

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Компьютерных технологий

Кафедра проектирования информационных компьютерных систем

Дисциплина "Современные технологии проектирования информационных систем"

К защите допустить:
Руководитель курсовой работы

10.07.2025

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

на тему

Закупочная логистика предприятия и разработка подсистемы оценки поставщиков

БГУИР КР 1-40 05 01-10 № 176 ПЗ

Студент

(подпись студента)

К.Г. Сарафинович

Курсовая работа
представлена на проверку
10.07.2025

(подпись студента)

Минск 2025

Реферат

БГУИР КР 1-40 05 01-10 № 176 ПЗ, гр. 784371

К.Г. Сарафинович, Закупочная логистика предприятия и разработка подсистемы оценки поставщиков, Минск: БГУИР - 2025.

Пояснительная записка 44022 с., 18 рис., 0 табл.

Ключевые слова: ООП, JavaScript, Node.js

Предмет Современные технологии проектирования информационных систем,

-

-

Содержание

[Введение](#)

1 [Описание предметной области и классификация сущностей решаемой задачи](#)

2 [Постановка задачи и обзор методов её решения](#)

3 [Модели представления системы и их описание](#)

4 [Информационная модель системы и её описание](#)

5 [Описание алгоритмов, реализующих бизнес-логику серверной части проектируемой системы](#)

6 [Руководство пользователя](#)

7 [Результаты тестирования разработанной системы и оценка выполнения задач](#)

8 [СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ](#)

9 [ПРИЛОЖЕНИЕ А](#)

10 [ПРИЛОЖЕНИЕ Б](#)

[Заключение](#)

[Список использованных источников](#)

[Приложения](#)

Введение

Закупочная логистика представляет собой наиболее совершенную форму управления материально-техническим обеспечением и является составной частью такой науки, как логистика. Логистика - наука об управлении движением материальных, финансовых, информационных и людских потоках в пространстве и во времени от их первичного источника до конечного потребителя. Исходя из этого, закупочная логистика тесно связана со всеми подсистемами логистики, и, тем не менее, выступает как самостоятельная дисциплина. Сегодня закупочная логистика –это подсистема управления материальными потоками в процессе обеспечения предприятия материальными ресурсами, который организует вход материального потока в логистическую систему. Управление материальными потоками на данном этапе развития экономики Республики Беларусь имеет известную специфику, что объясняет необходимость изучения закупочной логистики как отдельной дисциплины. Такая система должна позволять хранить информацию о клиентах, поставщиках предоставляемых товаров транспортируемой продукции и самих заказов. Реализация системы в виде Интернет-приложения, опирающегося на проверенные временем технологии функционирования корпоративных приложений, обеспечит ее высокую надежность и простоту работы для конечного пользователя. Служба закупок как составная часть аппарата управления предприятием. Задачи службы закупок предприятия. Функции службы закупок предприятия в

исследовании рынка закупаемых материальных ресурсов. Функции службы закупок предприятия в области планирования. Функции службы закупок предприятия в оперативной работе по закупкам материальных ресурсов. Функции службы закупок предприятия в организации хранения материальных ресурсов. План закупок как составная часть бизнес-плана предприятия, его содержание. Показатели плана закупок. Процедура разработки плана закупки. Закупочная политика предприятия. Текущие и перспективные планы закупок. Использование балансового метода в планировании закупок. Баланс материально-технического обеспечения предприятия. Остатки и внутренние источники обеспечения. Детерминанты, используемые для принятия решений о закупке: время выполнения заказа, доля своевременных поставок, доля материальных ресурсов в запасах, удобства доставки заказа, возможности экспедирования. Для разработки подсистемы будет использоваться язык программирования HTML, JavaScript, Node.js, база данных MongoDB. Целью работы является разработка подсистемы учета и регистрации заявок на перевозку груза. Функциональные возможности подсистемы должны включать возможность добавления, удаления, редактирования клиентов, поставщиков и товаров. Аутентификация пользователя по логину и паролю. В качестве технологий, повышающих надежность, масштабируемость и функциональность разрабатываемого приложения, будут использоваться AngularJS.

1 Описание предметной области и классификация сущностей решаемой задачи

Кратко рассмотрим предметную область, задачи которой подлежат автоматизации в ходе выполнения курсового проекта.

Рассматриваемая предметная область – закупочная логистика предприятия и разработка подсистемы оценки поставщиков. Основная задача предметной области – ведение списка поставщиков, клиентов и товаров. Подлежащая автоматизации задача – учет поступающих заказов.

Процесс начисления осуществляется следующим образом:

- 1) создается контрагент;
- 2) создается поставщик;
- 3) создается заказ на перевозку груза.

При помощи процессного подхода в системе BpWin 4.1 была построена контекстная диаграмма последовательности регистрации заказа на перевозку груза (приложение А).

На контекстной диаграмме отражены основные входные потоки, выходные данные, ресурсы и управление.

Управляющим воздействием здесь является руководство пользователя, составленное на основе общих правил пользования приложением. Ресурсами являются администратор, заказчик и исполнитель заказов.

В качестве входных данных используются:

- информация о заказе. Тут представлена информация, которая необходима исполнителю заказа.
- регистрационные данные, т.е. логин и пароль администратора/исполнителя заказов.
- информация о поставщике/клиенте, все необходимое для заказчика.

Были выделены основные процессы и более мелкие, затронутые в рассматриваемой

предметной области.

Данные процессы представлены в виде диаграммы методологии IDEF0 (приложение А).

2 Постановка задачи и обзор методов её решения

Прежде чем начинать разработку программного продукта, необходимо ознакомиться с требованиями к нему:

1. Необходимо создать интерфейс, который был бы прост и понятен для пользователя. Среда разработки должна позволять с наименьшими затратами справиться с поставленной на данном этапе задачей. Этим требованиям соответствует интегрированная среда разработки, такая как JavaScript.

2. Операционная система Windows 7 и выше.

3. В качестве хранилища информации требуется использовать реляционную базу данных, создаваемую с помощью MongoDB.

4. Интерфейс программы и данные должны быть только на русском языке.

5. Бизнес-логика системы должна быть реализована с использованием компонентов AngularJS.

6. Приложение требуется запускать без использования интегрированных средств разработки.

Цель данной работы - создание подсистемы, которая будет легка в управлении, но при этом эффективна.

Таким образом, поставлены следующие задачи:

Для заказчика:

- создание простого, понятного и удобного в использовании интерфейса;
- просмотр товаров;
- заполнение и отправка заказа;

Для администратора данного сервиса:

- просмотр, редактирование и удаления клиентов;
- просмотр, редактирование и удаления поставщиков;
- просмотр, редактирование и удаления товаров;
- возможность добавления, удаления, редактирования роли;

Для исполнителя заказов данной подсистемы:

- просмотр заказов;
- возможность исполнения заказов;

Для решения данных задач будут использоваться следующие методы:

1. Анализ предметной области с построением диаграммы IDEF0
2. Разработка структуры базы данных с помощью средств Erwin Data Modeler.
3. Анализ базы данных, приведение ее к 3-ей нормальной форме.
4. Использование паттернов проектирования.
5. Организация бесперебойной связи с базой данных.

3 Модели представления системы и их описание

UML (англ. Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования в основном программных систем. UML не является языком программирования, но в средствах выполнения UML-моделей как интерпретируемого кода возможна кодогенерация.

Диаграммы вариантов использования описывают функциональное назначение системы или то, что система должна делать. Разработка диаграммы преследует следующие цели:

- определить общие границы и контекст моделируемой предметной области;
- сформулировать общие требования к функциональному поведению проектируемой системы;
- разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей;
- подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Суть диаграммы вариантов использования состоит в следующем. Проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью вариантов использования.

Диаграмма классов отражает разрабатываемые пользовательские классы.

