назначает руководителей подразделений. Обеспечивает законную деятельность предприятия.

– Бухгалтерия – выполняет работу по бухгалтерскому учету, отражает движение денежных средств, составляет платежные поручения, оплачивает аренду, расплачивается с поставщиками и контролирует вовремя были ли получены денежные средства.

– Отдел маркетинга – изучение рынка потребителя. Анализ деятельности конкурентов. Изучение интересов и спрос покупателей. Мониторинг места на рынке фирмы.

– Отдел кадров – подбор компетентных рабочих кадров, отвечающие требованиям предприятия. Составление рабочего и отпускного графика. Мониторинг рабочих по выполнению своих обязанностей. Анализ текучести кадров.

– Магазин (торговый зал: продавец, кассир) – продавец расставляет товар согласно установленному порядку, размещает цену, информирует покупателей о проходящих скидках или акциях. Предоставляет помощь клиентам в поиске товара. Кассир, подчитывает окончательную сумму товара для покупателя, производит расчеты с поставщиками.

– Отдел поставок – анализ рынка поставщиков, участвует в выборе поставщиков, производит закупки, а также контролирует доставку.

– Склад – принятие, размещение и хранение товара. Предоставление товара в торговый зал продавцу для дальнейшей его продажи потребителю.

1.1 Функциональная модель бизнес-процессов книжного магазина

Прежде чем вносить изменения в какой либо процесс для начала необходимо разобраться, как первоначально все работает. Для этого необходимо создать функциональную модель бизнес-процессов книжного

магазина в системе разработки AllFusion ERwin Data Modeler. В рамках проектирования была выбрана методология IDEF0.

IDEF0 – это методология функционального моделирования и графическая нотация, предназначенная для формализации и описания бизнес-процессов. Представление происходит в виде набора элементов-работ, которые взаимодействуют между собой, а также показываются информационные, людские и производственные ресурсы, потребляемой каждой работой [15]. Контекстная диаграмма представлена на рисунке 1.2

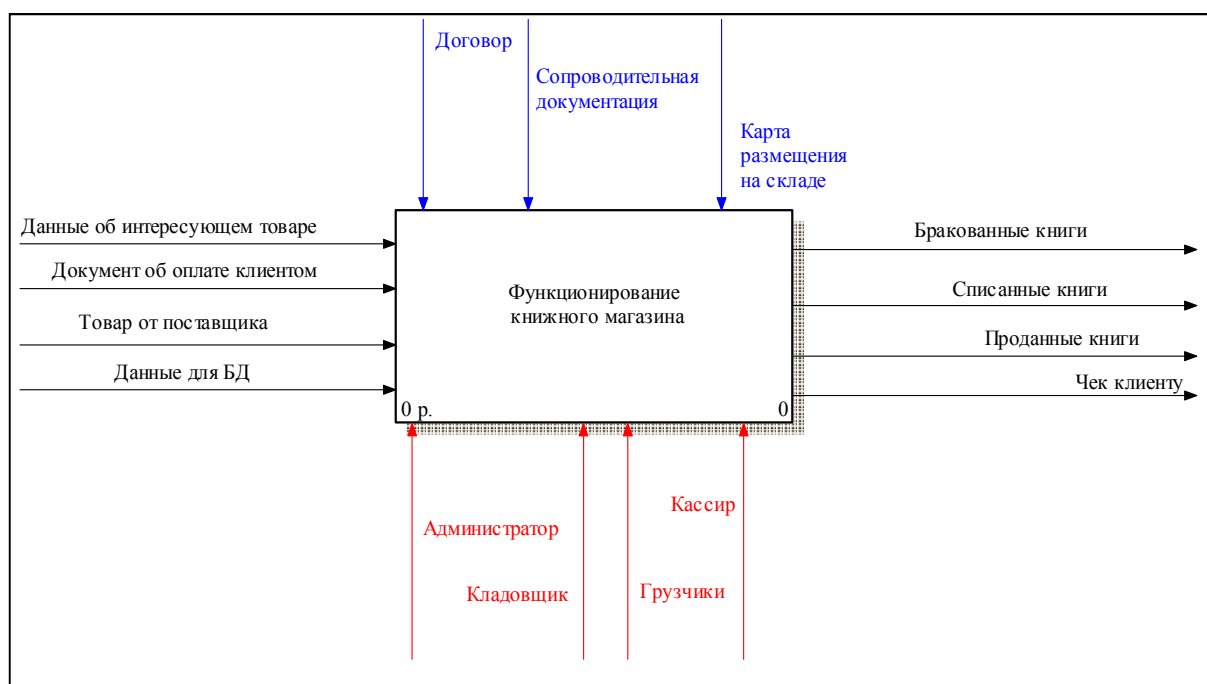


Рисунок 1.2 – Контекстная диаграмма

На рисунке 1.2 контекстной диаграммы изображены входные параметры такие как: товар от поставщика; данные об интересующем товаре; документ об оплате клиентом; данные для БД. Выходные параметры: бракованные книги; списанные книги; чек клиенту; проданные книги. Управление: договор; сопроводительная документация; карта размещение на складе. Механизм – сотрудники.

На рисунке 1.3 представлена декомпозиция контекстной диаграммы.

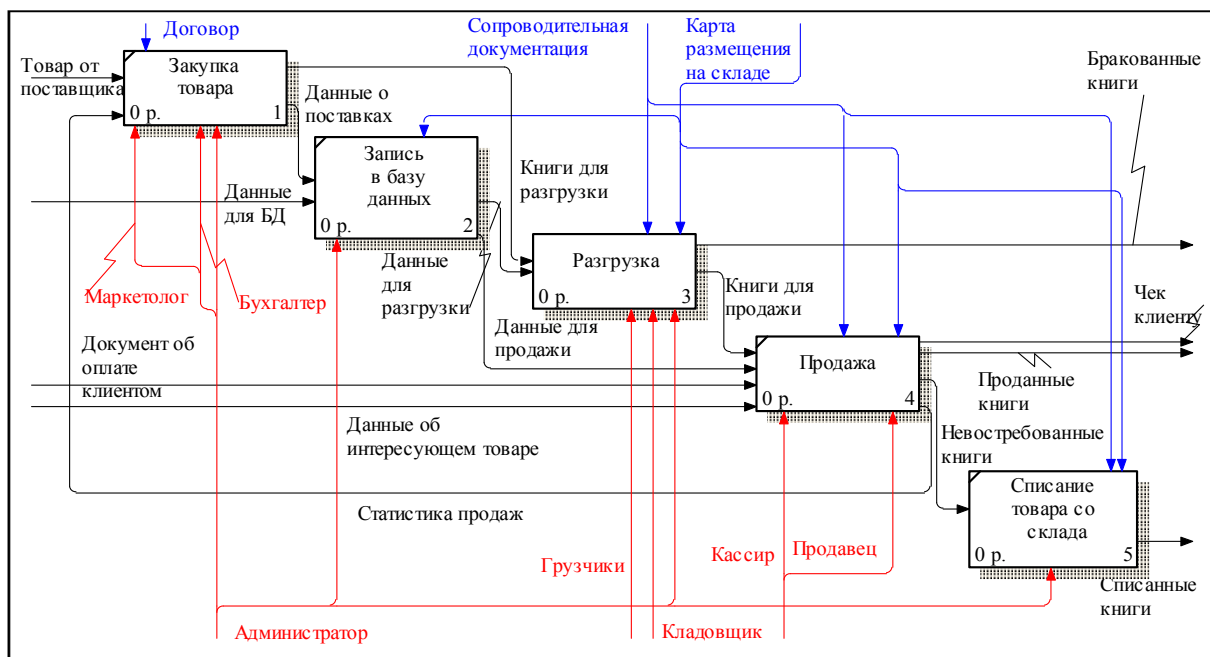


Рисунок 1.3 – Декомпозиция главной диаграммы

На рисунке 1.3 изображена декомпозиция контекстной диаграммы. Первым блоком является "закупка товара". В него входит: товар от поставщика, статистика продаж (предоставляет информацию, о наиболее востребованных книг среди покупателей). На выходе: данные о поставках, книги для разгрузки. Управлением является договор. Механизм представляет администратор, маркетолог и бухгалтер.

Вторым блоком является "запись в базу данных". Входными данными являются: данные для БД, а так же выходные данные с блока "закупка товара", данные о поставщиках. Выходными данными являются данные для разгрузки для блока "разгрузка"; данные для продажи. Механизмом является администратор. Управление, карта размещения на складе.

Третьим блоком является "разгрузка". Для этого блока входными параметрами являются: книги для разгрузки, данные для разгрузки. Выходными параметрами являются: книги для продажи, бракованные книги. Управление: сопроводительная документация, карта размещения на складе. Механизмом являются: администратор, грузчик, кладовщик.

Четвертым блоком является "продажа". На входе: книги для продажи; данные для продажи; документ об оплате клиентом; данные об интересующем товаре. На выходе: чек клиенту, не востребованные книги, проданные книги, статистика продаж. Механизмом является кассир и продавец. Управление: сопроводительная документация, карта размещения на складе.

Последний блок "списание товара со склада". Входные данные: не востребованные книги. Выходные данные: списанные книги. Механизмом является администратор. Управление: сопроводительная документация, карта размещения на складе.

Разобравшись с функциональной моделью бизнес-процессов книжного магазина, можно переходить к разбору процесса обслуживания клиента.

