МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**

**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**

**МЕХАНИКИ И ОПТИКИ»**

**ЛАБОРАТОРНАЯ РАБОТА №1**

**«Проектирование распределенной системы»**

по дисциплине: Разработка и внедрение распределенных систем.

Студент Кислюк Игорь Витальевич

Факультет Инфокоммуникационные технологии

Кафедра Программных систем

Группа К4120

Направление (специальность) 11.04.02 Программное обеспечение в инфокоммуникациях

Проверил: Иванов Сергей Евгеньевич, доцент, к.ф.-м.н.

Санкт-Петербург

2018 г.

# 1. Определение целей проекта и предметной области проекта

Система «Служба инкассации» – это новая система, которая заменяет текущие ручные процессы оформления заявок и составления графиков доставки и ручного распределения инкассаторов по машинам. С помощью этого приложения появиться возможность ускорения процесса регистрации клиентов в системе, упрощение процесса создания заявок. Отслеживание, учет и планирование маршрутов путем введения электронной системы упростятся, что позволит снизить нагрузку на персонал и уменьшить возможность ошибок.

# 2. Обзор информационных систем для реализации проекта

В данный момент не так много банковских организаций предоставляют открытую систему для службы инкассации, поэтому необходимость разработки открытой системы службы инкассации возрастает. Ввиду закрытости существующих информационных систем очень сложно проводить анализ и выявление достоинств и недостатков текущих решений. Среди рассмотренных банковских организаций были выбраны «Сбербанк», «Альфа», «Банк СПб». У двух последних система службы инкассации представлена телефонным справочником и необходимо осуществлять заявку вручную. Пример приведен на рисунке 1.

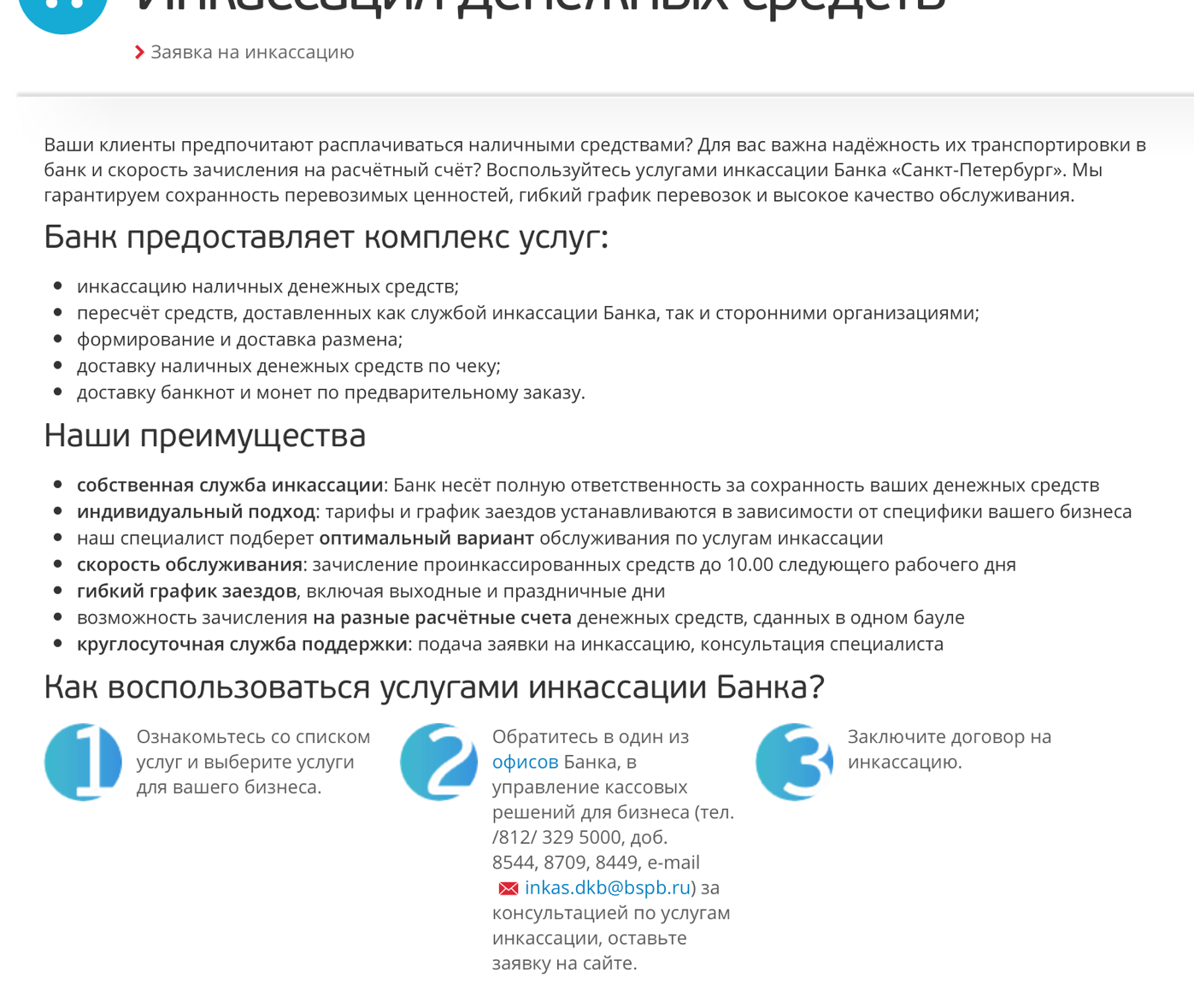


Рисунок 1 – пример службы «Банка СПб»

В случае «Сбербанка» на сайте сразу предоставлена система для ввода необходимых данных для запроса инкассации. Пример приведен на рисунке 2.

