**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**ОТЧЕТ**

Работу выполнил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.П.Демьяненко

Краснодар 2023

# **ВВЕДЕНИЕ**

Цель работы: разработка моделей бизнес-процессов информационной системы, автоматизирующей работу аптеки

Задачи информационной системы:

* Построение бизнес-процессов;
* Исследование документов и документооборота;
* Построение структуры БД;

# **1 Постановка задачи и описание предметной области.**

Фотоцентр имеет главный офис и сеть филиалов и киосков приема заказов, расположенных по определенным адресам. Филиалы и киоски различаются количеством рабочих мест. В киосках осуществляется только прием заказов, поэтому каждый киоск прикреплён к определенному филиалу, в котором эти заказы выполняются.

В филиалах имеется необходимое оборудование для проявки пленок и печати фотографий. Филиалы и киоски принимают заказы на проявку пленок, печать фотографий и проявку, и печать вместе. В заказе на печать указывается количество фотографий с каждого кадра, общее количество фотографий, формат, тип бумаги и срочность выполнения заказа. При заказе большого количества фотографий предоставляются скидки. Срочные заказы принимаются только в филиалах, и они имеют цену в два раза больше, чем обычный заказ. При приобретении дисконтной карты клиент получает значительные скидки на печать фотографий. Пленка, приобретенная в том же филиале, куда она принесена на проявку, проявляется бесплатно.

Клиентов можно разделить на профессионалов и любителей. Профессионалам, приносящим заказы в один и тот же филиал, могут быть предложены персональные скидки. Фотомагазины и киоски предлагают к продаже различные фототовары: фотопленки, фотоаппараты, альбомы и другие фотопpинадлежности. Фотомагазины также предлагают дополнительные виды услуг: фотографии на документы, реставрация фотографий, прокат фотоаппаратов, художественное фото, предоставление услуг профессионального фотографа.

Сведения о выполненных заказах и продаже различных фототоваров собираются и обрабатываются, и на основе этой информации делается общий заказ на поставку расходных материалов (фотобумага, фотопленка, химические реактивы), фототоваров и оборудования. Полученные товары и материалы pаспpеделяются в соответствии с запросами по киоскам и магазинам. У фотоцентра может быть несколько поставщиков, которые специализируются на различных поставках, либо на поставках фототоваров различных фирм.

# **2 Функции предметной области**

На сегодняшний день проблема автоматизирования ручного труда крайне важна. Поэтому для современного предпринимателя работа без ЭВМ просто невозможна, так как отсутствие таковой системы обработки информации повлечёт за собой огромные затраты рабочего времени, а также сам процесс обработки будет длиться долго, в то время как с помощью автоматизированной системы это можно сделать гораздо быстрее и эффективнее.

Выделены функции:

* формирование и ведение списка клиентов;
* формирование списков выданных заказов;
* своевременное обновление информации о наличии материалов;
* формирование необходимых отчетов.

# **3 Функциональная модель IDEF0**

Методология IDEFO представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели объекта какой-либо предметной области.

В основе методологии лежат четыре основных понятия: функциональный блок, интерфейсная дуга, декомпозиция, глоссарий.

Функциональный блок(Activity Box) представляет собой некоторую конкретную функцию в рамках рассматриваемой системы. По требованиям стандарта название каждого функционального блока должно быть сформулировано в глагольном наклонении.

На диаграмме функциональный блок изображается прямоугольником. Каждая из четырех сторон функционального блока имеет свое 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).

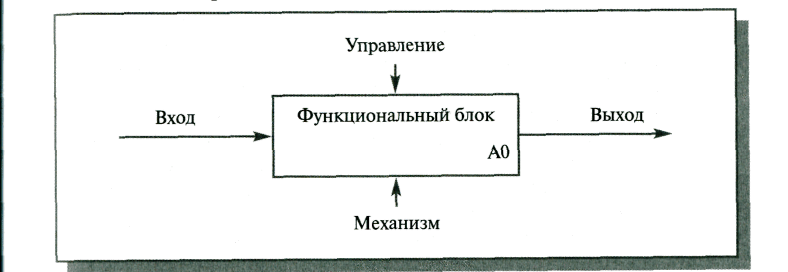


Рисунок 1. Функциональный блок

Интерфейсная дуга(Arrow) отображает элемент системы, который обрабатывается функциональным блоком или оказывает иное влияние на функцию, представленную данным функциональным блоком. Интерфейсные дуги часто называют потоками или стрелками.

Декомпозиция(Decomposition) является основным понятием стандарта IDEF0. Принцип декомпозиции применяется при разбиении сложного процесса на составляющие его функции. При этом уровень детализации процесса определяется непосредственно разработчиком модели.

Модель IDEF0 всегда начинается с представления системы как единого целого — одного функционального блока с интерфейсными дугами, простирающимися за пределы рассматриваемой области. Такая диаграмма с одним функциональным блоком называется контекстной диаграммой.

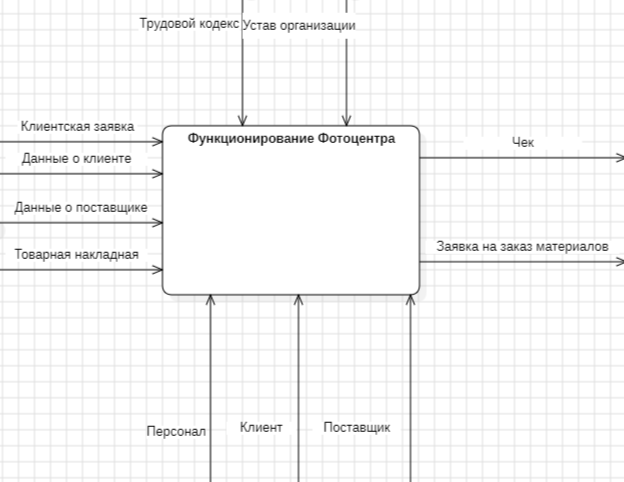


Рис1 – Модель IDEF0

Входными данными явлюяются:

- Клиентская заявка

- Данные о клиенте

- Данные о поставщике

- Товарная накладная

Выходные данные:

- Чек

- Заявка на заказ материалов

Управление:

- Трудовой кодекс

- Устав организации

Механизм:

- Персонал

- Клиент

- Поставщик

Рассмотрим также диаграмму декомпозиции.