Диаграмма последовательностей (sequence diagram) используется для точного определения логики сценария выполнения прецедента. Диаграммы

последовательностей отображают типы объектов, взаимодействующих при исполнении прецедентов, сообщения, которые они посылают друг другу, и любые возвращаемые значения, ассоциированные с этими сообщениями.

После подключения к БД разрабатываемая система может находиться в состоянии обработки данных и ожидания запроса на обработку, а также в режиме предоставления результата. Все диаграммы представлены в приложении А.

4 Информационная модель системы и её описание

Информационная модель системы – это схема, представляющая наиболее значимые объекты разрабатываемой системы, их атрибуты, а также связь между этими объектами.

Разработка ER-модели является важным этапом в создании информационной системы и проходит несколько этапов:

- идентификация сущностей и их атрибутов;
- идентификация отношений между сущностями и указания типов отношений;

Исследовав в первом разделе курсового проекта предметную область разработки информационной системы можно определить сущности. Для разрабатываемой системы имеют

значение выбранная продукция. Необходимо знать, какую продукцию для производства выбирал заказчик, идентификацию этого заказчика, его заказы.

Также важное значение имеет информация о заказах для исполнителя заказов .

Таким образом выделены следующие сущности:

- пользователь;
- администратор.

На рисунке 4.1 представлена информационная модель системы.

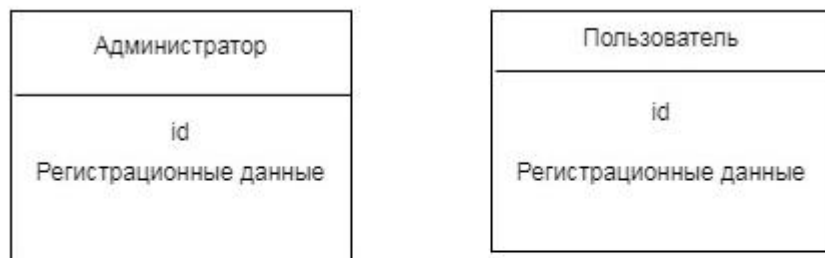


Рисунок 4.1 – ER – диаграмма

Пользователь и администратор не связаны между собой, обладают разными правами. Кроме того, один администратор может обработать несколько поставщиков и клиентов.

Чтобы подтвердить правильность разработанной диаграммы, рассмотрим следующие параметры:

Для того чтобы определить правильность и корректность разработанной ER-модели необходимо провести проверку качества сущностей, атрибутов и связей.

Проверка качества сущностей:

- имена сущностей - существительное в единственном числе или записаны компактным словосочетанием;

- смысл каждой сущности отражен в ее имени;
- сущности не являются разновидностью другой сущности с упущенной рекурсивной связью;
- каждая сущность согласуется с принципами нормализации отношений;
- ключ сущности действительно уникально идентифицирует каждый ее экземпляр;
- ключ сущности действительно минимален;

Проверим атрибуты выделенных сущностей:

- имя атрибутов - существительное в единственном числе;
- смысл атрибута отражен в его имени;
- описание атрибута является достаточно кратким и смысловым;
- атрибуты не представляют упущенную связь;
- атрибуты не являются агрегатами других данных;
- значение обязательного атрибута всегда известно.

Проверка связей:

- связи действительно необходимы;

- если связь обязательная, то всегда определена сущность с другого конца.

Разработанная ER - диаграмма соответствует всем указанным требованиям и может быть использована для построения базы данных.

5 Описание алгоритмов, реализующих бизнес-логику серверной части проектируемой системы

Обобщенный алгоритм работы клиента показан на рисунке 5.1



Рисунок 5.1 - Обобщенный алгоритм работы.

Алгоритм обработки действий пользователя:

- о Пользователь вводит в браузере адрес страницы или нажимает на гиперссылку
- о Браузер посылает запрос на сервер
- о В случае доступности сервера вызывается соответствующая JSP-страница
- о Страница обращается к серверным компонентам бизнес-уровня, отвечающим за работу с данными
- о Страница отображает результаты обработки данных (поиск, результаты удаления и т.п.)
- о Клиент просматривает результаты действия в окне браузера.

6 Руководство пользователя

Для того что бы установить данный проект и запустить его необходимо сервер jboss 5.1.0 и база данных Robo3T.

Система готова к развертыванию. Основные этапы этого процесса:

1. Создание базы данных на сервере mongoDB.
2. Запуск сервера, по сети localhost:3000.

После выполнения этих действий приложение будет доступно по адресу: <http://localhost:3000>

Итогом данного этапа проектирования является развёрнутое и запущенное приложение «Закупочная логистика предприятия и разработка подсистемы оценки поставщиков». Главное окно программы представлено на рис. 6.1.

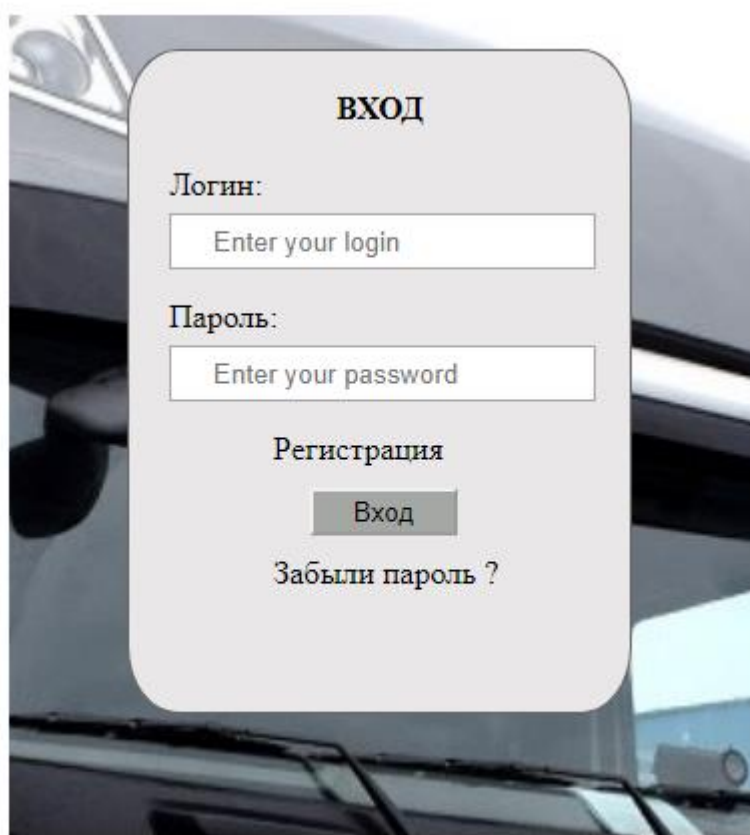


Рисунок 6.1 –Окно входа в программу

Как описывалось ранее, существует два пользователя системы: администратор, пользователь. Ниже представлены формы окон этих пользователей, если входить в систему под их учетными записями.