1.2 Обслуживание клиентов книжного магазина

В данном разделе рассматривается процесс оказания услуг (справочная информация о наличии книги и где она расположена, прохождение акций или распродаж, или скидках), продавцом магазина потенциальному клиенту.

Данный процесс рассматривается на рисунке 1.4.

Из рисунка 1.4 видно, что при входе потенциальный клиент в магазин его должен дружелюбно поприветствовать продавец. После чего он устанавливает с ним контакт, и в дружелюбной форме предлагает помощь клиенту. Далее покупатель сообщает консультанту, что он желает приобрести. Сотрудник магазина представляет товар. Если товар отсутствует на прилавке, то сотрудник может посмотреть на компьютере нет ли такого товара на складе. В случае не нахождения товара отвечающему критериям покупателя продавец любезно сообщает об этом клиенту, если клиент уходит сотрудник должен разместить представленный товар на свои места, если клиент решил что-то приобрести, то продавец передает товар кассиру, который заканчивает обслуживание клиента [30].



Рисунок 1.4 – Обслуживание продавцом клиента книжного магазина

На рисунке 1.4 представлено как должно проходить обслуживание клиента, когда он обращается за помощью или консультацией к продавцу данного магазина.

Клиент может отказаться от помощи продавца и самостоятельно выбрать книгу из представленного ассортимента.

1.3 Математическая бизнес модель обслуживания клиентов

Чтобы разобраться в причинах затраты большого количества времени на обслуживание клиентов в книжном магазине, необходимо построить математическую модель. Для ее построения будет использована система Сети Петри.

В книге Харитоновой Е. В. Графы и сети, было сказано:

"Сети Петри используются главным образом для моделирования параллельных процессов: для моделирования компонентов компьютера,

параллельных вычислений, в робототехнике и даже для описания музыкальных структур. Сети Петри используют для нахождения дефектов в проекте системы, хотя имеют и многие другие применения. Они обладают многими свойствами блок-схем и конечных автоматов" [2].

Сетью Петри называется совокупность множеств $S=\{P, T, I, O\}$ где:

- P – конечное множество, элементы которого называются позициями;
- T – конечное множество, элементы которого называются позициями;
- I – множество входных функций;
- O – множество выходных функций [1].

Для расчета сколько уходит времени у клиента на приобретение продукции необходимо задать множество P [26]. В таблице 1.1 представлены это множество.

Таблица 1.1– Множество P для сети Петри

Элемент множества	Название позиции	Описание
P1	Начало	Покупатель входит в магазин
P2	Покупатель проходит в торговый зал	Покупатель решает найти товар без помощи найти товар
P3	Покупатель решает обратиться за помощью к продавцу	Требуется помощь продавца
P4	Покупателю оказана консультация	Покупатель потратил время на ожидания оказание консультации и получил ее
P5	Подобран товар	Самостоятельно либо с продавцом покупатель подобрал удовлетворяющий его запрос товар
P6	Покупка товара	Покупатель оплатил товар и забрал его

Теперь необходимо установить переходы, они будут не только измеряться временной шкалой, но и делиться на автоматизированные и нет [24]. В таблице 1.2 представлено множество Т переходов.

Таблица 1.2 – Множество Т переходов

Элемент	Тип выполнения	Время выполнения	Название перехода	Описание
T1	Обычный	1 минута	Покупатель проходит в торговый зал	При входе покупателя в торговый зал, его встречает продавец и предлагает помощь, клиент решает принять помощь, либо совершить поиск самостоятельно.
T2	Обычный	5-30 минут	Покупатель самостоятельно производит поиск	Покупатель производит поиск исходя из вывесок на полках с названием направления литературы
T3	Обычный	5 минут	Консультирование покупателя продавцом	Покупатель обращается за помощью к продавцу, говоря ему что он хочет. Продавец помогает клиенту в нахождение товара который его интересует
T4	Автоматизированный	1-2 минуты	Подбор книги продавцом с помощью поисковой системы	Продавец с помощью ПК магазина, производит поиск книге в БД
T5	Автоматизированный	1-2 минуты	Покупка товара	Сделав выбор из представленного товара покупатель оплачивает покупку и забирает ее

Далее все входящие множества и исходящие ребра нужно представить в матрице инцидентности [25]. В таблице 1.3 представлено множества I входящие в переходы ребер.

Таблица 1.3 – Множества I

	T1	T2	T3	T4	T5
P1	I1	0	0	0	0
P2	0	I2	0	0	0
P3	0	0	I3	0	0
P4	0	0	0	I4	0
P5	0	I5	0	0	I6
P6	0	0	0	0	0

Теперь в таблице 1.4 построим матрицу инцидентности множества O исходящие ребра из переходов.

Таблица 1.4 – Множества O

	P1	P2	P3	P4	P5	P6
T1	0	O1	O2	0	0	0
T2	0	0	O3	0	O4	0
T3	0	O5	0	O6	0	0
T4	0	0	0	0	O7	0
T5	0	0	0	0	0	O8

Исходя из данных представленных в таблицах можно сделать временную оценку бизнес-процессу по обслуживанию клиентов в книжном магазине. Данная оценка производится по формуле [16]:

$$T = t_1 + (1 - k) \sum_{i=1}^n t_2 + \sum_{j=1}^m t_3 + k \cdot t_4 + t_5 \quad (1.1)$$

где:

- n – среднее количество обращений за консультациями от покупателя;
- m – среднее количество консультаций по общим вопросам, без дальнейшего подбора книг;
- k – флаг использования поиска с помощью продавца, равен 0 если покупатель ищет книги сам, и 1 – если с помощью консультанта.

Покупатель может произвести самостоятельный поиск тогда $k=0$, или воспользоваться помощью консультанта тогда $k=1$. Предположим что клиент обращается за консультацией два раза, а задает общие вопросы три раза.

Расчеты при $k=1$ будут равны:

$$T = 1 + (1 - 1) \cdot 2 \cdot 17,5 + 3 \cdot 5 + 1 \cdot 1,5 + 1,5 = 19 \text{ минут} \quad (1.2)$$

При $k=0$ расчеты равны:

$$T = 1 + (1 - 0) \cdot 2 \cdot 17,5 + 3 \cdot 5 + 0 \cdot 1,5 + 1,5 = 52,5 \text{ минут} \quad (1.3)$$

Проанализировав результаты расчетов можно сделать следующие выводы:

- Если покупатель решает отказаться от помощи сотрудника, то он тратит в 2,7 раза больше своего времени;
- Больше всего времени уходит на неавтоматизированный процесс, такой как самостоятельный поиск товара;

– Время которое уходит на неавтоматизированный процесс составляет 97% от общего нахождения клиента в магазине.

Вывод по первому разделу

Рассмотрев первый раздел выпускной квалификационной работы можно сделать выводы: о том, как устроена работа в книжном магазине. Кто работает и за что отвечает. Проанализировав сценарий работы обслуживание клиента продавцом, и показателей расчета сети Петри, выявлены проблемы в не автоматизированном процессе работы подбора товара для потенциального клиента.

2 Описание оптимизации функциональной модели

В данном разделе речь пойдет об оптимизации бизнес-процесса обслуживания клиента в книжном магазине. На рисунке 2.1 представлена функциональная модель "как будет".

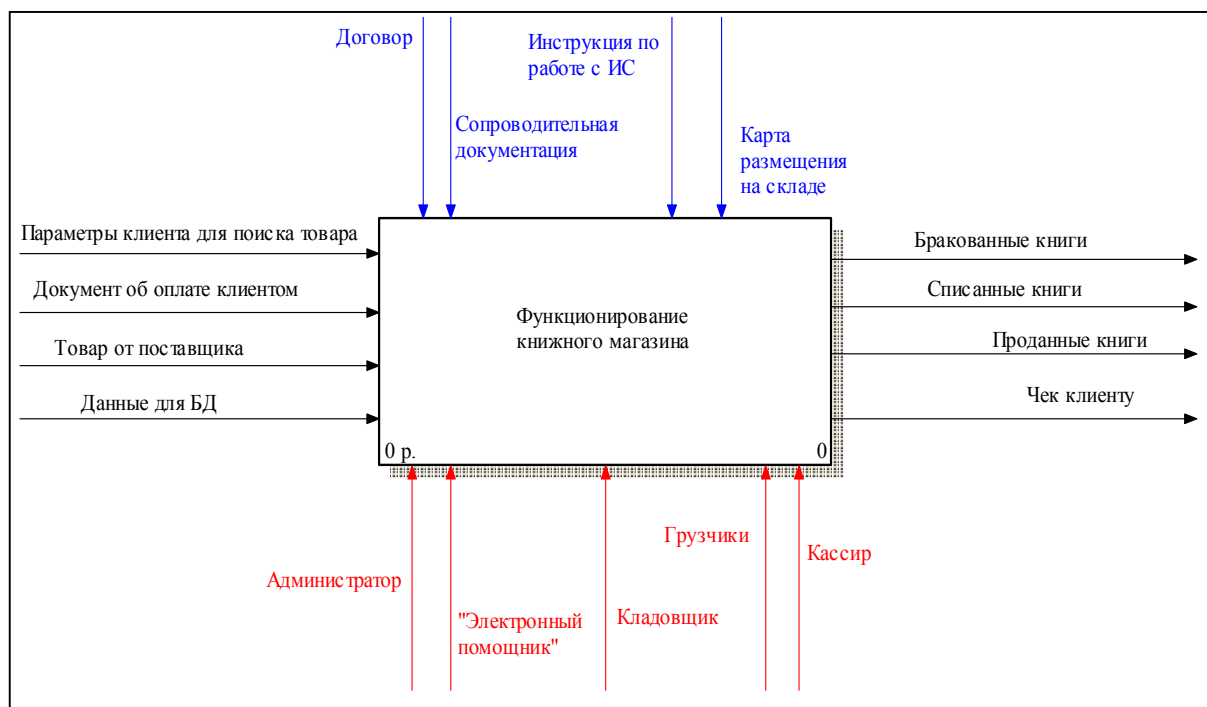


Рисунок 2.1 – Функциональная модель "как будет"

На рисунке 2.1 изображена функциональная модель функционирования "как будет". Входными параметрами являются: параметры клиента для поиска товара, документ об оплате клиентом, товар от поставщика, данные для БД. На выходе: бракованные книги, списанные книги, проданные книги, чек клиенту. Управление: договор, сопроводительная документация, инструкция по работе с ИС, карта размещения на складе. Механизм администратор, "Электронный помощник", кладовщик, грузчик, кассир.