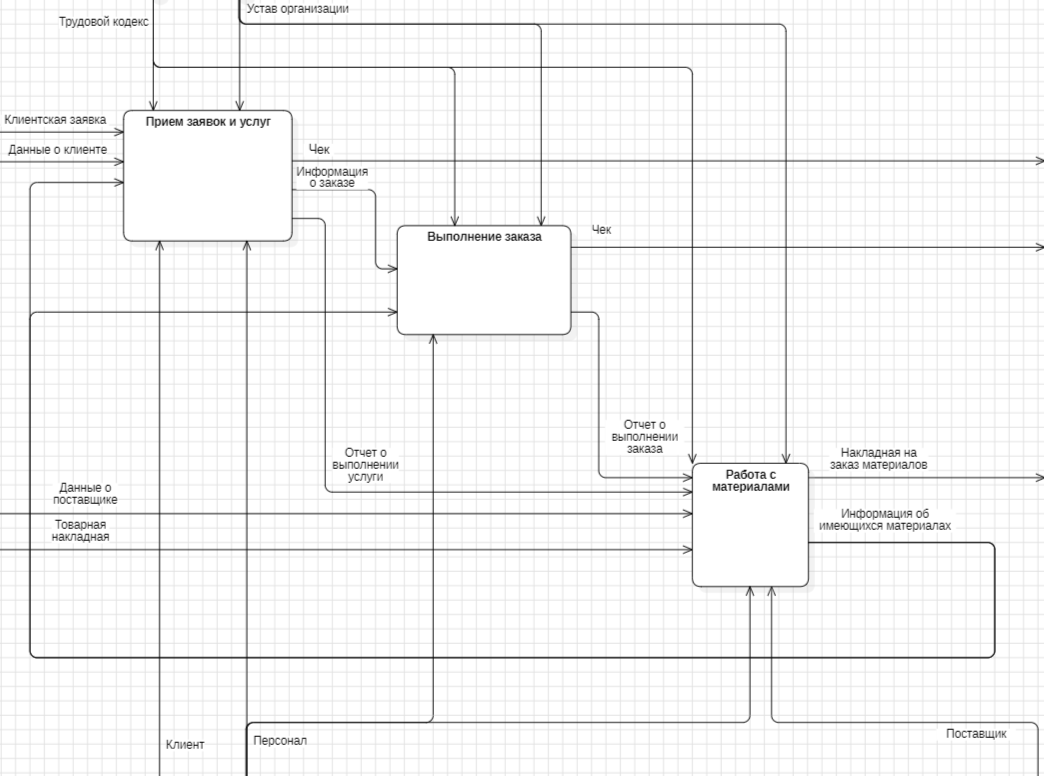


Рис2 – Диаграмма декомпозиции

В данную диаграмму входят три процесса. «Приём заявок и услуг», «Выполнение заказа», «Работа с материалами». Каждый из процессов в свою очередь также декомпозирован. Рассмотрим диаграммы декомпозиции второго уровня на рисунках ниже.

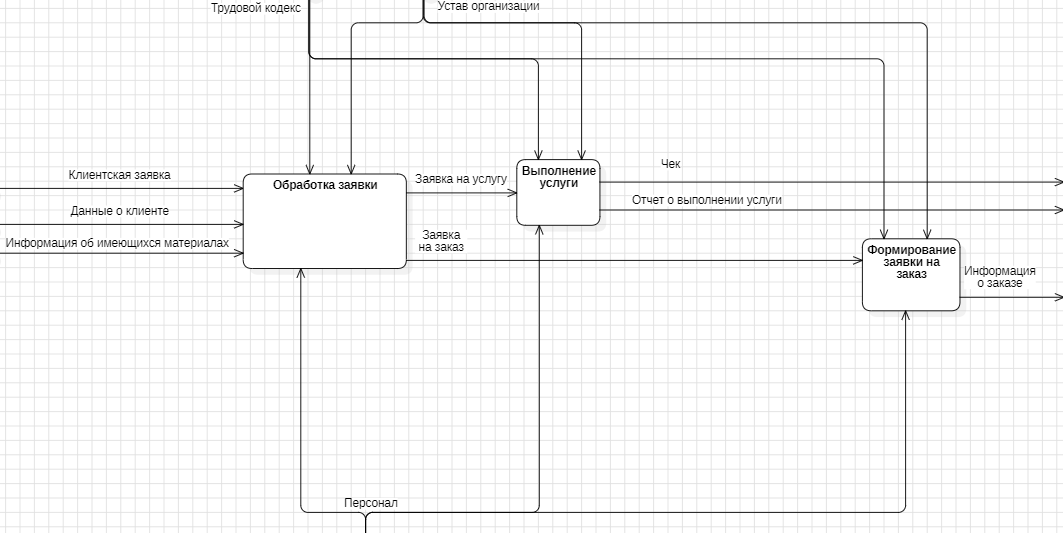


Рис3 - Диаграмма декомпозиции второго уровня процесса «Приём заявок и услуг»

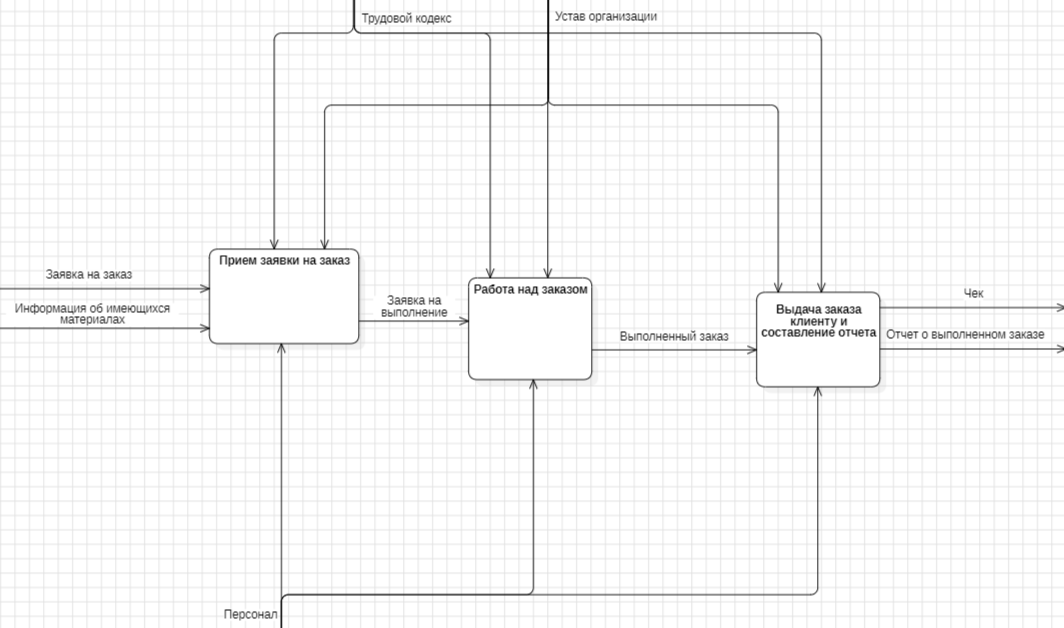


Рис 4 - Диаграмма декомпозиции второго уровня процесса «Выполнение заказа»