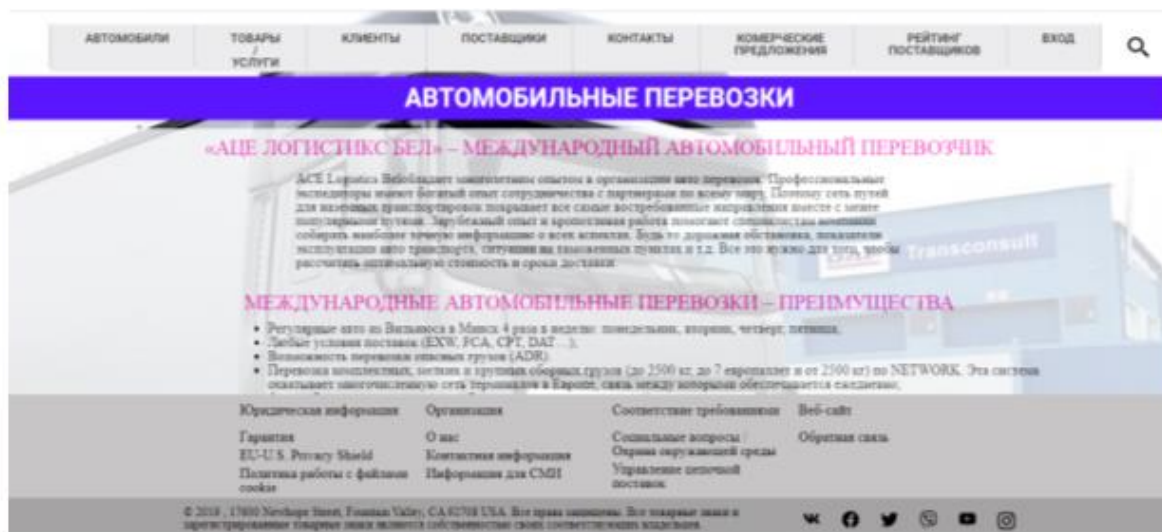


Рисунок 6.2 – Страница пользователя Администратор

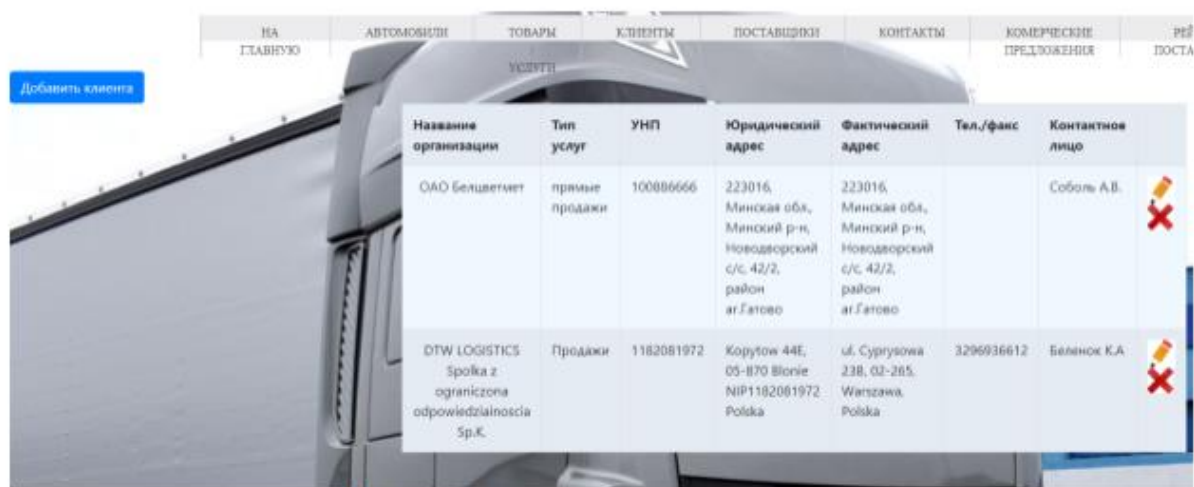


Рисунок 6.3 – Страница клиенты

ВА ГЛАВНУЮ	АВТОМОБИЛИ	ТОВАРЫ	КЛИЕНТЫ	ПОСТАВЩИКИ	КОНТАКТЫ	КОМЕРЧЕСКИЕ ПРЕДЛОЖЕНИЯ	РЕЙТИНГ ПОСТАВЩИКОВ
УСЛУГИ							
Добавить поставщика							
Название организации	Тип услуг	УНП	Юридический адрес	Фактический адрес	Тел./факс	Контактное лицо	Банковский счет
DTW LOGISTICS Spółka z ograniczoną odpowiedzialnością Sp.K.	Продажи	00891340150	Корутов 44E, 05-870 Błonie NIP1182081972 Polska	ul. Surysowa 238, 02-265, Warszawa, Polska	3296936612	Беленок К.А.	AKBBBY21121999
ОАО Белдизмел	Продажи	00891340150	223016, Минская обл., Минский р-н, Новодворский с/с 42/2, район ат.Гатково	223016, Минская обл., Минский р-н, Новодворский с/с 42/2, район ат.Гатково	9401637	Ханн Анастасия Анатольевна	AKBBBY21121999
EMONS IMPEX ITALIANA SPA	Продажи	465466654	223016, Гомельская обл., Минский р-н, Новодворский с/с 42/2, район	223016, Гомельская обл., Минский р-н, Новодворский с/с 42/2, район	0	Соболев А.В.	AKBBBY215087

Рисунок 6.4 – Страница поставщики

ВА ГЛАВНУЮ	АВТОМОБИЛИ	ТОВАРЫ	КЛИЕНТЫ	ПОСТАВЩИКИ	КОНТАКТЫ	КОМЕРЧЕСКИЕ ПРЕДЛОЖЕНИЯ	РЕЙТИНГ ПОСТАВЩИКОВ
УСЛУГИ							
Добавить товар/услугу							
Наименование товара/услуги	Код	Единица измерения	Артикул	Страна происхождения	Вес	Услуга	
Тросс	9098989	шт.	0000789	Казахстан	5	и	
Тросс	9098989	шт.	0000789	Казахстан	5	и	
Тросс 2	9098989	шт.	0000789	Казахстан	5	и	
Тросс 2	9098989	шт.	0000789	Казахстан	5	и	
Тросс 2	9098989	шт.	0000789	Казахстан	5	и	
Тросс 2	9098989	шт.	0000789	Казахстан	5	и	

Рисунок 6.5 – Страница товаров и услуг

ВА ГЛАВНУЮ	АВТОМОБИЛИ	ТОВАРЫ	КЛИЕНТЫ	ПОСТАВЩИКИ	КОНТАКТЫ	КОМЕРЧЕСКИЕ ПРЕДЛОЖЕНИЯ	РЕЙТИНГ ПОСТАВЩИКОВ
УСЛУГИ							
Добавить авто							
ФИО	Модель автомобиля	Номер шасси	Год автомобиля	Направление перевозки			
Сеуратович Павел	Scania 434	ПА4654654ВАН	2011	Грузы			
Петрова Андрей Сергеевич	MAN 9043	65445AAD890ЛОД	2016	Россия			