На рисунке 2.1 представлен новый блок "поиска товара с помощью "Электронного помощника".

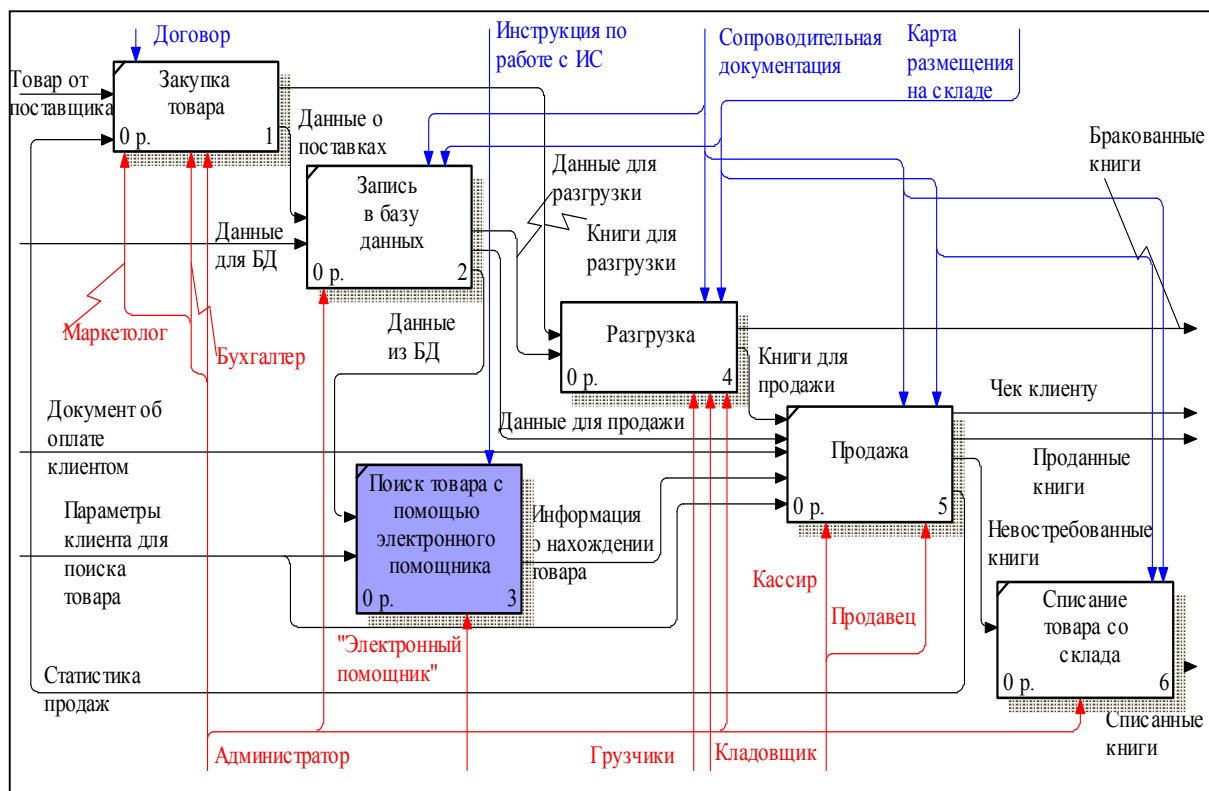


Рисунок 2.2 – Декомпозиция контекстной диаграммы "как будет"

На рисунке 2.2 входные параметры: параметры клиента для поиска товара, данные из БД. Выходные данные информация о нахождении товара. Управлением является инструкция по работе с ИС. Механизм: покупатель и обслуживающий персонал.

На рисунке 2.3 представлена декомпозиция "поиск товара с помощью электронного помощника". Декомпозиция состоит из трех блоков, которые отражают работу клиента с программой электронный помощник для книжного магазина.

На рисунке 2.3 изображены такие блоки: ввод параметров поиска, выполнение поиска товара, выбор товара удовлетворяющего условию покупателя. Для всех блоков единое управление: инструкция по работе с ИС, и единый механизм "Электронный помощник".

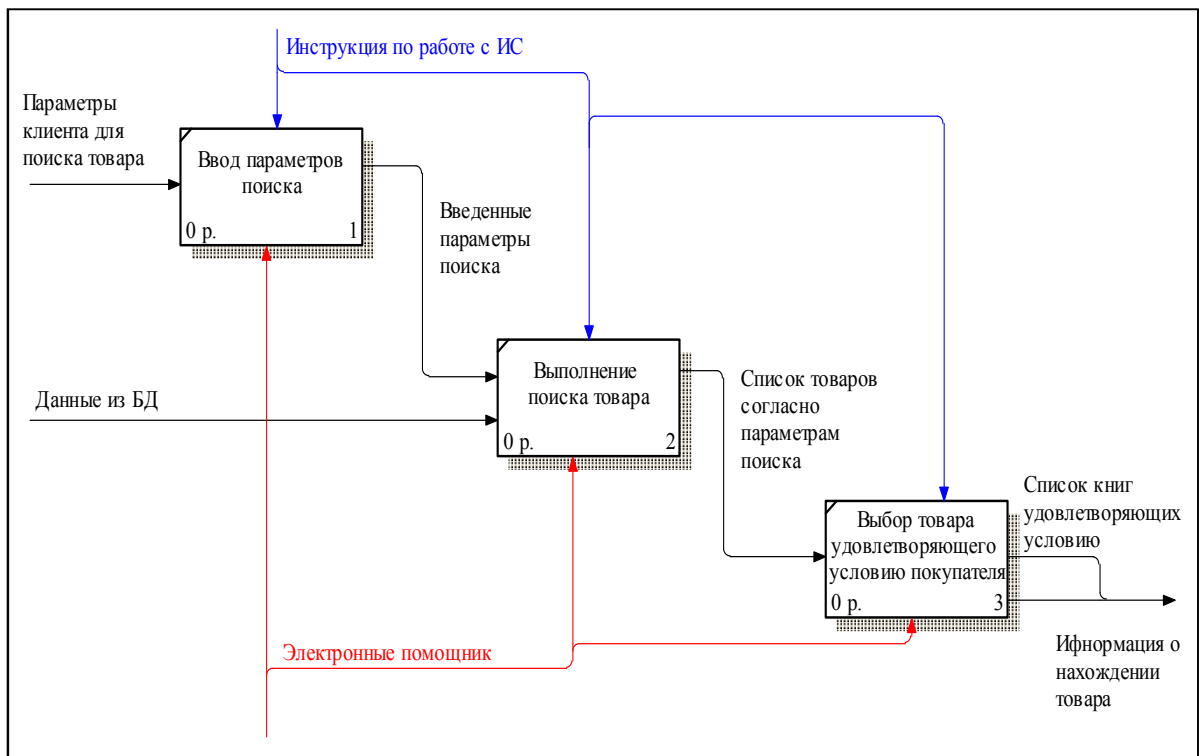


Рисунок 2.3 – Декомпозиция блока "поиск товара с помощью электронного помощника"

На рисунке 2.3 Входными данными для блока "ввод параметров поиска" является "параметры клиента для поиска товара", а на выходе "введенные параметры поиска", которые служат входными параметрами для блока "выполнение поиска товара". Так же для этого блока "данные из БД" являются входными параметрами, а на выходе "список товаров согласно параметрам поиска". Заключительным блоком является "выбор товара удовлетворяющего условию покупателя". На выходе стрелка "список книг удовлетворяющих условию", которая сливается со стрелкой "информация о нахождении товара".

Перед проектированием автоматизированной системы электронного помощника, нужно убедиться что данная система уменьшит время подбора товара для клиента, для этого воспользуемся системой Петри.

2.1 Оптимизация математической модели подбора товара

В первом разделе расчеты показали, что покупатель тратит на самостоятельный поиск пятьдесят две минуты, это время надо уменьшить примерно в 1,5 раза.

Для начала проведем оценку процентного соотношения для каждой выполняемой операции на основании полученного ранее времени, и модифицируем сеть Петри таким образом, чтобы время, затрачиваемое на поиск, уменьшилось. В таблице 2.1 представлены результаты оценки процентов времени.