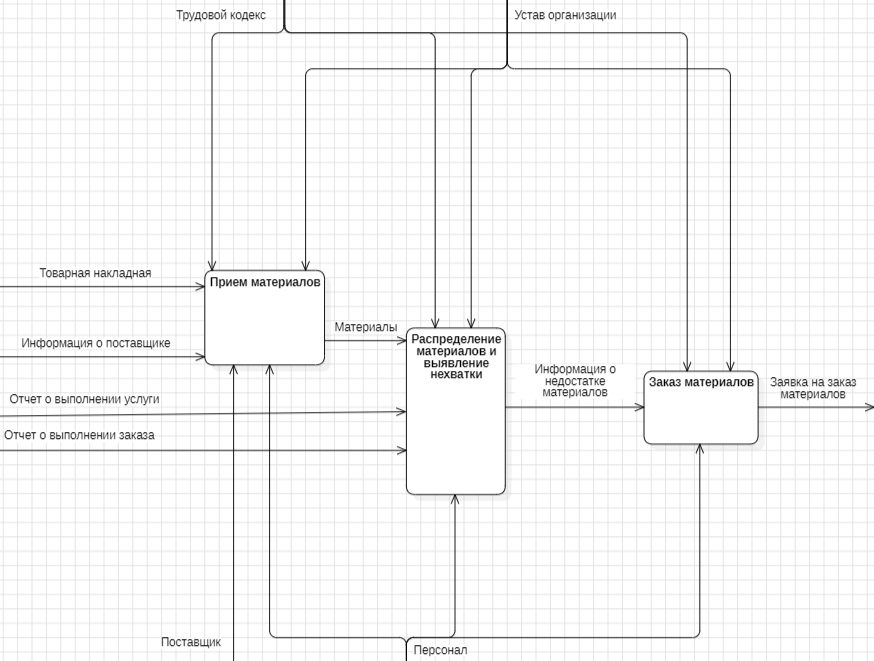


Рис 5 - Диаграмма декомпозиции второго уровня процесса «Работа с материалами»

# **4 Документооборот и диаграмма потока данных**

## **4.1 Диаграмма потока данных DFD**

Диаграммы потоков данных (Data Flow Diagrams — DFD) представляют собой иерархию функциональных процессов, связанных потоками данных. Цель такого представления — продемонстрировать, как каждый процесс преобразует свои входные данные в выходные, а также выявить отношения между этими процессами.

Основными компонентами диаграмм потоков данных являются:

* внешние сущности;
* системы и подсистемы;
* процессы;
* накопители данных;
* потоки данных.

Внешняя сущность представляет собой материальный объект или физическое лицо, являющиеся источником или приемником информации, например, заказчики, персонал, поставщики, клиенты, склад. Определение некоторого объекта или системы в качестве внешней сущности указывает на то, что она находится за пределами границ анализируемой системы.

Процесс представляет собой преобразование входных потоков данных в выходные в соответствии с определенным алгоритмом. Физически процесс может быть реализован различными способами: это может быть подразделение организации (отдел), выполняющее обработку входных документов и выпуск отчетов, программа, аппаратно реализованное логическое устройство и т.д.

Накопитель данных — это абстрактное устройство для хранения информации, которую можно в любой момент поместить в накопитель и через некоторое время извлечь, причем способы помещения и извлечения могут быть любыми.

Поток данных определяет информацию, передаваемую через некоторое соединение от источника к приемнику. Реальный поток данных может быть информацией, передаваемой по кабелю между двумя устройствами, пересылаемыми по почте письмами, магнитными лентами или дискетами, переносимыми с одного компьютера на другой и т.д.