Рисунок 6.6 – Страница автомобилей

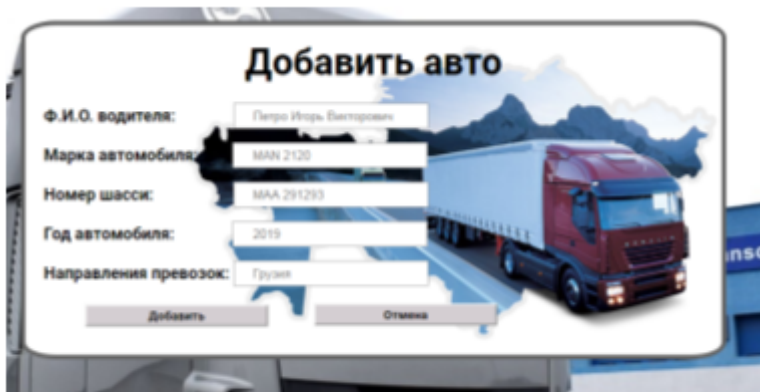


Рисунок 6.7 – Страница создания автомобилей

Представила схему базы данных и как она выглядит в Robo3T

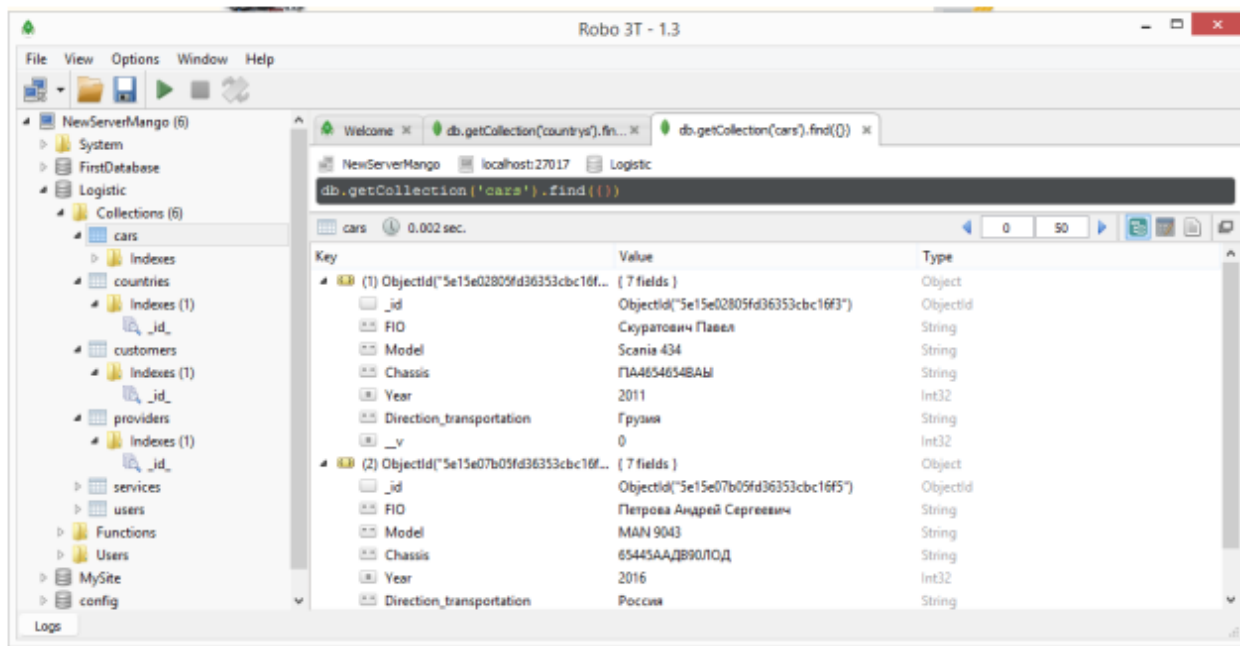


Рисунок 6.6 – База данных

7 Результаты тестирования разработанной системы и оценка выполнения задач

7.1 Результаты тестирования разработанной системы

Тестирование приложения является неотъемлемой частью цикла разработки, а написание и поддержка модульных тестов могут гарантировать корректную работу отдельных методов исходного кода.

Создавался test класс, в котором тестировались DAO методы (test.java). Junit-тесты для тестирования не использовались.

Тестирование бизнес - логики сводится к проверке функциональных возможностей методов, удаленно вызываемых со стороны клиентского приложения.

Методы и классы, реализующие логику приложения, успешно справляются со своими функциями.

7.2 Оценка выполнения задач

В результате решения поставленной задачи была разработана простейшая подсистема создания и регистрации клиентов, поставщиков и услуг закупочной логистики на предприятии.

Разработанная программа после дальнейшей модернизации и дополнительного тестирования может использоваться в любой организации, занимающейся закупочной логистикой перевозкой грузов.

Возможные усовершенствования подсистемы:

1. возможность создания печатных форм и отчетов (данные по результатам клиентов, виды услуг и товаров т.п.)
2. расширение набора сущностей предметной области – добавление сведений о клиентах, поставщиках, отправка документов и печатных форм, просмотр рейтинга поставщиков и т.п.
3. добавление возможностей по расширенному поиску записей системы
4. разграничение категорий пользователей системы и предоставление возможности сотрудникам доступа к полным объемам данных по поставщикам, клиентам.

Дополнительно в результате выполнения данного курсового проекта были получены и закреплены на практике навыки в работе с языком JavaScript, AngularJS, Node.js, HTML, принципы использования методик проектирования IDEF0, IDEF1x, СУБД MongoDB и технологией EJB.

8 СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кей С. Хорстманн, Гари Корнелл. Java 2. Библиотека профессионала. Том 2. Тонкости программирования. – Вильямс, 2006 г., 1168 с.
2. Итан Браун, Веб-разработка с применением Node и Express, 2017 г..
3. Саймон Холмс Стек MEAN, 2017 г..

9 ПРИЛОЖЕНИЕ А

Функциональная модель (IDEF0)