Таблица 2.1 – Результаты временных затрат

Элемент сети Петри	Тип выполнения	Среднее время выполнения	Процент от общего времени
T1	Обычный	1 минута	1,9%
T2	Обычный	5-30 минут (2 повторения)	66,7%
T3	Обычный	5 минут (3 повторения)	25,8%
T4	Автоматизированный	1-2 минуты (0 повторений)	0%
T5	Автоматизированный	1-2 минуты	2,9%

Исходя из данных в таблице 2.1, можно сделать вывод, что больше всего покупатель тратит время на самостоятельный поиск значит, его и стоит модифицировать. Чтобы это сделать нужно самостоятельный поиск разделить на две части: ознакомление с представленными товарами и самостоятельный поиск. Как бы не был автоматизирован процесс поиска, покупатель, заходя в магазин, окидывает взглядом витрины и только после этого приступает к поиску, этот процесс происходит примерно 5 минут. Учитывая этот факт, задаем новые значения для сети Петри. В таблице 2.2 представлены новые значения множества Р позиций.

Таблица 2.2 – Множества Р для модифицированной сети Петри

Элемент множества	Название позиции	Описание
P1	Начало	Покупатель входит в магазин
P2.1	Покупатель проходит в торговый зал	Покупатель решает найти товар без помощи найти товар
P2.2	Покупатель ищет книгу	Покупатель производит самостоятельный поиск
P3	Покупатель решает обратиться за помощью к продавцу	Требуется помощь продавца
P4	Покупателю оказана консультация	Покупатель потратил время на ожидания оказание консультации и получил ее
P5	Подобран товар	Самостоятельно либо с продавцом покупатель подобрал удовлетворяющий его запрос товар
P6	Покупка товара	Покупатель оплатил товар и забрал его

В таблице 2.3 представлены множества Т переходов.

Таблица 2.3 – Модифицированные переходы множества Т

Элемент	Тип выполнения	Время выполнения	Название перехода	Описание
T1	Обычный	1 минута	Покупатель проходит в торговый зал	При входе покупателя в торговый зал, его приветствует продавец и предлагает помощь, клиент решает принять помощь, либо совершить поиск самостоятельно.

Продолжение таблицы 2.3

T2.1	Обычный	5-30 минут	Покупатель изучает	Покупатель ходит по магазину изучая
------	---------	------------	--------------------	-------------------------------------

			ассортимент	ассортимент, но не производит поиск
T2.2	?	?	Покупатель самостоятель но производит поиск	Покупатель производит поиск исходя из вывесок на полках с названием направления литературы
T3	Обычный	5 минут	Консультирование покупателя продавцом	Покупатель обращается за помощью к продавцу, говоря ему что он хочет. Продавец помогает клиенту в нахождение товара который его интересует
T4	Автоматизированный	1-2 минуты	Подбор книги продавцом с помощью поисковой системы	Продавец с помощью ПК магазина, производит поиск книге в БД
T5	Автоматизированный	1-2 минуты	Покупка товара	Выбрав из представленного ассортимента покупатель оплачивает товар и забирает его

Множества из Р и Т из таблиц 2.2, 2.3 строятся матрица инцидентности, задаются такие матрицы для описание переходов и состояний, которые потребуются для расчетов по формуле, которая была описана в первой главе. В таблице 2.4 представлены инцидентности для множества I переходов.

Таблица 2.4 – Множества I входящих в переходы ребра инцидентности

	T1	T2.1	T2.2	T3	T4	T5
--	----	------	------	----	----	----

P1	I1	0	0	0	0	0
P2.1	0	I2	0	0	0	0
P2.2	0	0	I3	0	0	0
P3	0	0	0	I4	0	0
P4	0	0	0	0	I5	0
P5	0	I5	I5	0	0	I7
P6	0	0	0	0	0	0

В таблице 2.5 представлена матрица инцидентности для множества исходящих O переходов.

Таблица 2.5 – Множества O исходящих переходов

	P1	P2.1	P2.2	P3	P4	P5	P6
T1	0	O1	0	O2	0	0	0
T2.1	0	0	O3	0	0	0	0
T2.2	0	0	0	0	0	O4	0
T3	0	O5	O5	0	O6	0	0
T4	0	0	0	0	0	O7	0
T5	0	0	0	0	0	0	O8

Проанализировав данные из представленных таблиц можно произвести расчеты по формуле [16]:

$$T = t_1 + (1 - k) \sum_{i=1}^n t_{2,i} + t_{2,2} + \sum_{j=1}^m t_3 + k t_4 + t_5 \quad (2.1)$$

Пусть $k=0$, тогда расчеты будут равны:

$$T = 1 + (1 - 0) \cdot 2 \cdot 3 + 3 \cdot 5 + 1,5 + t_{2,2} = 23,5 + t_{2,2} \quad (2.2)$$

Так как временная оценка для процесса $t_{2.2}$ неизвестна, однако известно, то что должно выполняться неравенство $23,5 + t_{2.2} \leq 28,5$ или $t_{2.2} \leq 5$, эти неравенства учитываются при самостоятельном поиске.

Из предыдущих данных можно сделать следующий вывод: улучшив математическую модель сети Петри можно выполнять самостоятельный поиск за двадцать восемь с половиной минут, при учете не превышения времени на поиска книги не более пяти минут.

2.2 Описание подбора товара с использованием электронного помощника

При использования электронного помощника процесс нахождения товара будет проходить как представлено на рисунке 2.4. Покупатель заходя в магазин имеет какое-то представление об интересующем его товаре, а так же знает минимальную информацию о книге или знает имя автора, который написал интересующую покупателя книгу. Эта информация вводится в поисковую строку, так же покупатель может указать стоимость товара, на которую он рассчитывает потратить на заинтересовавший его товар. Если данная информация содержится в базе данных магазина, то покупатель получает сведения о нахождение данного товара, а так же его стоимость и краткое описание. Если покупатель знает данные, которые объединяют несколько товаров, то будет предложен список, из которого можно выбрать что подходит больше всего или полностью удовлетворяет запросу покупателя. После использования приложения "электронный помощник", клиент направляется к месту нахождения товара, которое указала программа.



Рисунок 2.1 – Процесс нахождения товара

На рисунке 2.4 представлен алгоритм, который описывает процесс подбора товара покупателем самостоятельно с использованием электронного помощника.

2.3 Анализ среды для разработки базы данных

Для того чтобы система эффективно выполняла свои функции, необходимо провести комплексную оценку каждого из наиболее подходящих вариантов решения и выбрать наиболее оптимальный вариант.

Решение должно соответствовать таким критериям:

- быстродействие;
- простота;
- удобство;
- функциональность;
- совместимость.

2.3.1 Выбор среды проектирование базы данных

Базы данных – это инструмент для сбора и структурирования информации. В базе могут храниться данные о людях, товарах, заказах и о многом другом. Многие базы данных изначально представляют собой небольшой список в текстовом редакторе или электронной таблице [3].

Стержневые идеи современных информационных технологий базируются на концепции баз данных.

В настоящее время основой информационной технологии являются данные, для адекватного отображения изменяющегося реального мира и удовлетворения информационных потребностей пользователя должны быть организованы в базы данных.

Информация является одним из важнейших понятий. Информация – это любые сведения о каком-либо событии, процессов, объекте.

Данные – это информация, представленная в определенном виде, позволяющем автоматизировать её сбор, хранение и дальнейшую обработку человеком или информационным средством [3].

База данных (БД) – это именованная совокупность данных, которая отражает состояние объектов, который относятся к рассматриваемой области [5].

Информация в базах данных делится на таблицы. Столбцы и строки образуют таблицу. Столбец – это фрагмент данных, хранящихся в таблице. Строка (или запись) – это набор столбцов, описывающих атрибуты одного объекта [4].

Система управления базами данных (СУБД) – программные средства, предназначенные для создания, введения, изменения и совместного использования БД несколькими пользователями.

Сервер БД позволяет проводить обслуживание и управления базой данных и отвечать за целостность данных, а так же обеспечивает операции ввода-вывода при доступе клиента к информации.

В данной работе используется база данных Microsoft Office Access 2007. В базе данных Access все таблицы хранятся в одном файле вместе с другими объектами, такими как формы, отчеты, макросы и модули. Для файлов баз данных, созданных в формате Access 2007 (который также используется в Access 2016, Access 2013 и Access 2010), используется расширение ACCDB, а для баз данных, созданных в более ранних версиях Access, – MDB. С помощью Access 2016, Access 2013, Access 2010 и Access 2007 можно создавать файлы в форматах более ранних версий приложения (например, Access 2000 и Access 2002–2003) [10].

Использование Access позволяет:

- добавлять новую информацию в базу данных, например новый артикул складских запасов;
- изменять информацию, уже находящуюся в базе, например перемещать артикул;
- удалять информацию, например если артикул был продан или утилизирован;

- упорядочивать и просматривать данные различными способами;

- обмениваться данными с другими людьми с помощью отчетов, сообщений электронной почты, внутренней сети или Интернета.

Компонентами базы данных являются:

- таблица
- форма
- отчеты
- запросы
- макросы
- модули.