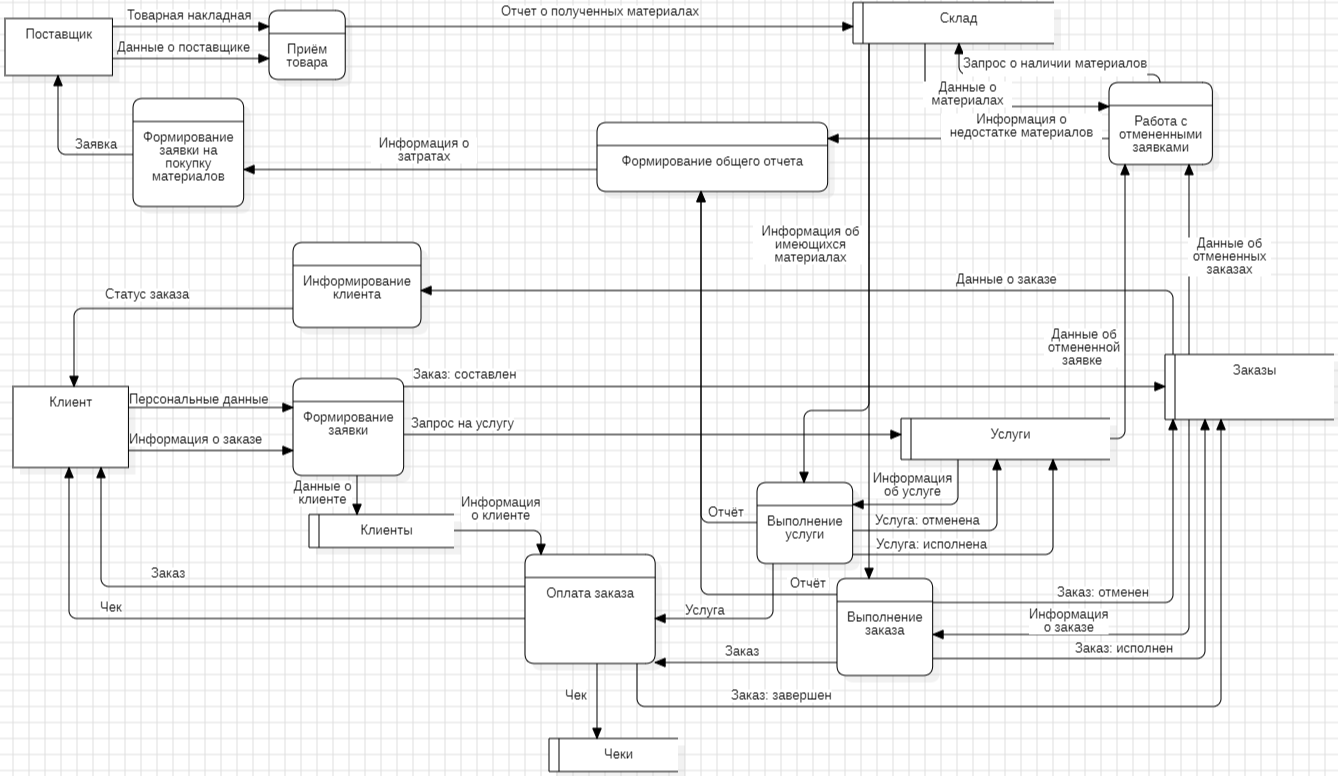


Рис 6 – Диаграмма потока данных DFD

# **5 Диаграмма сущность-связь (ER-диаграмма)**

Для создания базы данных, структура которой не зависит от конкретных информационных потребностей и позволяет выполнить любые запросы пользователей, в индивидуальном задании следует разработать диаграмму инфологической модели «сущность-связь» (ER-диаграмма).

Общий подход к проблеме семантического моделирования характеризуется четырьмя основными этапами:

1. Прежде всего выявляется некоторое множество семантических концепций, понятий, называемых «сущностями» которые могут быть полезны при неформальном обсуждении реального мира. Сущности могут быть классифицированы по разным типам сущностей и тогда все сущности определенного типа обладают некоторыми общими свойствами. Каждая сущность обладает некоторым особым свойством, предназначенным для ее идентификации. Каждая сущность может быть связана с другими сущностями посредством некоторых связей.

2. Далее определяется набор соответствующих символических, формальных объектов, которые могут использоваться для представления описанных семантических концепций.

3. Определяется набор формальных общих правил целостности, предназначенных для работы с такими формальными объектами.

4. Также определяется набор формальных операторов, предназначенных для манипулирования формальными объектами.

Необходимо подчеркнуть, что правила целостности и операторы являются такой же частью модели данных, как и объекты.

Базовыми понятиями ER-диаграмм являются:

Сущность (Entity) — реальный или абстрактный объект, явление или процесс, имеющий существенное значение для рассматриваемой предметной области.

Каждая сущность должна обладать уникальным идентификатором (ами), благодаря чему каждый экземпляр сущности однозначно идентифицируется и отличается от всех других экземпляров данного типа сущности. Каждая сущность должна обладать свойствами:

* иметь уникальное имя;
* обладать одним или несколькими атрибутами, которые принадлежат сущности;
* обладать одним или несколькими идентифицирующими атрибутами, однозначно идентифицирующими каждый экземпляр сущности.

Сущности обладают некоторыми свойствами. Все сущности одного и того же типа обладают некоторыми общими свойствами. Атрибут — любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для классификации, идентификации, классификации, количественной характеристики или выражения состояния сущности.

Каждая сущность может обладать любым количеством связей с другими сущностями модели.

Связь (Relationship)— поименованная ассоциация между двумя сущностями, значимая для рассматриваемой предметной области. Связь — это ассоциация между сущностями, при которой каждый экземпляр одной сущности ассоциирован с произвольным количеством экземпляров второй сущности, и наоборот. Тип связи рассматривается между типами сущностей, а конкретный экземпляр связи рассматриваемого типа существует между конкретными экземплярами рассматриваемых типов сущностей. Связь может быть обязательной, возможной, условной. Чаще всего встречаются связи между двумя типами сущностей (бинарные связи), хотя в модели могут быть выделены связи между любым количеством сущностей (тернарные, … n-арные).

Мощностьсвязи представляет собой отношение количества экземпляров одной сущности к соответствующему количеству экземпляров другой сущности. Связи в модели «сущность-связь» могут иметь мощность:

«один-к-одному» (1:1), «один-ко-многим» (1:M) и «многие-ко-многим» (M:N).

Связь 1:1 («один-к-одному») – это связь между двумя типами сущностей A и B, при которой каждому экземпляру сущности A соответствует один и только один экземпляр сущности B.

Связь 1:M («один-ко-многим») – это связь между двумя типами сущностей A и B, при которой одному экземпляру сущности A может соответствовать ноль, один или несколько экземпляров сущности B, но каждому экземпляру сущности B соответствует только один экземпляр сущности A.

Связь M:N (**«**многие**-**ко**-**многим**»**) – это связь между двумя типами сущностей A и B, при которой каждому экземпляру сущности A может соответствовать ноль, один или несколько экземпляров сущности B и наоборот.

На первом этапе моделирования пытаются выделить множество семантических концепций, полезных для описания предметной области (части реального мира, имеющей интерес для проектировщика ИС).

Цель моделирования данных состоит в обеспечении разработчика ИС концептуальной схемой базы данных в форме одной или нескольких моделей, которые относительно легко могут быть отображены в любую систему баз данных. Наиболее распространенным средством моделирования данных являются модели "сущность-связь" (ER-модели).