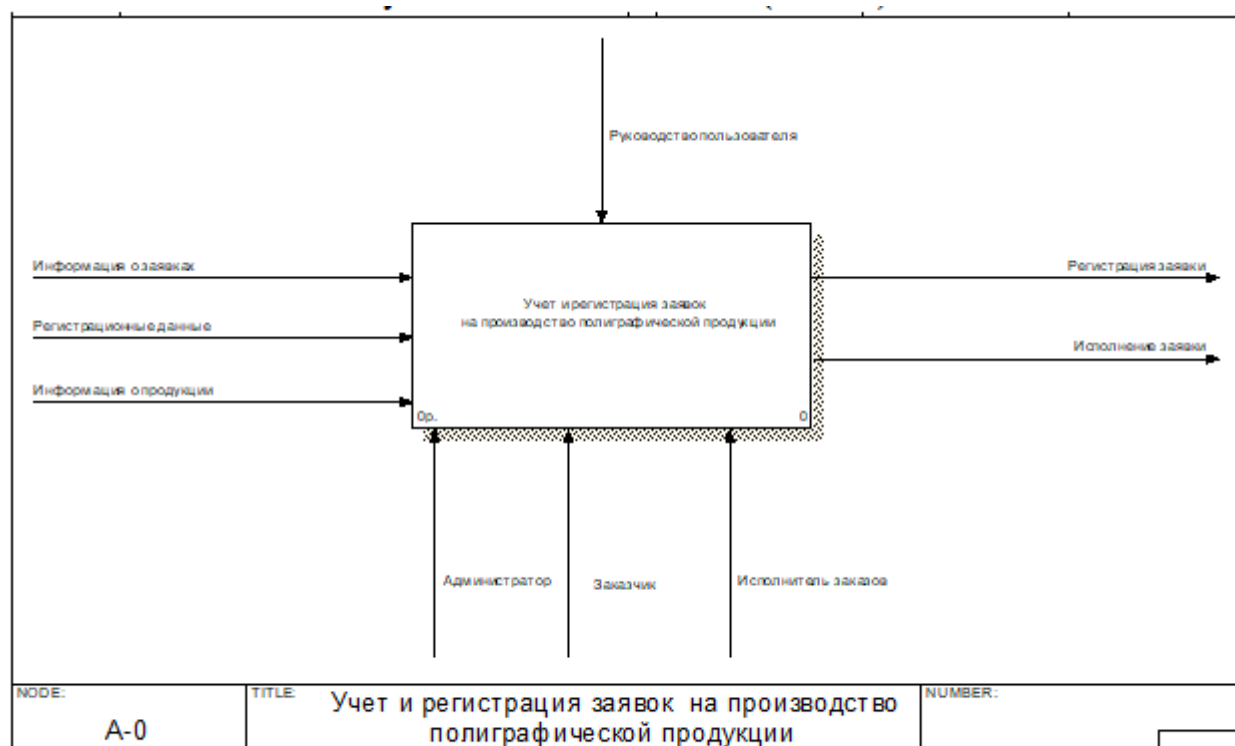


Рисунок. А.1 – Контекстная диаграмма

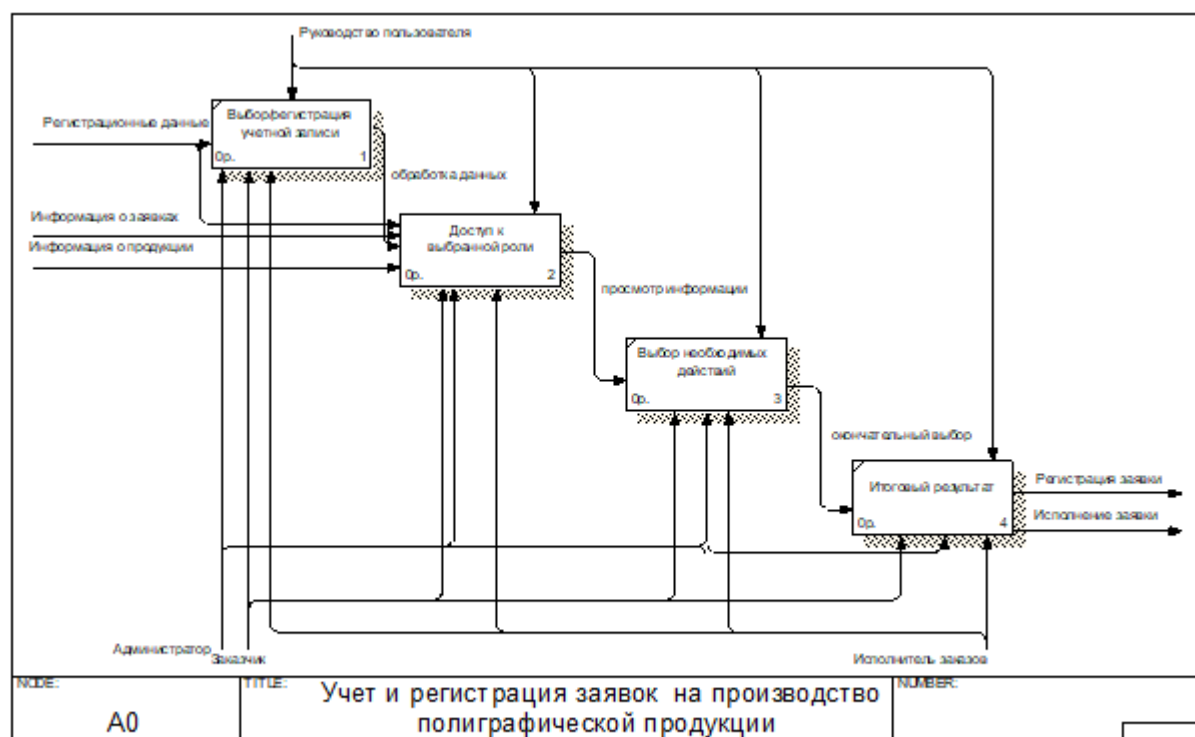


Рисунок. А.2 – Декомпозиция контекстной диаграммы

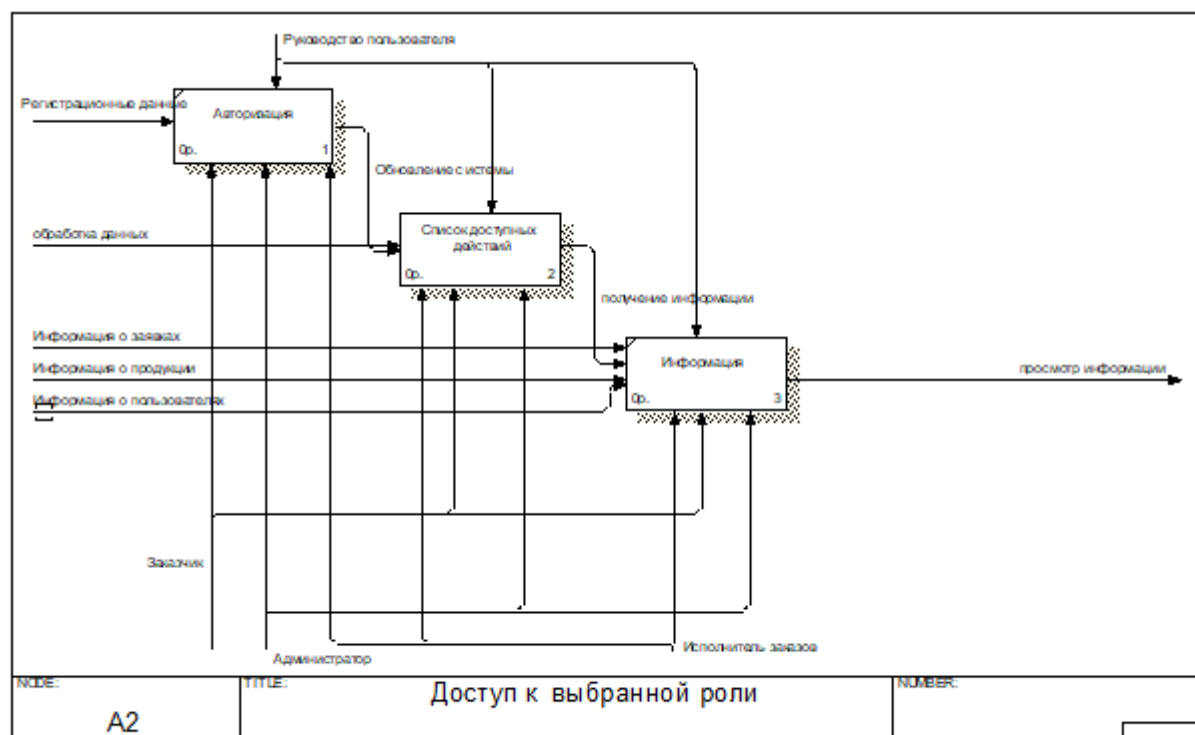


Рисунок. А.3 – Декомпозиция блока «Доступ к выбранной роли»

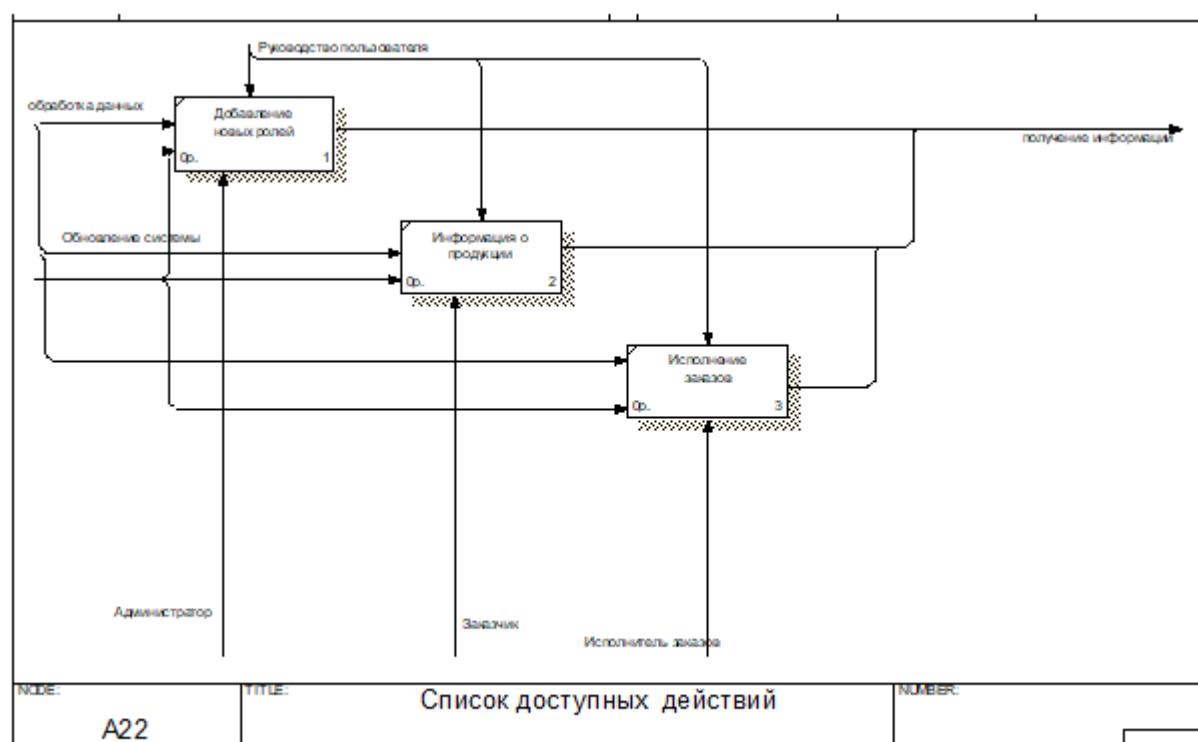


Рисунок. А.4 – Декомпозиция блока «Список доступных действий»