Microsoft Access, обладая всеми чертами классической СУБД, предоставляет и дополнительные возможности. Access – это не только мощная, гибкая и простая в использовании СУБД, но и система для разработки, работающих с базами данных, приложений. С помощью Access можно создать приложение, работающее в среде Windows и полностью соответствующее потребностям по управлению данными. Используя запросы, есть возможность выбирать и обрабатывать хранящуюся в таблицах информацию. Можно создавать формы для ввода, просмотра и обновления данных, а также использовать Access для создания как простых, так и сложных отчетов. Формы и отчеты «наследуют» свойства базовой таблицы или запроса, так что в большинстве случаев необходимо указать форматы, условия на значения и некоторые другие характеристики данных только один раз. К числу наиболее мощных средств Access относятся средства разработки объектов – Мастера, которые можно использовать для создания таблиц, запросов различных типов форм и отчетов, просто выбрав с помощью мыши нужные опции.

Microsoft Access имеет множество областей применения: на малом предприятии; корпорации и даже в сфере домашнего применения.

2.3.2 Разработка базы данных для книжного магазина

При использовании книжным магазином электронного помощника, данная программа тянет информацию о расположении книг в торговом зале. Поэтому без проектирования БД книжного магазина не будет работать «Электронный помощник». База данных разработана в программе Microsoft Access. В программе были созданы таблицы. После создания, необходимо определить связи между таблицами. На рисунке 2.5 представлены все таблицы и связи между ними.

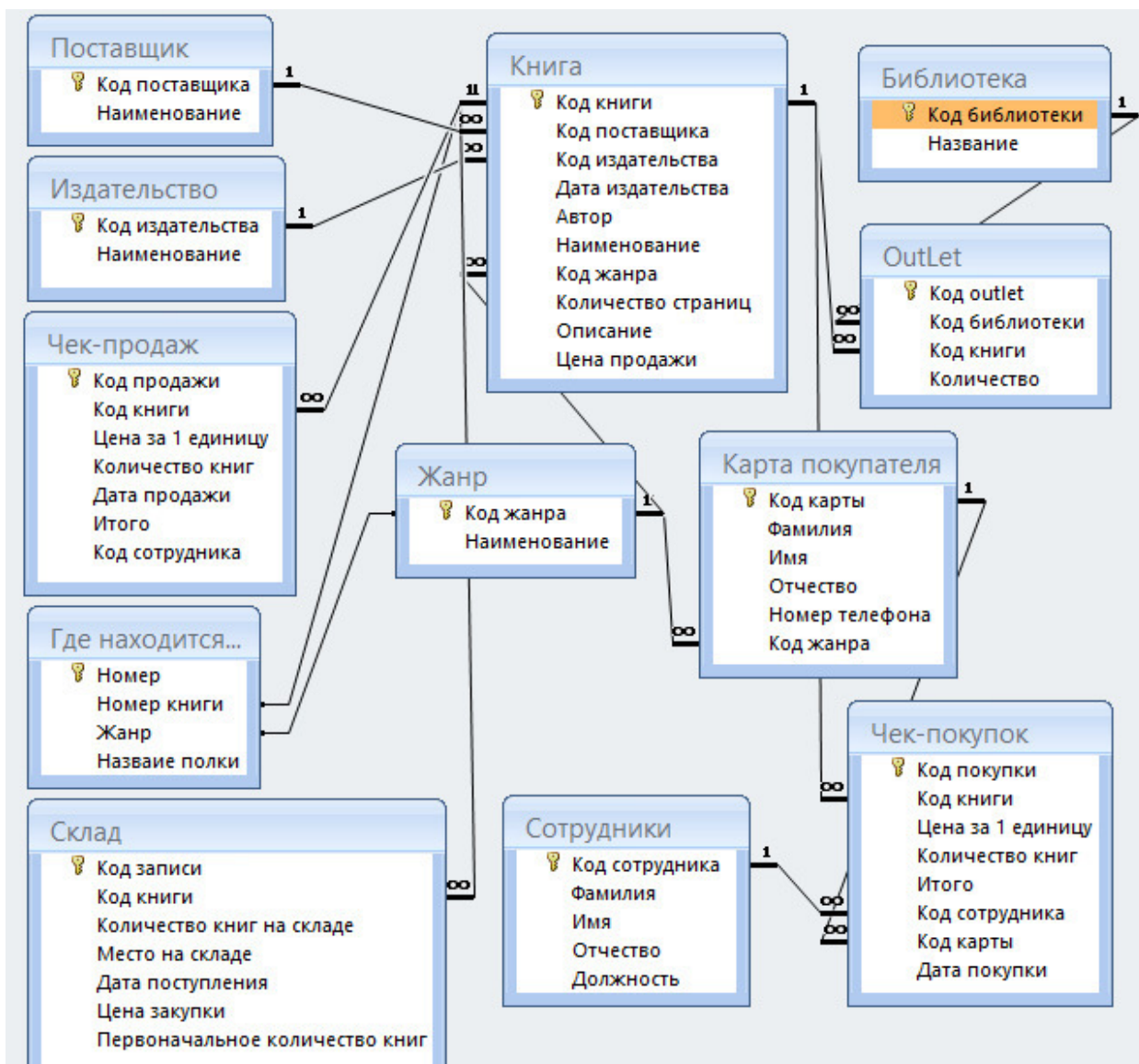


Рисунок 2.5 – Схема данных

Схема данных представленная на рисунке 2.2 подробно рассмотрена каждая таблица и описана ниже.

Поставщик: имеет связь с таблицей "Книга". Данная таблица предназначена для хранения данных о поставщиках с которыми сотрудничает книжный магазин. В данную таблицу входит два поля:

- a) Код поставщика – счетчик;
- b) Наименование – текстовой.

Издательство: имеет связь с таблицей "Книга". Данная таблица предназначена для хранения данных об издателях, который публикуют книги, которые будут продаваться в книжном магазине . В данную таблицу входит два поля:

- a) Код издательства – счетчик;
- b) Наименование – текстовой.

Склад: Имеет связь с таблицей "Книга". Таблица предназначена для отслеживание количества книг для продажи, закупочная цена, а также их размещение. В данную таблицу входят поля:

- a) Код склада – счетчик;
- b) Код книги – числовой;
- c) Количество книг на складе – числовой;
- d) Место на складе – текстовой;
- e) Дата поступление – дата/время;
- f) Цена закупки – денежный;
- g) Первоначальное количество книг – числовой.

Чек-продаж: Имеет связи с таблицами "Книга" и "Сотрудники". Таблица предназначена для создание чека для магазина, это позволяет отслеживать продажи. Имеет такие поля:

- a) Код продажи – счетчик;
- b) Код книги – числовой;

- с) Цена за 1 единицу – денежный;
- д) Количество книг – числовой;
- е) Дата продажи – дата/время;
- ф) Итого – денежный;
- г) Код сотрудника – числовой.

Сотрудники: имеет связь с таблицами "Чек-продаж" и "Чек-покупок". Предназначена для хранения данных о сотрудниках, а также для отображения информации и сотруднике в чеках. Имеет поля:

- а) Код сотрудника – счетчик;
- б) Фамилия – текстовой;
- с) Имя – текстовой;
- д) Отчество – текстовой;
- е) Должность – текстовой.

Outlet: связи с таблицами "Книга" и "Библиотека". Таблица предназначена для списания не востребованных книг в библиотеке с которыми сотрудничает магазин. В неё входят поля:

- а) Код outlet – счетчик;
- б) Код библиотеки – числовой;
- с) Код книги – числовой;
- д) Количество – числовой.

Библиотека: связь с таблицей "Outlet". Предназначена для хранения данных о библиотеках. Содержит следующие поля:

- а) Код библиотеки – счетчик;
- б) Название – текстовой.

Жанр: связи с таблицами "Книга" и "Карта покупателя". Предназначена для хранения данных о жанрах продаваемых книг. Имеет поля:

- а) Код жанра – счетчик;
- б) Наименование – текстовой.

Карта покупателя: связи с таблицами "Жанр" и "Чек-продажи". Предназначена для хранения данных о покупателях у которых есть

скидочная дисконтная карта, а также для рассылки о новых поступлениях книг с учетом выбранного жанра покупателем. Содержит такие поля:

- a) Код карты – счетчик;
- b) Фамилия – текстовой;
- c) Имя – текстовой;
- d) Отчество – текстовой;
- e) Номер телефона – числовой;
- f) Код жанра – числовой.

Чек-покупка: связи с "Карта покупателя", "Книга" и "Сотрудники".

Хранит данные о покупке для покупателя. Имеет поля:

- a) Код покупки – счетчик;
- b) Код книги – числовой;
- c) Цена за 1 единицу – денежный;
- d) Дата покупки – дата/время;
- e) Количество книг – числовой;
- f) Итого – денежный;
- g) Код сотрудника – числовой;
- h) Код карты – числовой.

Книга: имеет связи с таблицами "Склад", "Поставщик", "Издательство", "Чек-продаж", "Чек-покупка", "Жанр" и "Outlet". Предназначена для хранения информации о книгах продаваемых в книжном магазине. Содержит поля:

- a) Код книги – счетчик;
- b) Код поставщика – числовой;
- c) Код издательства – числовой;
- d) Дата издательства – дата/время;
- e) Автор – текстовой;
- f) Наименование – текстовой;
- g) Код жанра – числовой;
- h) Количество страниц – числовой;

- i) Описание – текстовой;
- j) Цена продажи – денежный;
- k) Код склада – числовой.

После создание базы данных необходимо её заполнить, далее можно переходить к созданию приложения.