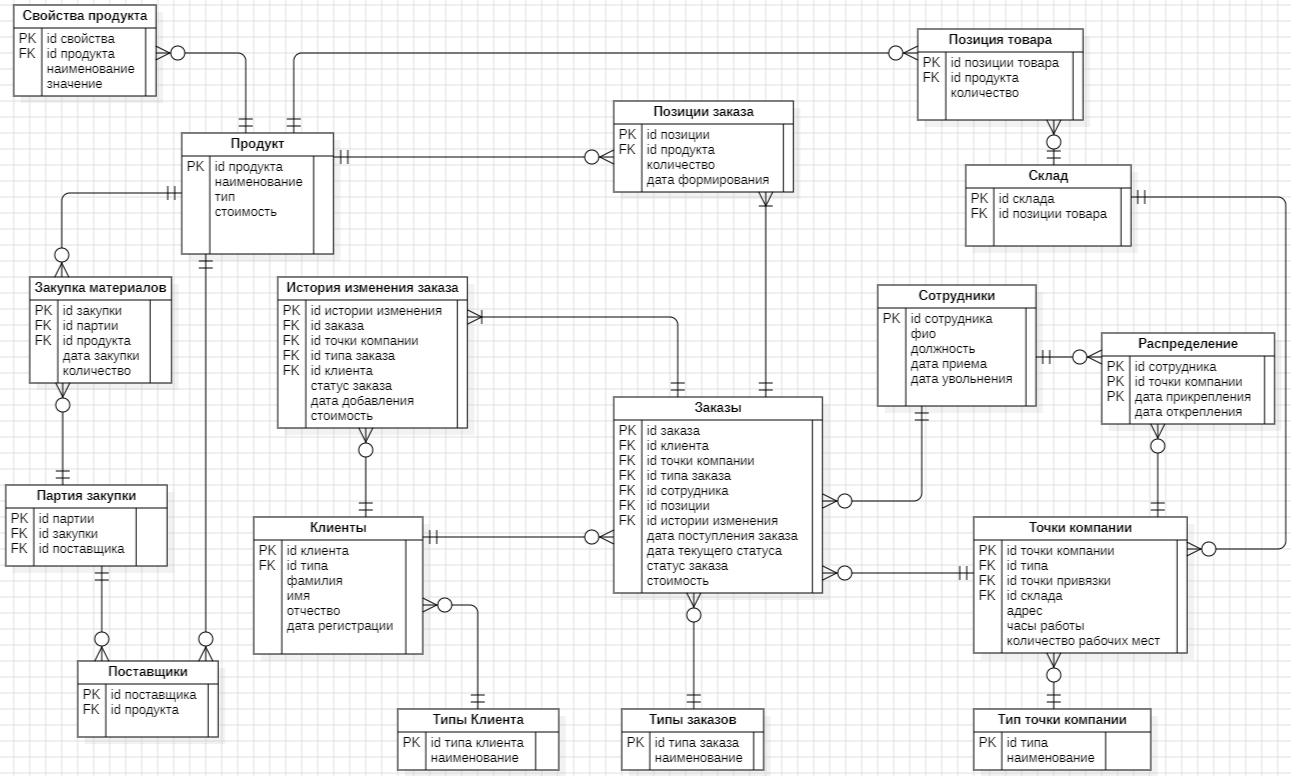


Рис 6 – ER- диаграмма

# **6 Язык UML. Use-case диаграммы. Activity диаграммы**

UML ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Unified Modeling Language* — унифицированный язык моделирования) — язык [графического](https://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%B7%D1%83%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F) описания для [объектного моделирования](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%BE%D0%B5_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5&action=edit&redlink=1) в области [разработки программного обеспечения](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8F), для [моделирования](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) [бизнес-процессов](https://ru.wikipedia.org/wiki/%D0%91%D0%B8%D0%B7%D0%BD%D0%B5%D1%81-%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81), [системного проектирования](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5) и отображения [организационных структур](https://ru.wikipedia.org/wiki/%D0%9E%D1%80%D0%B3%D0%B0%D0%BD%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0).

UML является языком широкого профиля, это — [открытый стандарт](https://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BA%D1%80%D1%8B%D1%82%D1%8B%D0%B9_%D1%81%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82), использующий графические обозначения для создания [абстрактной модели](https://ru.wikipedia.org/wiki/%D0%90%D0%B1%D1%81%D1%82%D1%80%D0%B0%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C) [системы](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0), называемой *UML-моделью*. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML-моделей возможна [генерация кода](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%B4%D0%BE%D0%B3%D0%B5%D0%BD%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D1%8F).

## **6.1 Use-Case диаграммы**

Основное назначение диаграммы — описание функциональности и поведения, позволяющее [заказчику](https://ru.wikipedia.org/wiki/%D0%97%D0%B0%D0%BA%D0%B0%D0%B7%D1%87%D0%B8%D0%BA), [конечному пользователю](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D0%B5%D1%87%D0%BD%D1%8B%D0%B9_%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C) и [разработчику](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%81%D1%82) совместно обсуждать проектируемую или существующую [систему](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_(%D0%BA%D0%BE%D0%BC%D0%BF%D1%8C%D1%8E%D1%82%D0%B5%D1%80)).

При моделировании системы с помощью диаграммы прецедентов [системный аналитик](https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%BD%D1%8B%D0%B9_%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D1%82%D0%B8%D0%BA) стремится:

* чётко отделить систему от её окружения;
* определить действующих лиц (актёров), их взаимодействие с системой и ожидаемую функциональность системы;
* определить в глоссарии предметной области понятия, относящиеся к детальному описанию функциональности системы (то есть прецедентов).

Работа над диаграммой может начаться с текстового описания, полученного при работе с заказчиком. При этом нефункциональные требования (например, конкретный язык или система программирования) при составлении модели прецедентов опускаются (для них составляется другой документ).

Для отражения модели прецедентов на диаграмме используются:

* рамки системы ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) system boundary) — прямоугольник с названием в верхней части и эллипсами (прецедентами) внутри. Часто может быть опущен без потери полезной информации,
* [актор](https://ru.wikipedia.org/wiki/%D0%90%D0%BA%D1%82%D1%91%D1%80_(UML)) (англ. actor) — стилизованный человечек, обозначающий набор ролей пользователя (понимается в широком смысле: человек, внешняя сущность, класс, другая система), взаимодействующего с некоторой сущностью (системой, подсистемой, классом). Акторы не могут быть связаны друг с другом (за исключением отношений обобщения/наследования),
* прецедент — эллипс с надписью, обозначающий выполняемые системой действия (могут включать возможные варианты), приводящие к наблюдаемым акторами результатам. Надпись может быть именем или описанием (с точки зрения акторов) того, «что» делает система (а не «как»). Имя прецедента связано с непрерываемым (атомарным) сценарием — конкретной последовательностью действий, иллюстрирующей поведение. В ходе сценария акторы обмениваются с системой сообщениями. Сценарий может быть приведён на диаграмме прецедентов в виде UML-комментария. С одним прецедентом может быть связано несколько различных сценариев.

Часть дублирующейся информации в модели прецедентов можно устранить указанием связей между прецедентами:

* обобщение прецедента — стрелка с незакрашенным треугольником (треугольник ставится у более общего прецедента),
* включение прецедента — пунктирная стрелка со [стереотипом](https://ru.wikipedia.org/wiki/%D0%A1%D1%82%D0%B5%D1%80%D0%B5%D0%BE%D1%82%D0%B8%D0%BF_(UML)) «include»,
* расширение прецедента — пунктирная стрелка со стереотипом «extend» (стрелка входит в расширяемый прецедент, в дополнительном разделе которого может быть указана точка расширения и, возможно в виде комментария, условие расширения)

При работе с вариантами использования важно помнить несколько простых правил:

* каждый прецедент относится как минимум к одному действующему лицу;
* каждый прецедент имеет инициатора;
* каждый прецедент приводит к соответствующему результату.