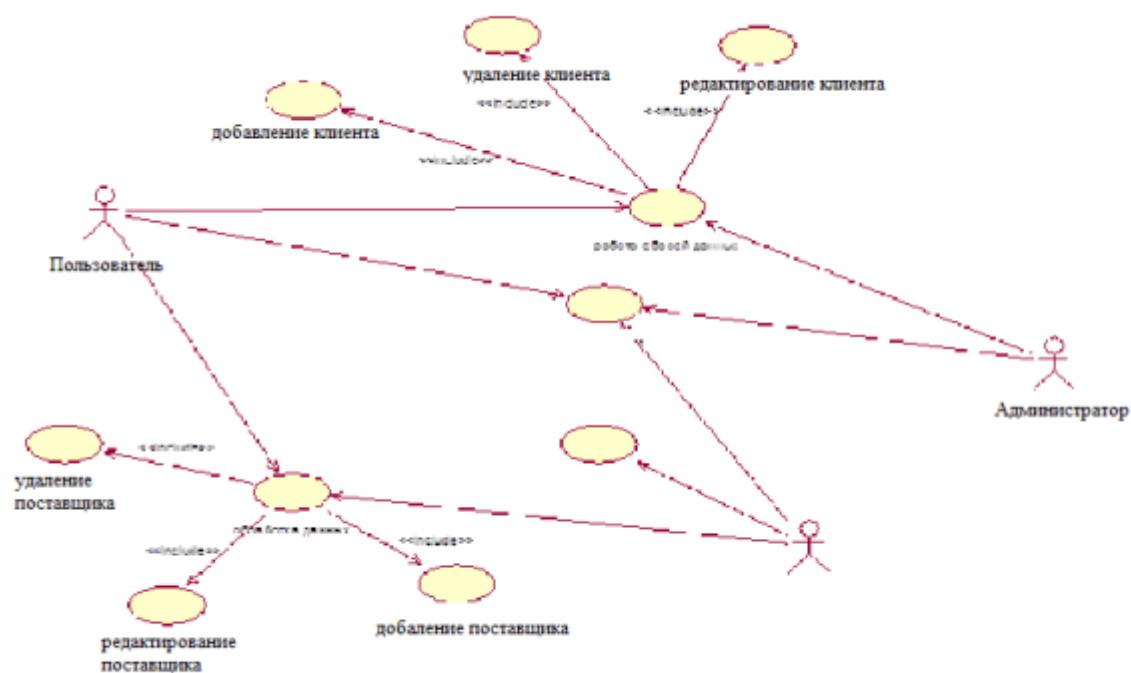


Рисунок. А.5 - Диаграмма вариантов использования

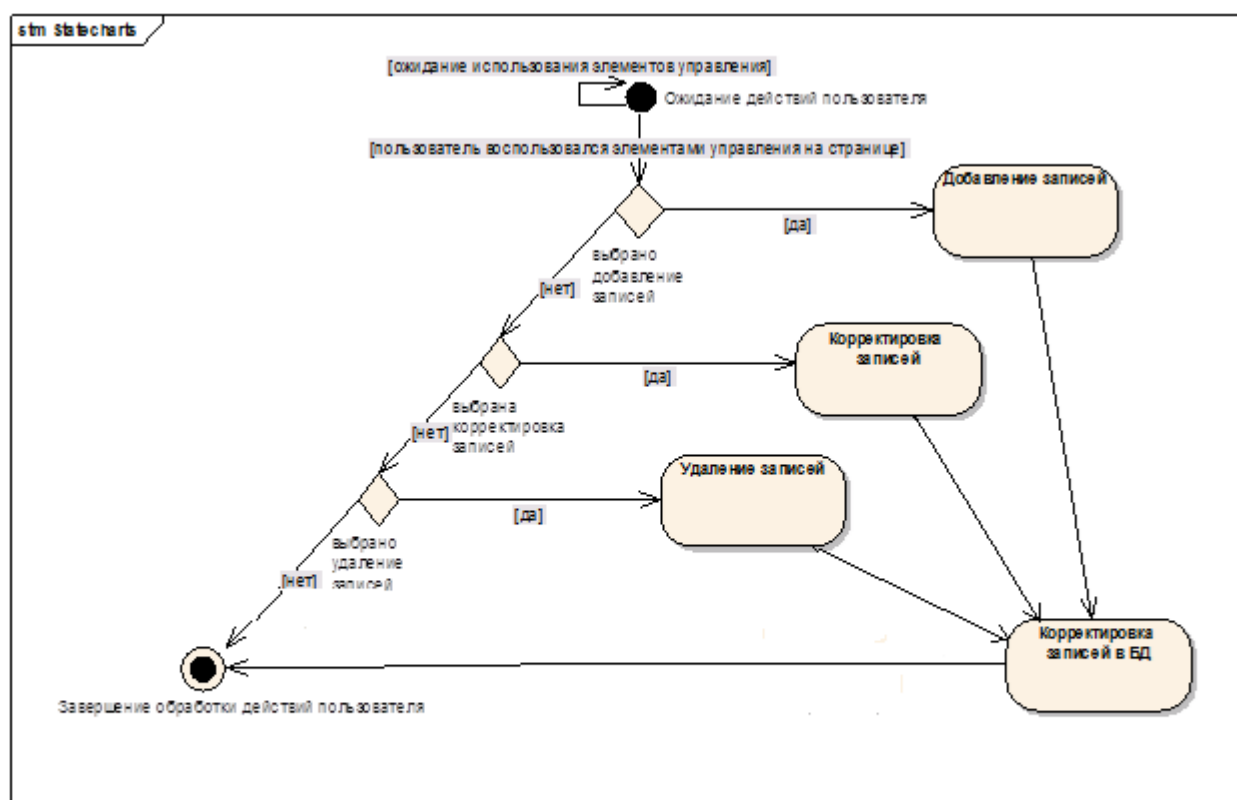


Рисунок. А.6 – Диаграмма состояний

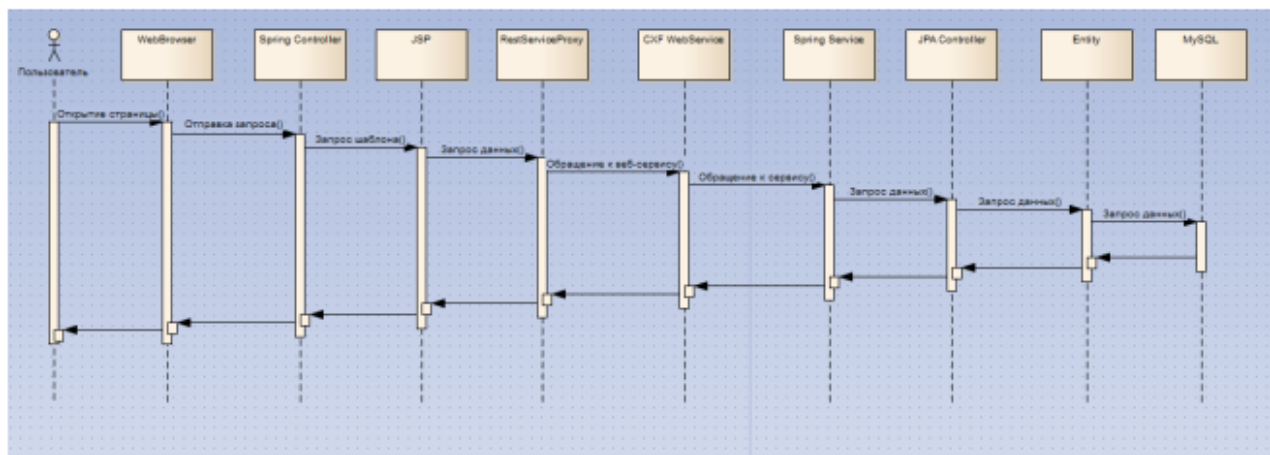


Рисунок А.6 - Диаграмма последовательности

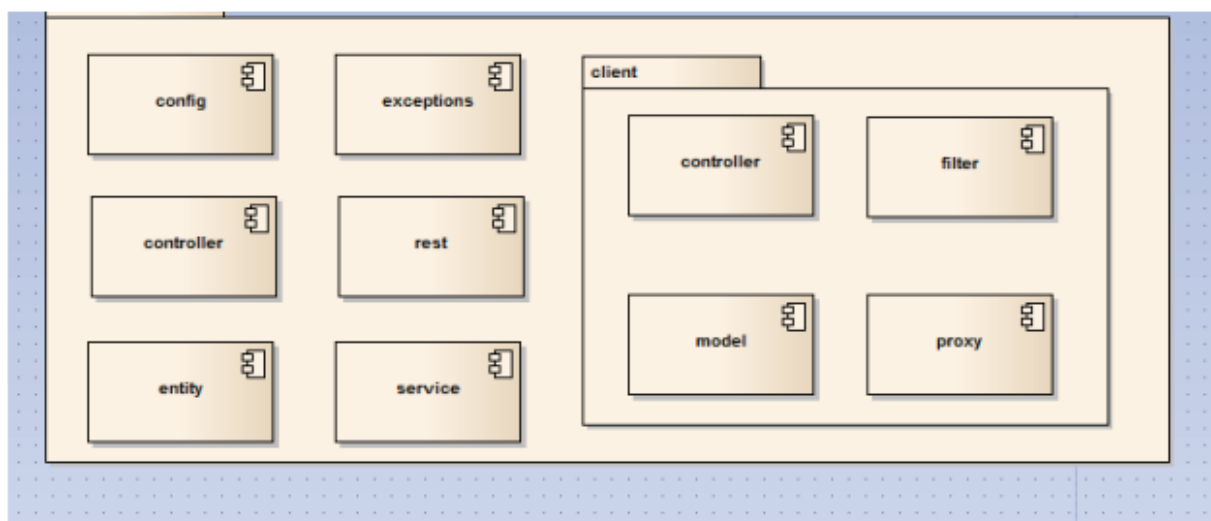


Рисунок А.7 - Диаграмма компонентов

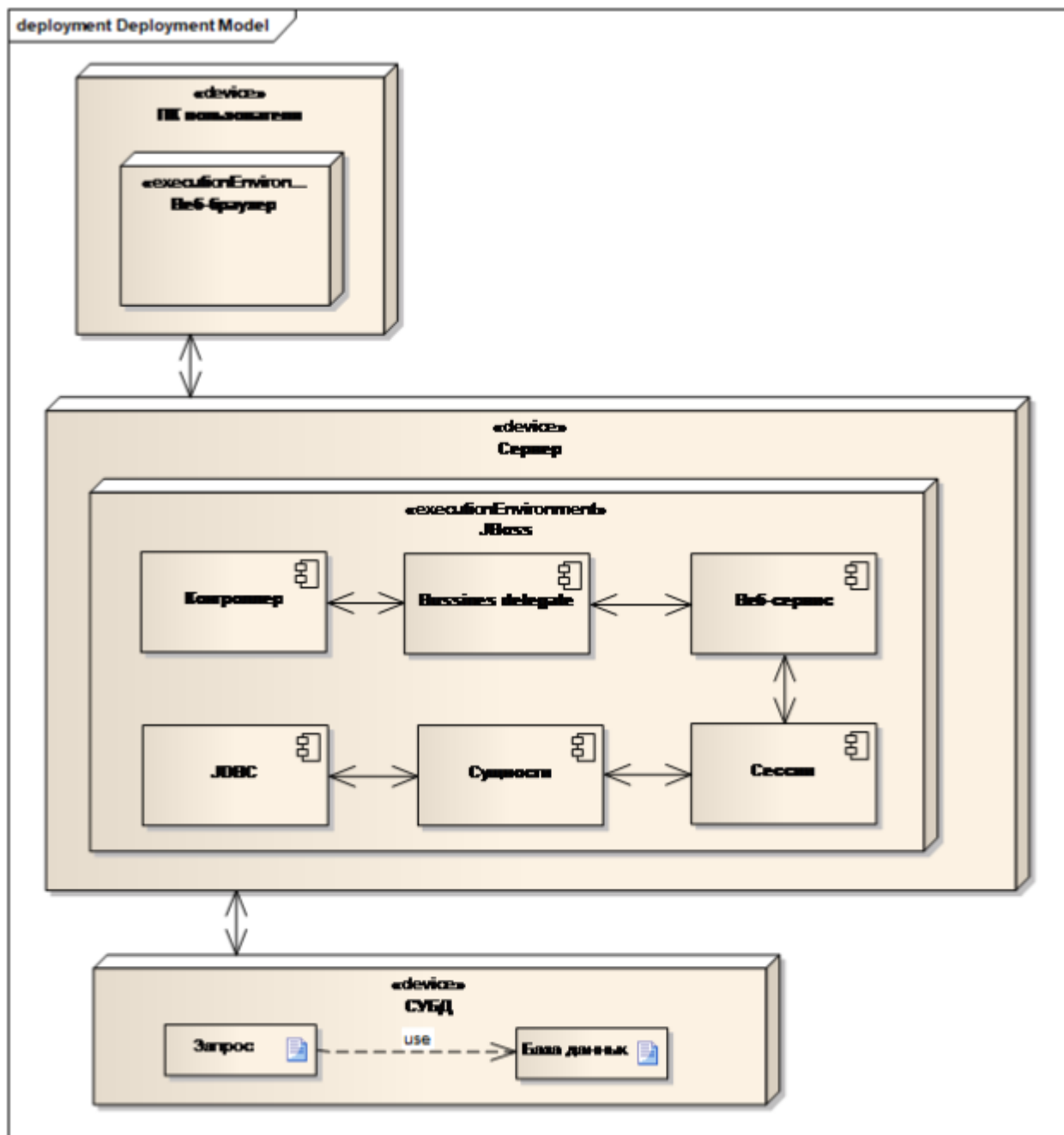


Рисунок А.8 - Диаграмма развёртывания

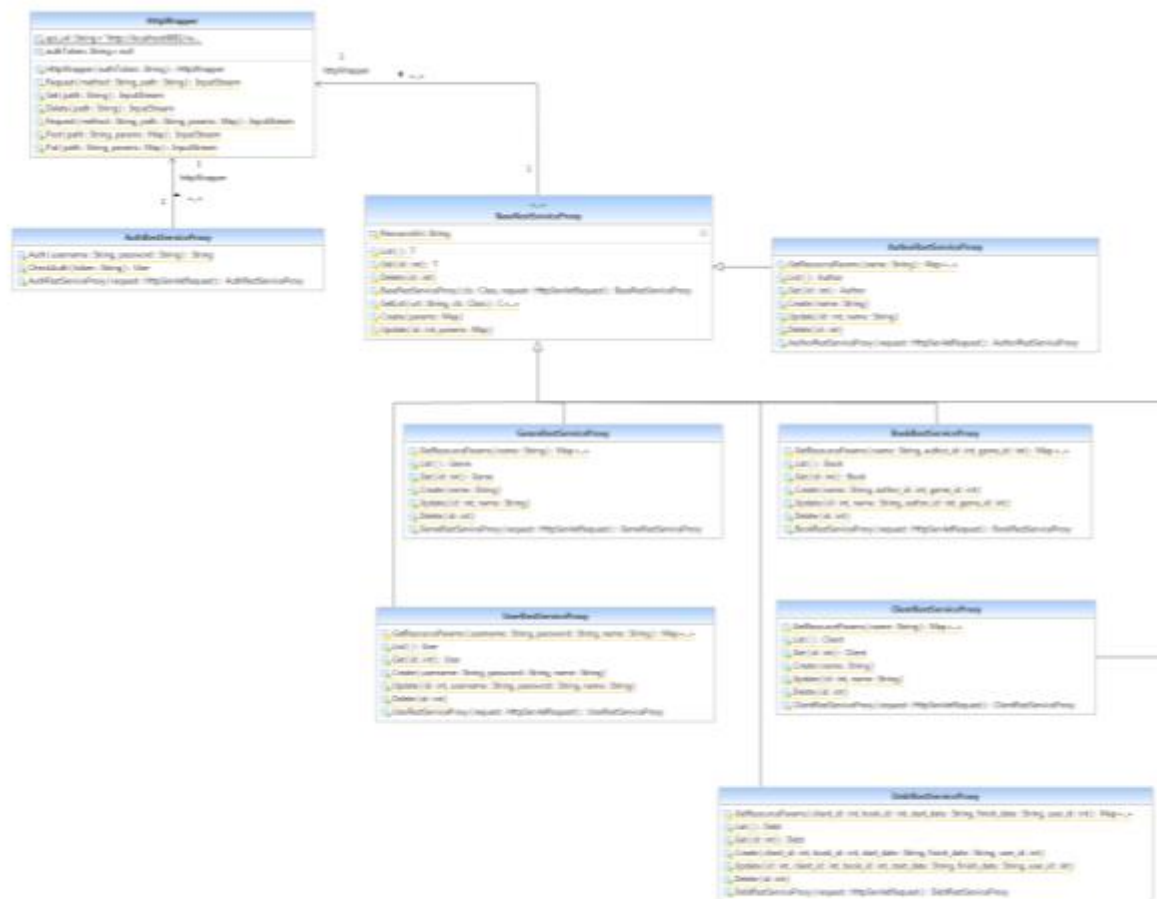
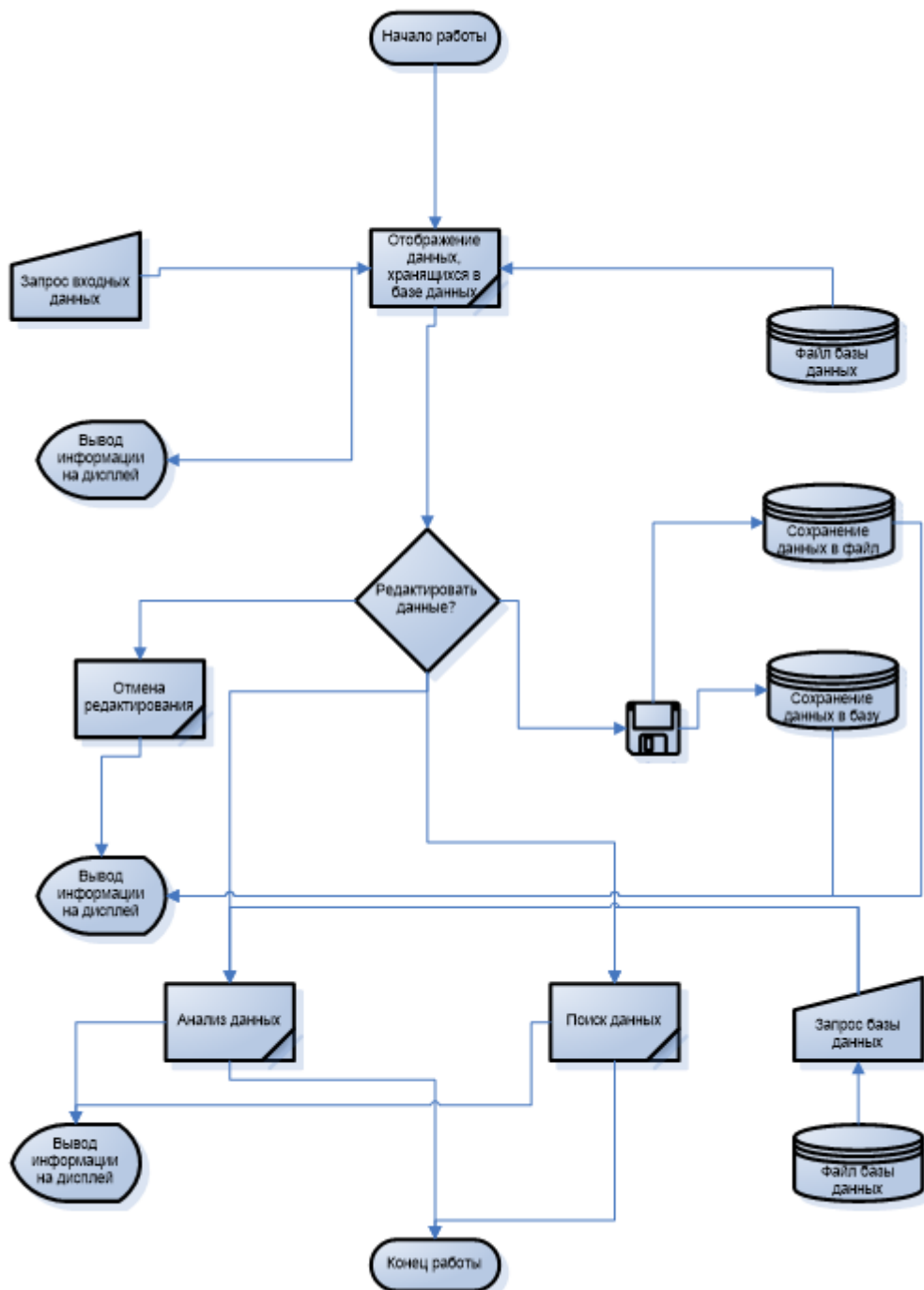


Рисунок А.9 - Диаграмма классов

10 ПРИЛОЖЕНИЕ Б

Блок-схема алгоритма, реализующего бизнес-логику



Заключение

В ходе выполнения курсового проекта была разработана подсистема закупочная логистика предприятия и разработка подсистемы оценки поставщиков. ПО представляет собой образец реального корпоративного