Выводы по второму разделу

Проанализировав второй раздел, можно сделать вывод: что при автоматизации процесса самостоятельного поиска интересующего товара покупателя, время на поиск уменьшится в 1.5 раз, а так же это позволит уменьшить потерю потенциальных клиентов. Что в свою очередь повысит доход книжного магазина. Так же в данном разделе разработана база данных, без которой не возможно существование приложения.

3 Проектирование программы

В данном разделе находятся информация о выборе аппаратно-программной платформе, о выборе языка программирования и выбор среды разработки.

3.1 Выбор аппаратно-программной платформы

На сегодняшний день, в развитое информационное общество, существует большой выбор аппаратно-программных платформ. Так же следует отметить, что выбор какой либо платформы и конфигурации непосредственно связано с рядом требований, который предъявляются к характеристикам современных вычислительных систем, такие как:

- Отношение стоимость к производительности;
- надежность и отказоустойчивость;
- масштабируемость;
- совместимость и мобильность ПО [28].

Стоит разобрать что же определяет каждый пункт, первое – это отношение стоимости и производительности. Когда на рынке появляется кокой либо новый электронный продукт к нему предъявляется ряд требований. У разработчиков несколько целей при создании новой техники, так как большая универсальная вычислительная система или суперкомпьютер стоят достаточно дорого, производителям приходится игнорировать высоко производительность либо конструкцией. В этих крайностях и находятся техника основанная на отношении "стоимость и производительность", где производителям нужно соблюдать баланс между стоимостью комплектующих и производительностью. Пример таких разработок это миникомпьютеры или рабочие станции.

Надежность и отказоустойчивость являются важными характеристиками вычислительных систем. Повышение надежности основано на принципе предотвращения неисправностей путем снижения интенсивности отказов и сбоев за счет применения электронных схем и компонентов с высокой и сверхвысокой степенью интеграции, снижения уровня помех, облегченных режимов работы схем, обеспечение тепловых режимов их работы, а также за счет совершенствования методов сборки аппаратуры. Отказоустойчивость – это свойство вычислительной системы, которое обеспечивает ей, как логической машине, возможность продолжения действий, заданных программой, после возникновения неисправностей. Введение отказоустойчивости требует избыточного аппаратного и программного обеспечения. Направления, связанные с предотвращением неисправностей и с отказоустойчивостью, – основные в проблеме надежности. Концепции параллельности и отказоустойчивости вычислительных систем естественным образом связаны между собой, поскольку в обоих случаях требуются дополнительные функциональные компоненты. Поэтому на параллельных вычислительных системах достигается как наиболее высокая производительность, так и, во многих случаях, очень высокая надежность. Имеющиеся ресурсы избыточности в параллельных системах могут гибко использоваться как для повышения производительности, так и для повышения надежности. Структура многопроцессорных и многомашинных систем приспособлена к автоматической реконфигурации и обеспечивает возможность продолжения работы системы после возникновения неисправностей. Следует помнить, что понятие надежности включает не только аппаратные средства, но и ПО. Главной целью повышения надежности систем является целостность хранимых в них данных.

Масштабируемость – это возможность наращивания мощность процессора, объем оперативной и внешней памяти, а также остальных ресурсов вычислительной системы. Все это должно обеспечиваться

конструкцией и архитектурой ВС, а так же ПО. При добавлении нового процесса в масштабируемую систему должно соблюдаться увеличение производительности и пропускная способность при правильных затрат. Одна из главных задач при построении таких систем – это уменьшение стоимости расширения компьютеров. В лучшем случае при добавление процессоров должна в линейном росте увеличиваться производительность, но это не всегда так. Потеря производительности возникает при плохой пропускной способности шин. Масштабируемость программного обеспечения касается всех его уровней, так же ПО должно минимизировать трафик межпроцессорного обмена, который может препятствовать линейному росту производительности системы. В правильно масштабируемой системы должен быть баланс между все конструкцией [27].

Совместимость и мобильность ПО – основная задача при проектировании всего программного обеспечения это создать такое устройство, которое не смотря на отличие цены и производительности была одинакова с точки зрения пользователей.

3.2 Выбор языка программирования

В данной выпускной квалификационной работе был выбран язык программирования C#.

В книге Герберт Шилдт "C# 4.0 полное руководство" было сказано: "C# – это современный и прогрессивный язык программирования, который включает возможности, доступные в наиболее распространенных промышленных и исследовательских языках" [9].

Синтаксис C# очень богат, но при этом прост и удобен в изучении. Характерные фигурные скобки C# мгновенно узнаются всеми, кто знаком с C, C++ или Java. Разработчики, знающие любой из этих языков, обычно очень быстро начинают эффективно работать в C#. Синтаксис C# упрощает многие сложности C++, но при этом предоставляет отсутствующие в Java

мощные функции, например обнуляемые типы значений, перечисления, делегаты, лямбда-выражения и прямой доступ к памяти. С# поддерживает универсальные методы и типы, которые обеспечивают более высокий уровень безопасности и производительности, а также итераторы, позволяющие определять в классах коллекций собственное поведение итерации, которое может легко применить в клиентском коде. Выражения LINQ создают очень удобную языковую конструкцию для строго типизированных запросов [17].

С# является объектно-ориентированным языком, а значит поддерживает инкапсуляцию, наследование и полиморфизм. Все переменные и методы, включая метод Main, представляющий собой точку входа в приложение, инкапсулируются в определения классов. Класс наследуется непосредственно из одного родительского класса, но может реализовывать любое число интерфейсов. Методы, которые переопределяют виртуальные методы родительского класса, должны содержать ключевое слово `override`, чтобы исключить случайное переопределение. В языке С# структура похожа на облегченный класс: это тип, распределяемый в стеке, реализующий интерфейсы, но не поддерживающий наследование [20].

Помимо этих основных принципов объектно-ориентированного программирования, С# предлагает ряд инновационных языковых конструкций, упрощающих разработку программных компонентов [11]:

- инкапсулированные сигнатуры методов, именуемые *делегатами*, которые позволяют реализовать якобы безопасные уведомления о событиях;
- свойства, выполняющие функцию акцессоров для закрытых переменных-членов;
- атрибуты, предоставляющие декларативные метаданные о типах во время выполнения;
- внутри строчные комментарии для XML-документации;

- LINQ для создания запросов к различным источникам данных.

Для взаимодействия с другим программным обеспечением Windows, например с объектом COM или собственными библиотеками DLL Win32, можно применить процесс C#, известный как "Взаимодействие". Взаимодействие позволяет программам на C# делать практически все, что возможно в приложении машинного кода C++. C# поддерживает даже указатели и понятие "небезопасного" кода для тех случаев, в которых критически важен прямой доступ к памяти [12].

Процесс построения в C# проще по сравнению с C или C++, но более гибок, чем в Java. Отдельные файлы заголовка не используются, и нет необходимости объявлять методы и типы в определенном порядке. Исходный файл C# может определить любое число классов, структур, интерфейсов и событий [18].

3.3 Выбор среды программирования

В настоящее время существует несколько сопутствующих технологий для разработки на языке программирования C#, такие как:

- LINQ (Language Integrated Query) – язык интегрированных запросов, напоминающий SQL, в языках для платформы .NET;
- Microsoft Visual Studio – один из самых известных продуктов от Microsoft. Среда разработки, которая позволяет писать код на C#, C++ и некоторых других (при установке плагинов). С недавнего времени вышла версия VS Community для некоммерческого использования и небольших команд разработчиков [21];
- ASP.NET – технология для веб-приложений (и сервисов), продолжение старой технологии ASP (Active Server Pages);

- TPL (Task Parallel Library) – библиотека для создания многопоточных приложений;
- WCF(Windows Communication Foundation) – среда выполнения и набор API-интерфейсов для создания систем, которые обеспечивают обмен сообщениями между службами и клиентами;
- WPF (Windows Presentation Foundation) – система для построения приложений с визуальным интерфейсом для Windows, позволяет добавлять двухмерную трехмерную графику, анимацию и т.д;
- Resharper – плагин для Visual Studio, который позволяет обеспечивает дополнительные функции при написании кода - анализ и подсветка ошибок и опечаток, улучшенную навигацию по коду и статический анализ кода (без выполнения программы);
- F# – язык для функционального программирования под .NET [13].

Проанализировав все предоставляемые среды разработки была выбрана Microsoft Visual Studio так как имеет ряд преимуществ.

Это интегрированная среда разработки предлагает ряд высокоуровневых функциональных возможностей, которые выходят за рамки базового управления кодом.

Visual Studio предоставляет возможность написания кода на нескольких языках используя один и тот же интерфейс, а также позволяет создавать Web-страницы на различных языках при этом объединяя их в одно приложение [22].

Интуитивный стиль кодирования. По умолчанию Visual Studio форматирует код по мере его ввода, автоматически вставляя необходимые отступы и применяя цветовое кодирование для выделения элементов типа комментариев. Такие незначительные отличия делают код более удобным для чтения и менее подверженным ошибкам. Применяемые Visual Studio автоматически параметры форматирования можно даже настраивать, что

очень удобно в случаях, когда разработчик предпочитает другой стиль размещения скобок [14].

В среде Visual Studio предоставляется высокая скорость разработки и многие функциональные возможности направлены на помощь разработчику при выполнении работы. Разработаны функции перехвата ошибок, поиск и замена, добавление и удаление комментариев, все эти функции позволяют затрачивать меньше времени на разработку.