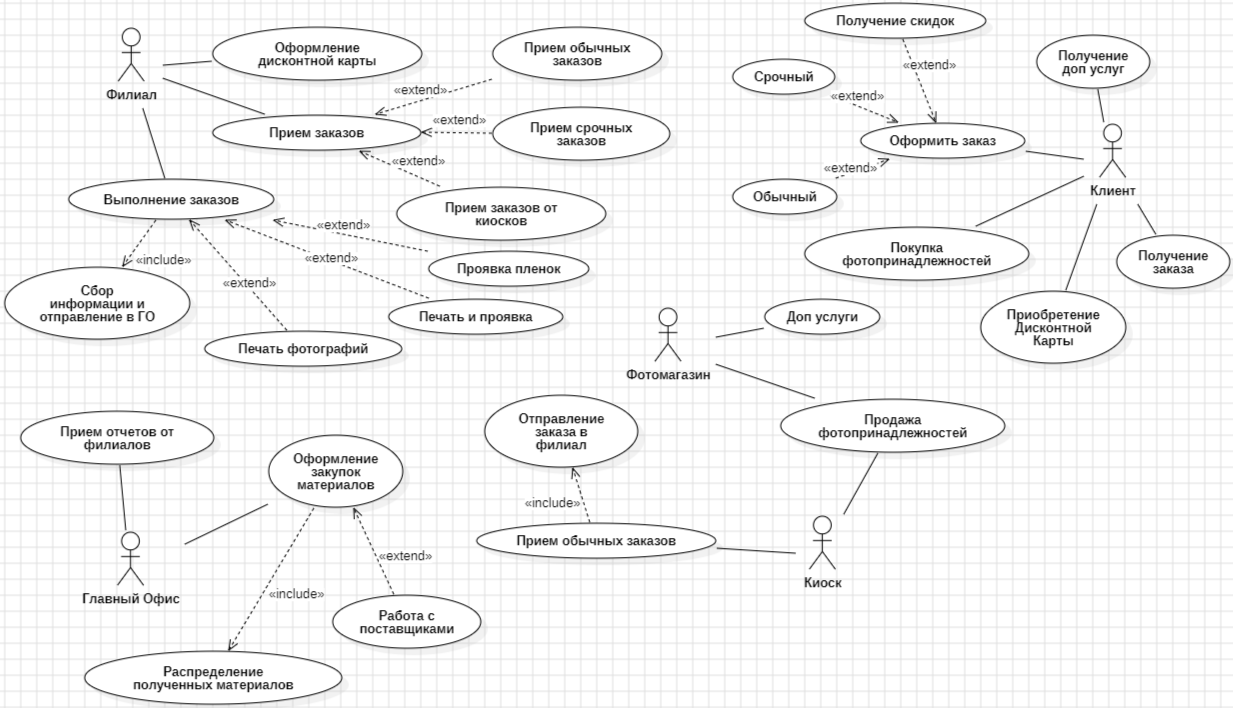


Рис 7 – Use-Case диаграмма

## **6.2 Activity диаграммы**

Диаграмма деятельности ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *activity diagram*) — [UML](https://ru.wikipedia.org/wiki/UML)-диаграмма, на которой показаны действия, состояния которых описано на диаграмме состояний. Под деятельностью ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *activity*) понимается спецификация исполняемого поведения в виде координированного последовательного и параллельного выполнения подчинённых элементов — вложенных видов деятельности и отдельных действий [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *action*, соединённых между собой потоками, которые идут от выходов одного узла ко входам другого.

Диаграммы деятельности используются при моделировании бизнес-процессов, технологических процессов, последовательных и параллельных вычислений.

Диаграммы деятельности состоят из ограниченного количества фигур, соединённых стрелками. Основные фигуры:

* Прямоугольники с закруглениями — действия
* Ромбы — решения
* Широкие полосы — начало (разветвление) и окончание (схождение) ветвления действий
* Чёрный круг — начало процесса (начальный узел)
* Чёрный круг с обводкой — окончание процесса (финальный узел)

Стрелки идут от начала к концу процесса и показывают потоки управления или потоки объектов (данных).

Ниже представлены activity диаграммы для трёх видов деятельности: «Выдача скидочной карты», «Выдача лекарств» и «Отслеживание перемещения лекарств».

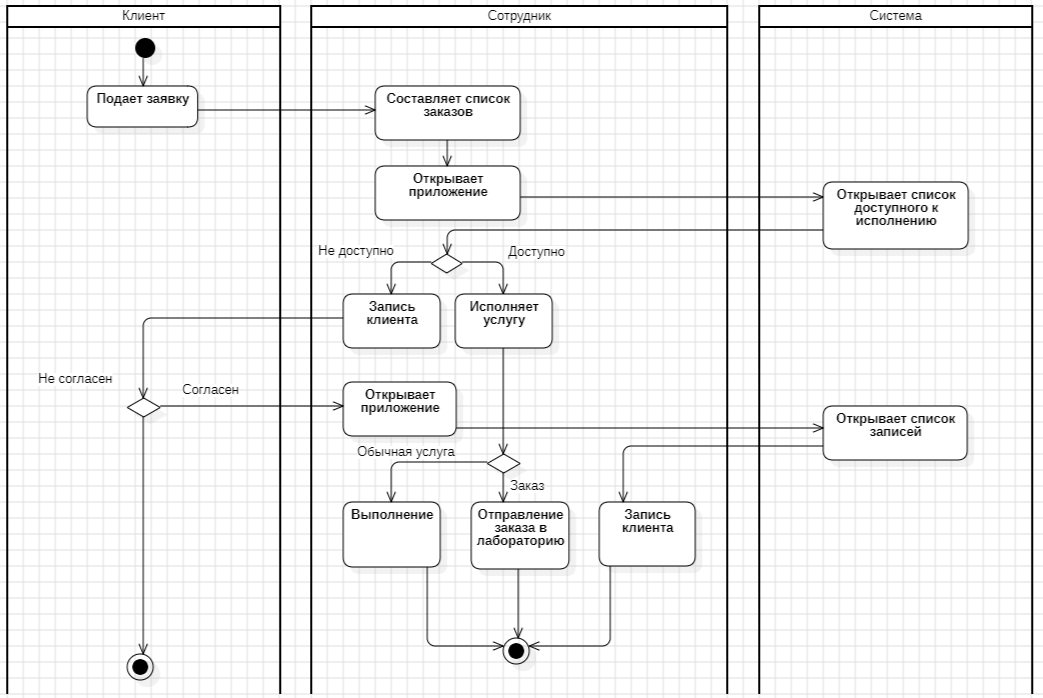


Рис 8 – Activity диаграмма «Приём заявок»