J2EE приложения, функционирующего на основе сетевых технологий. Применение архитектуры веб-клиента в разработанной подсистеме предоставляет пользователям возможность с различных компьютеров обращаться к серверу за необходимой информацией. Сервер предоставляет возможность для клиента для работы с базой данных. Он выступает в качестве посредника между клиентом и базой данных. Он принимает запросы от клиента, их обрабатывает и направляет в базу данных. Также он перенаправляет запросы назад клиенту. Клиент имеет возможность работы с информацией, хранимой в базе данных. Такая информация представлена в виде таблиц базы данных. Клиент может просматривать, добавлять и редактировать необходимую информацию. Применение базы данных в качестве хранилища информации позволяет оптимально и эффективно хранить информацию, ее

структурировать. Кроме того, в результате выполнения данного курсового проекта были получены дополнительные навыки в работе с языком JavaScript, Node.js, HTML были получены знания о различных методах построения функциональных и информационных моделей. В ходе тестирования разработанного ПО было установлено, что оно работает корректно и соответствует заявленным функциональным требованиям. В целом можно считать, что цель курсовой работы достигнута.

Список использованных источников

1. [url] **Описание Erwin** . <http://www.ecsocman.edu.ru/db/msg/72668.htm>
2. [url] **Описание BPwin 4.0** <http://www.interface.ru/ca/bpwin2.htm>
3. [url] **Основы javascript** <https://learn.javascript.ru>
4. [url] **Основы java** <http://www.java.sun.com>
5. [url] **JSDK** JSDK 1.7 javadoc

Приложения