Так же есть инструменты отладки, которые помогают найти ошибки и неправильное выполнение программного кода. Программист может просмотреть изменения в программе построчно, что позволяет определить в какой момент выполнения программы произошла ошибка. А также сохранять режим отладки и просматривать данные из памяти.

Visual Studio также имеет и множество других функций: возможность управления проектом; встроенная функция управления исходным кодом; возможность рефакторизации кода; мощная модель расширяемости [14].

3.4 Разработка программы

В данном разделе представлен интерфейс программы "Электронный помощник" для книжного магазина.

На рисунке 3.1 изображена главная форма программы. На рисунке 3.1 можно увидеть главное меню программы, в состав которого входят, такие разделы как: поиск книг, бестселлеры, календари акций, отзывы и предложения.

При нажатии кнопки клиент переходит на следующие формы. На рисунке 3.2 представлена форма поиска интересующей книги, а на рисунке 3.3 изображена форма с бестселлерами.

Кнопка "Календарь акций", позволяет потенциальному клиенту перейти на форму с календарем, в котором представлена информация об акциях или скидках, которые будут проводиться в текущем месяце.

Кнопка "Отзывы и предложения", переходит на форму, где потенциальный клиент может оставить свои как положительный, так и отрицательный отзыв о программе. Так же пользователь может написать, что не хватает "Электронному помощнику" с их точки зрения.

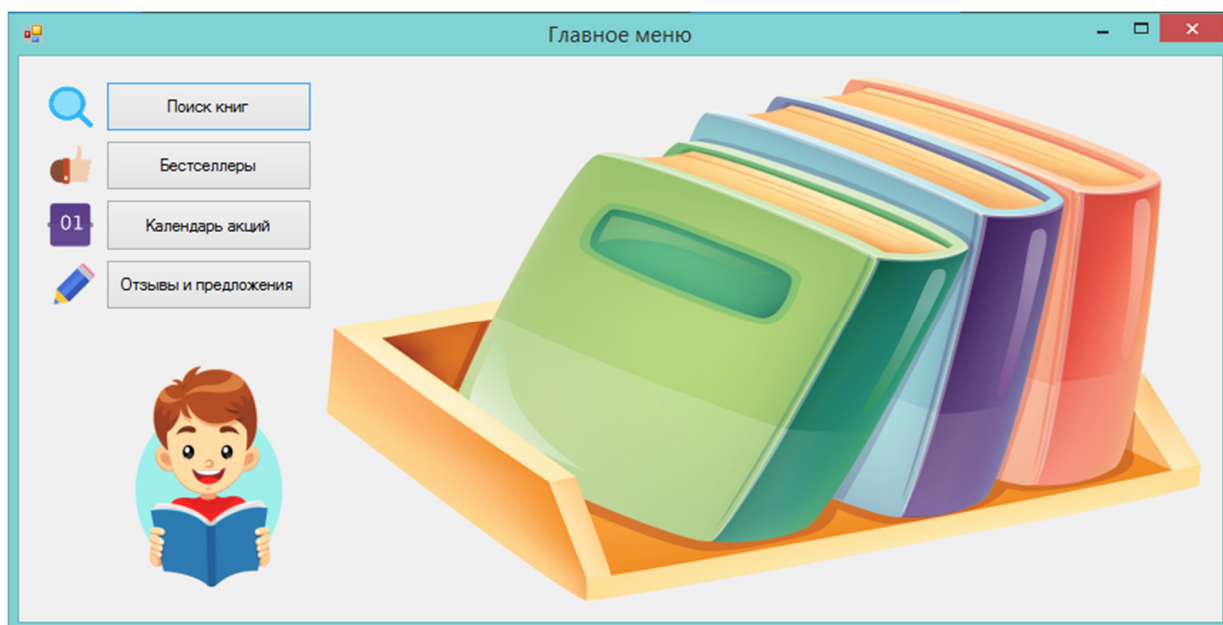


Рисунок 3.1 – Главное меню

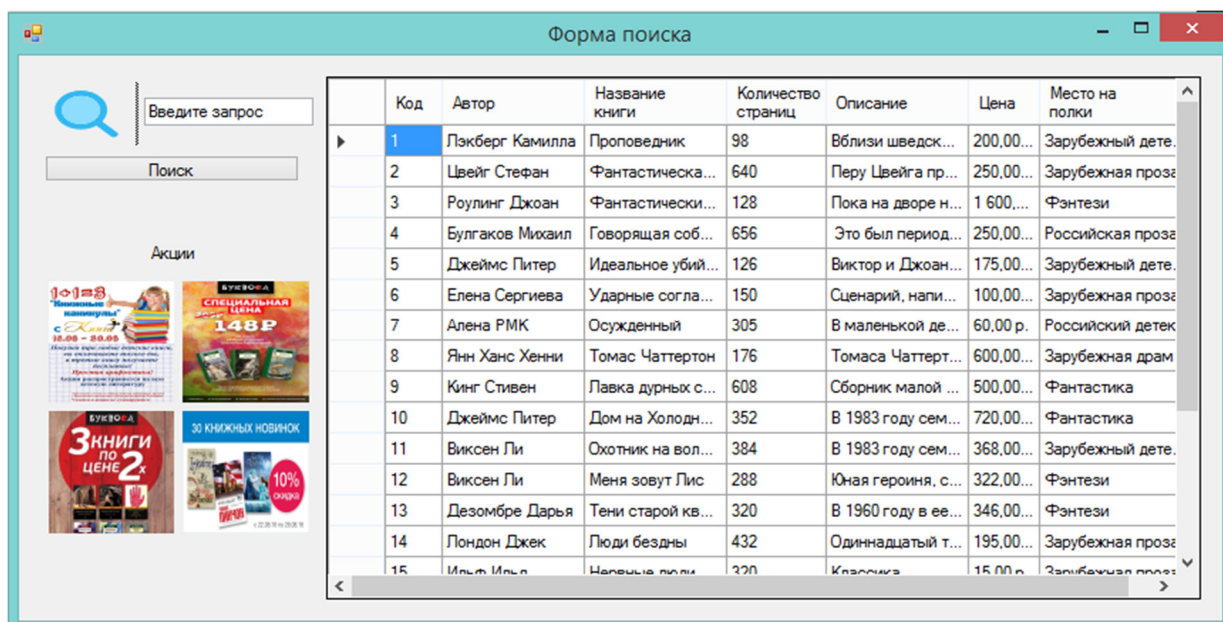


Рисунок 3.2 – Форма поиска товара

На рисунке 3.2 форма поиска интересующей клиента литературы. Чтобы осуществить поиск, потенциальный покупатель должен ввести информацию о товаре, которой он располагает. Например: имя автора, название книги, количество страниц, часть описания книги, цена товара. После ввода информации для поиска, программа оставит данные только о книгах, которые удовлетворяют поиску клиента, и покажет на какой полке искать данный товар.

The screenshot shows a window titled 'Бестселлеры' (Bestsellers) with three columns of data. Each column has a header indicating the time period: 'На этой неделе' (This week), 'В этом месяце' (This month), and 'В этом году' (This year). Each column contains a table with 10 rows of book data, including columns for 'Код' (Code), 'Название книги' (Book title), 'Цена' (Price), and 'Количество проданных' (Quantity sold).

На этой неделе				В этом месяце				В этом году			
Код	Название книги	Цена	Количество проданных	Код	Название книги	Цена	Количество проданных	Код	Название книги	Цена	Количество проданных
1	Лавка дурн...	500...	23	1	Фантастиче...	1 60...	27	1	Фантастиче...	1 60...	53
2	Охотник на ...	368...	20	2	Охотник на ...	368...	26	2	История Лизи	655...	44
3	История Лизи	655...	19	3	История Лизи	655...	22	3	Меня зовут ...	322...	40
4	Фантастиче...	1 60...	15	4	Лавка дурн...	500...	20	4	Охотник на ...	368...	37
5	Тени старой...	346...	14	5	Меня зовут ...	322...	17	5	Дом на Хол...	720...	32
6	Идеальное ...	175...	10	6	Фантастиче...	250...	16	6	Лавка дурн...	500...	27
7	Дом на Хол...	720...	8	7	Дом на Хол...	720...	14	7	Фантастиче...	250...	24
8	Меня зовут ...	322...	7	8	Программи...	556...	3	8	Идеальное ...	175...	20
9	Фантастиче...	250...	3	9	Идеальное ...	175...	2	9	Тени старой...	346...	15
10	Программи...	556...	1	10	Тени старой...	346...	1	10	Программи...	556...	5

Рисунок 3.3 – Форма с бестселлерами

На рисунке 3.3 представлена форма с бестселлерами, которая разделена на три раздела. В первом представлен список топ-10 продаваемых книг на текущей недели. Во второй части, топ-10 продаваемых в текущем месяце, и в третьей, топ-10 продаваемых за текущей год. Таким образом если покупатель не смог найти интересующего его товара, он может подобрать себе что-то из списка бестселлера, положившись на выбор покупателей.

Выводы по второму разделу

Вывод: в данном разделе были рассмотрены и выбраны наиболее подходящие аппаратно-программная платформа, язык программирования и среда программирования. Разработано и протестировано приложение информационной системы для интерактивного книжного терминала. Был представлен графический интерфейс приложения с описанием бизнес логики каждого модуля программы. Были описаны функциональные возможности приложения и дана краткая инструкция по его использованию. Разработанное приложение позволяет уменьшить время покупателя для самостоятельного поиска, исходный код которой представлен в приложение А.