Рис 9 – Activity диаграмма «Формирование заказа материалов»

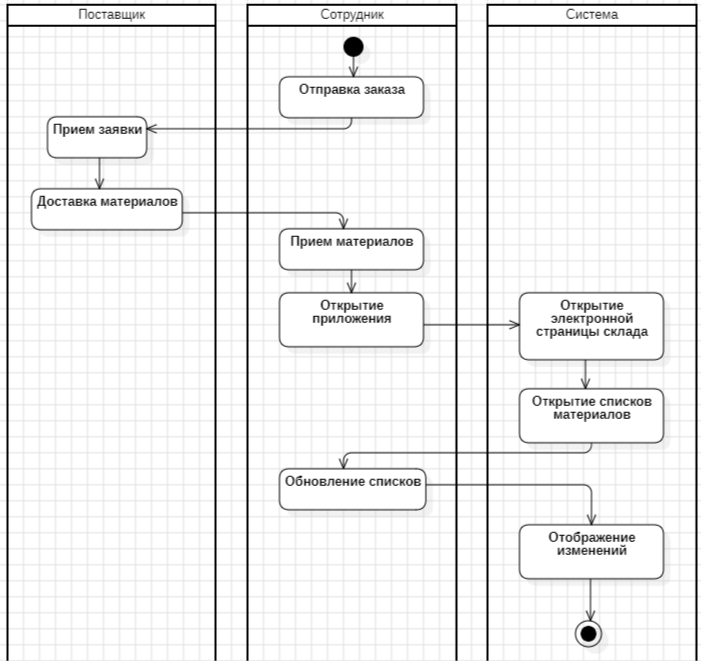


Рис 10 – Activity диаграмма «Закупка и прием материалов»

**6.3 Диаграмма последовательностей**

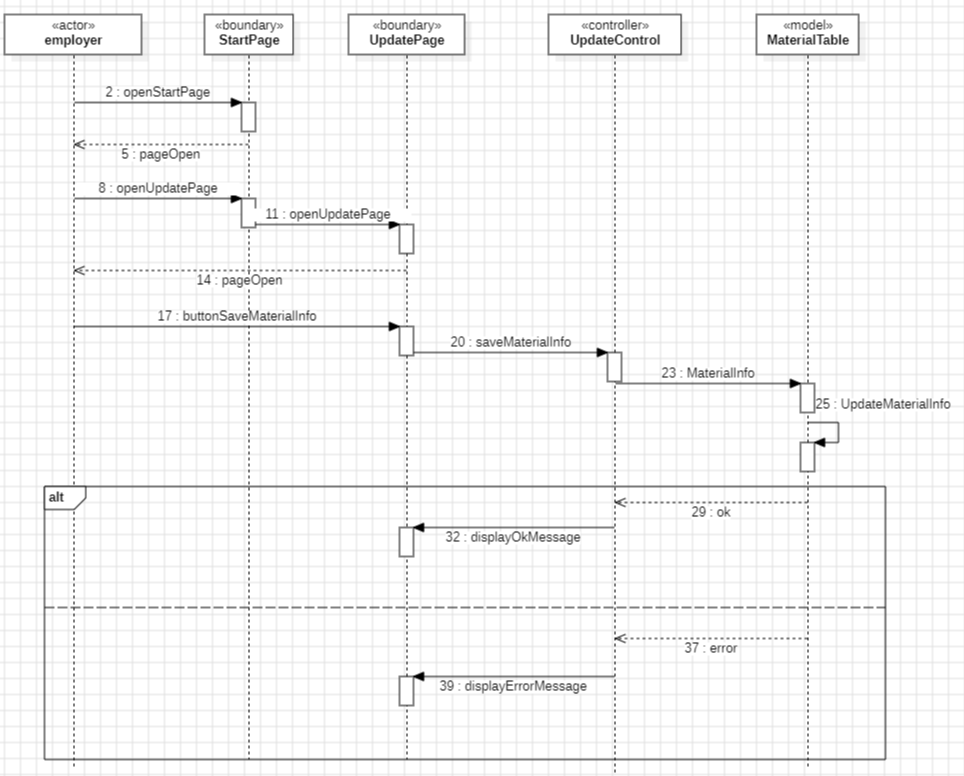


Рис 11 – Диаграмма последовательности «Обновление списков доступных материалов»

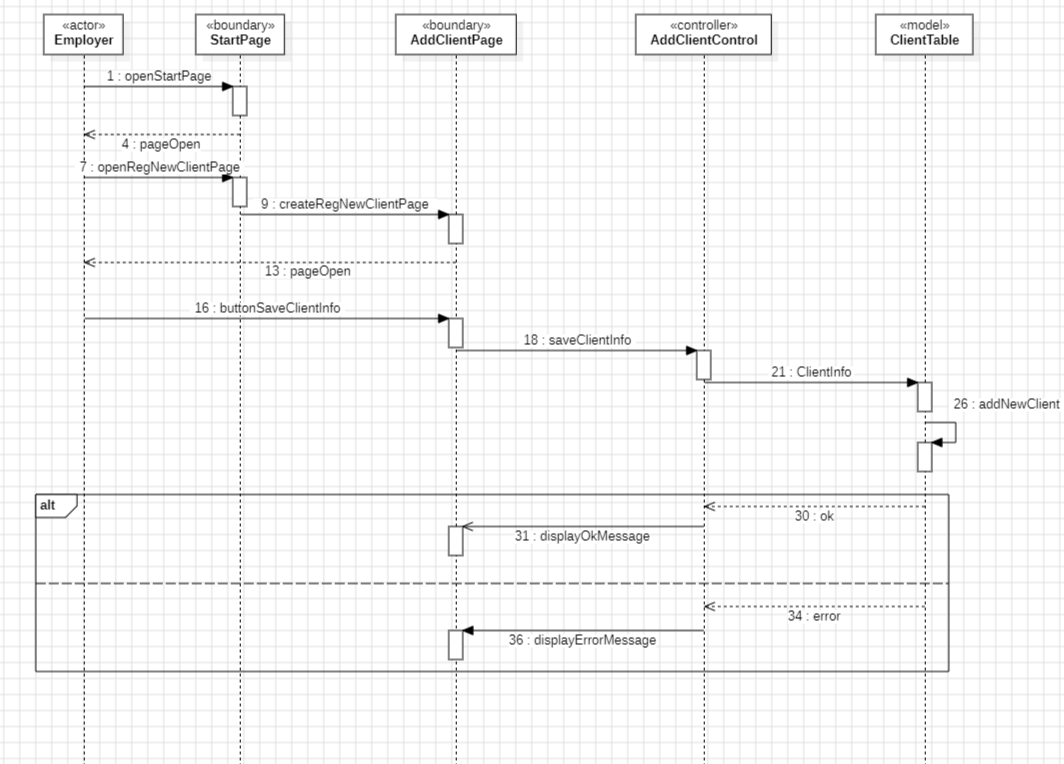


Рис 12- Диаграмма последовательности «Регистрация клиента»