ЗАКЛЮЧЕНИЕ

В ходе выпускной квалификационной работы изучена производственная структура книжного магазина. Выявлен недостаток в ходе подбора товара продавцом для потенциального покупателя.

В работе была улучшена математическая модель сети Петри, что позволит тратить клиентам меньше своего времени при самостоятельном поиске, не обращаясь за помощью к продавцам данного магазина.

В ходе выполнения дипломной работы решены следующие задачи:

- изучена предметная область;
- спроектирована информационная система;
- выбрана аппаратно-программная платформа, язык программирования и среда разработки;
- разработана информационная система для интерактивного книжного терминала.

Достигнута цель работы: повышение эффективности обслуживания клиентов книжного магазина за счет разработки информационной системы для интерактивного книжного терминала.

Разработанная информационная система позволяет оптимизировать и ускорить процессы самостоятельного поиска клиентом, улучшить качество обслуживания покупателей и эффективность работы магазина в целом.

ПРИЛОЖЕНИЕ А

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace electroPomohnik

private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "") return;
    if (textBox2.Text == "") return;
    if (textBox3.Text == "") return;
    if (textBox4.Text == "") return;
    if (textBox5.Text == "") return;
    DataRow[] datarows = null;
    string s = string.Empty;
    {
        datarows = Emptable.Select("id=max(id)");
        s = datarows[0]["id"].ToString();
    }
    catch (Exception)
    {
    }
    if (s == "" || s == string.Empty)
    {
        s = "0";
    }
}
```

```

DataRow datarow = Employees.Tables[0].NewRow();
datarow[0] = Convert.ToString(i);
datarow[1] = textBox1.Text.Trim();
datarow[2] = textBox2.Text.Trim();
datarow[3] = textBox3.Text.Trim();
datarow[4] = textBox4.Text.Trim();
datarow[5] = textBox5.Text.Trim();
DateTime date1 = DateTime.Now;
datarow[6] = date1.ToString("G");
Employees.Tables[0].Rows.Add(datarow);
textBox1.Text="";
textBox2.Text="";
textBox3.Text="";
textBox4.Text="";
textBox5.Text="";
Employees.WriteXml(myDirectory
@"employees.xml",XmlWriteMode.WriteSchema);
Employees = new DataSet();
Employees.ReadXml(myDirectory + @"employees.xml",XmlReadMode.Auto);
Emptable = Employees.Tables[0];
public class Employee
{
    public string EmployeeID { get; set; }
    public string DepartmentName { get; set; }
    public string ePosition { get; set; }
    public string EmployeeName { get; set; }
    public string ProjectName { get; set; }
    public string PostalCode { get; set; }
    public string TaskName { get; set; }
}
public List<Employee> GetTaskList()
{
    if (EmployeeList == null)
        createList();
    return EmployeeList;
}

```

+

```

    }
private void createList()
{
EmployeeList = (
    from e in XDocument.Load("employees.xml").
        Root.Elements("employee")
select new Employee
{
    EmployeeID = (string)e.Element("id"),
    DepartmentName = (string)e.Element("department"),
    ePosition = (string)e.Element("position"),
    EmployeeName = (string)e.Element("name"),
    ProjectName = (string)e.Element("project"),
    TaskName = (string)e.Element("task"),
})
.ToList();
}
List<Employee> ourtasks = GetTaskList();
n = ourtasks.Count;
foreach (var adddep in ourtasks)
{
    ListBox1.Items.Add(adddep.DepartmentName);
}
var Tsks = from myjob in OurTasks
    where myjob.DepartmentName == ListBox1.SelectedItem.ToString();
select myjob;
foreach (var addPos in Tsks)
{
    ListBox2.Items.Add(addPos.Position);
}

```

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Е.Б. Замятина, А.О. Сухов, В.Ю. Ходырева. Опыт применения многомодельного подхода к моделированию бизнес-процессов: Учебное пособие [Электронный ресурс] / Е.Б. Замятина, А.О. Сухов, В.Ю. Ходырева. – 8с. – Режим доступа: <https://www.hse.ru/pubs/share/direct/document/159158812>
2. Харитоновна, Е. В. Графы и сети : учебное пособие / Е. В. Харитоновна. – Ульяновск : УлГТУ, 2006. – 92 с.
3. Бураков,П.В. Введение в системы баз данных: Учебное пособие[Электронный ресурс] / П.В. Бураков, В.Ю. Петров. – СПб, 2010. – 129с. – Режим доступа: <http://books.ifmo.ru/file/pdf/677.pdf>
4. Бейли,Линн. Изучаем SQL / ЛиннБейли. – СПб.:Питер, 2012. – 592с.
5. Молинаро,Э.SQL. Сборник рецептов / Энтони Молинаро– СПб, 2009. – 672с.
6. Питерсон Дж. Теория сетей Петри и моделирование систем. – М: Мир, 1984. 264 с.
7. Котов В.Е. Сети Петри. – М: Наука, 1984. – 160 с.
8. Фадин Д.Н. Методика трансляции графического представления бизнес-процесса предприятия в раскрашенную сеть Петри / Д.Н. Фадин, Д.В. Александров // Вестник ТГТУ. – 2007. – Т. 13. – № 2А. – С. 439–444.
9. Шилдт,Г. С# 4.0 полное руководство / Герберт Шилдт. –СПБ, 2011. – 1056с.
10. Евсеева,О.Н. Работа с базами данных на языке С#. Технология ADO .NET / О.Н. Евсеева, А.Б. Шамшев. – Ульяновск: УлГТУ, 2009. – 170 с.
11. Хейлсберг.А.Язык программирования С#. Классика ComputersScience /А. Хейлсберг, М. Торгерсен, С. Вилта5мут, П. Голд. – СПб, 893с.
12. Стиллмен,Э.Изучаем С#. 3-е изд. / Э. Стиллмен, Дж. Грин – СПб, 643с.

13. Вадим Строганов. Язык программирования С# [Электронный ресурс] – Режим доступа: <https://vadimstroganov.com/yazykirogrammirovaniya/c-sharp>.

14. Описание среды разработки Microsoft Visual Studio [Электронный ресурс] – Режим доступа: http://studbooks.net/2258619/informatika/opisanie_sredy_razrabotki_microsoft_visual_studio.

15. Документация Business Studio » Документы » Проектирование системы управления » Моделирование бизнес-процессов » Нотация IDEF0 [Электронный ресурс] – Режим доступа: <http://www.businessstudio.ru/wiki/docs/v4/doku.php/ru/csdesign/bpmodeling/idef0>

16. Автоматизация технологических процессов и производств. курс лекций для студентов специальности 2203.01 Январь 2010 года [Электронный ресурс] – Режим доступа: <https://studfiles.net/preview/4130283/>

17. Лахатин, А.С. Языки программирования. Учеб. пособие / А.С. Лахатин, Л.Ю. Искакова. - Екатеринбург, 1998. - 548с.: ил.

18. Язык программирования С#. Достоинства и недостатки, основные возможности. Сопоставление с другими языками программирования [Электронный ресурс] – Режим доступа: [http://iupr.ru/domains_data/files/zurnal_22/PERCEV%20S.A.%20%20\(Informacionnye%20i%20kommunikativnye%20tehnologii\).pdf](http://iupr.ru/domains_data/files/zurnal_22/PERCEV%20S.A.%20%20(Informacionnye%20i%20kommunikativnye%20tehnologii).pdf)

20. Microsoft Руководство по языку С# [Электронный ресурс] – Режим доступа: <https://docs.microsoft.com/ru-ru/dotnet/csharp>

21. Microsoft Руководство по .NET Framework [Электронный ресурс] – Режим доступа: <https://docs.microsoft.com/ru-ru/dotnet/framework/>

22. Microsoft Создание приложений UWP с помощью Visual Studio [Электронный ресурс] – Режим доступа: <https://msdn.microsoft.com/library/windows/apps/dn609832.aspx>

23. Учебный курс МГТУ им. Баумана “Основы САПР. Моделирование”. Сети Петри. Анализ сетей Петри

24. В. Л. Конюх, А. Ю. Михайлишин – Имитатор сетей Петри и опыт его применения [Электронный ресурс] – Режим доступа: <https://bourabai.ru/cm/simulator.htm> <https://bourabai.ru/cm/simulator.htm>
25. Мурата Т. Сети Петри: свойства, анализ и приложения//Тр. ТИИИЭР/Пер. с англ. – 1989. – Т. 77. – № 4. – С. 41–79.
26. Котов В.Е. Сети Петри. - М.: Наука, 1984.- 160с.
27. Аппаратно-программная платформа [Электронный ресурс] – Режим доступа: <https://habr.com/company/scancode/blog/194862/>
28. Требования к аппаратной платформе [Электронный ресурс] – Режим доступа: <http://koriolan404.narod.ru/tipis/31.htm>
29. Книжная торговля: от бизнес-идеи до открытия магазина [Электронный ресурс] – Режим доступа: <https://ipexperts.ru/biznes/idei/biznes-plan-knizhnogo-magazina.html>
30. Книжный магазин [Электронный ресурс] – Режим доступа: <http://www.to-ural.ru/магазинностроения>

Выпускная квалификационная работа выполнена мной совершенно самостоятельно. Все использованные в работе материалы и концепции из опубликованной научной литературы и других источников имеют ссылки на них.

« » _____ Г.

(подпись)

(Ф.И.О.)