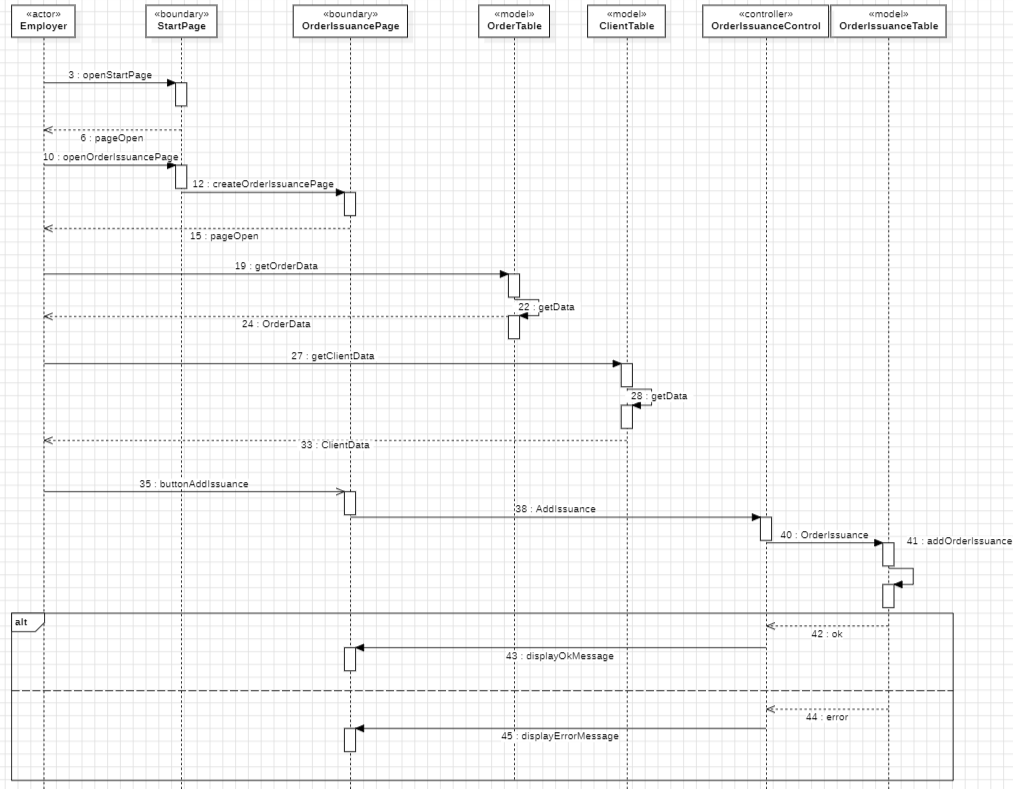


Рис 13 – Диаграмма последовательности «Выдача заказа»

## **8.1 Диаграмма классов**

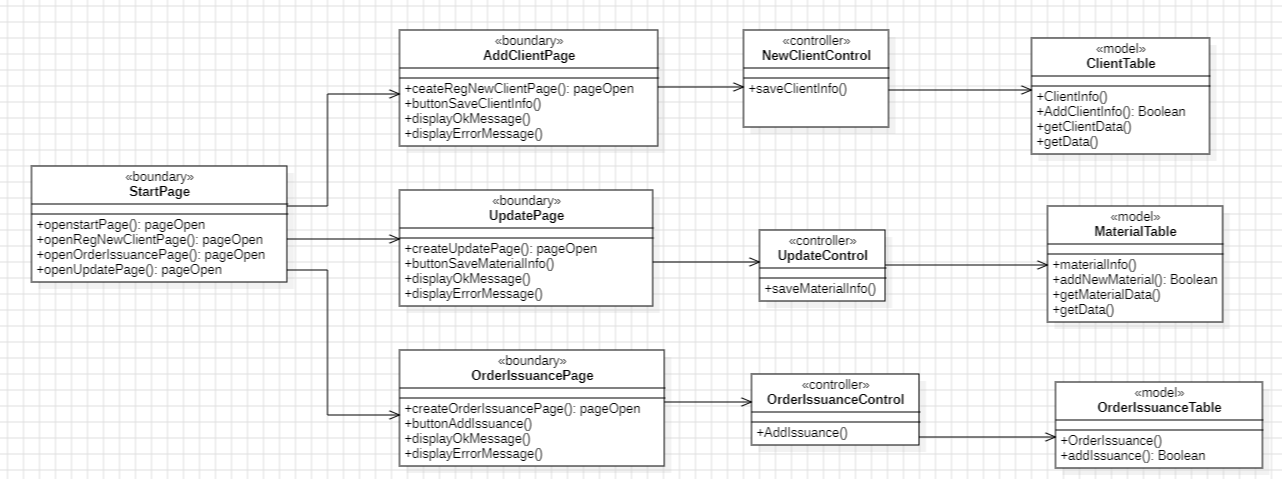


Рис 14 – Диаграмма классов

# **ЗАКЛЮЧЕНИЕ**

Целью работы было создание моделей бизнеса информационной системы. Была исследована деятельность организации и достигнута поставленная цель.

В процессе проделанной работы, получен практический опыт по организации плана работ над проектом, произошло ознакомление с CASE- средствами, помогающими создать план работ по проекту, распределенный во времени.

При проведении анализа предметной области, а также предлагая подход к решению проблем предметной области, было произведено ознакомление с основными методами по созданию моделей методологией IDEF0 и выполнены следующие задачи:

1. Построение функциональной модели бизнес – процессов;
2. Исследование документов и документооборота;
3. Построение структуры БД, объектной модели в системы в виде классов;
4. Построение use-case и activity диаграмм;
5. Описание проделанной работы.