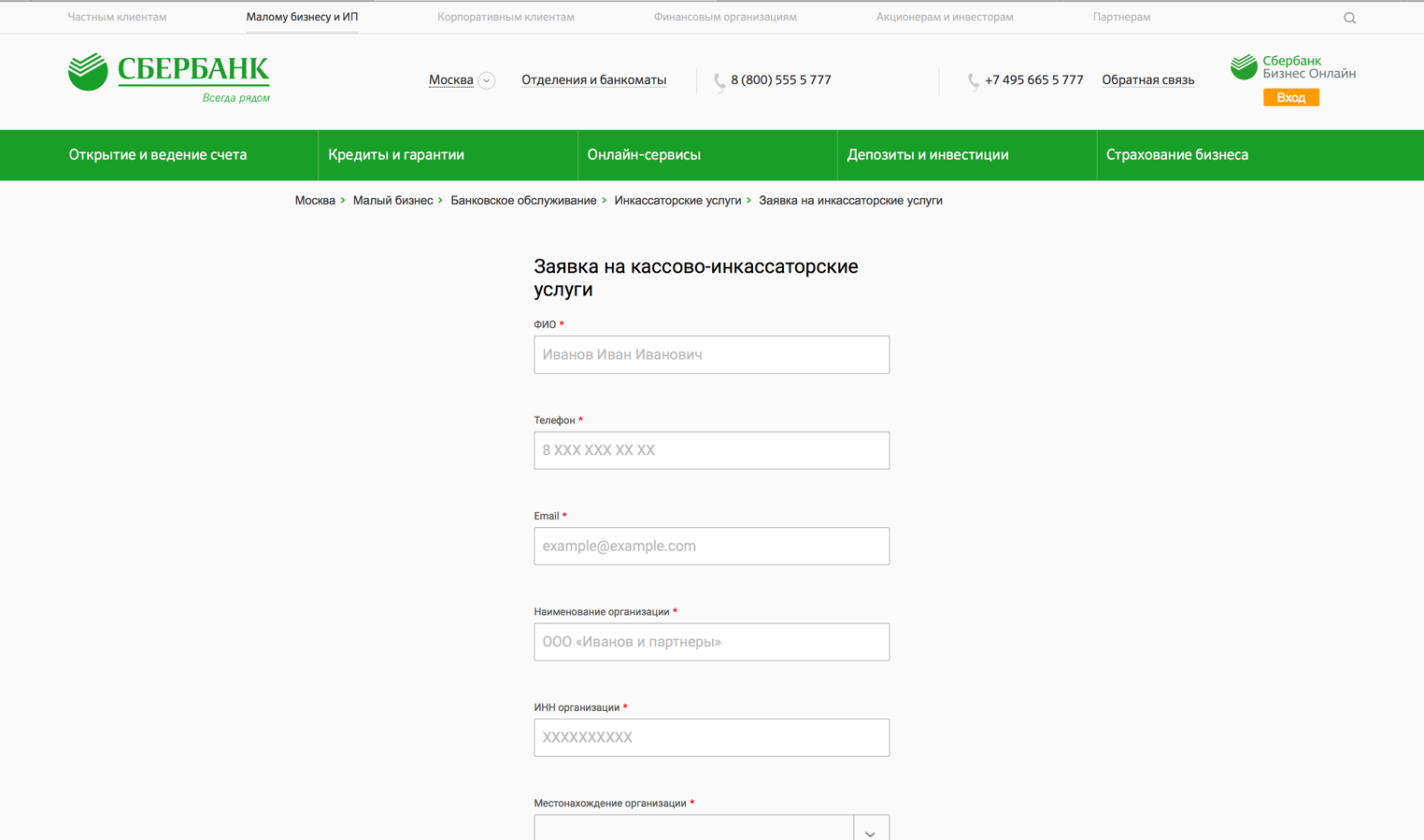


Рисунок 2 – Пример службы от «Сбербанка»

# 3. Определение функциональных и нефункциональных требований

## 3.1. Функциональные требования

Для полноценной работы системы требуется реализовать следующие функциональные требования:

* Система должна предоставлять возможность создания заявки на инкассацию со стороны клиента
* Система должна позволять отслеживать состояние заявки
* Система должна предоставлять возможность регистрации клиентов в системе
* Должна быть возможность добавления объектов со стороны как клиентов, так и сотрудников
* Система должна предоставлять возможность создания и отправки отзывов любым участником процесса
* Система должна позволять пользователям генерировать и отправлять отчеты в удобном им формате

## 3.2. Нефункциональные требования

В информационной системе нефункциональные требования обычно представлены внешними интерфейсами, атрибутами качества и ограничениями системы. Для наиболее эффективной работы информационной системы следует выделить следующие атрибуты качества:

* *Требования к производительности.* Контент, генерируемый системой, должен полностью загружаться за фиксированное время при соблюдении скоростных условий сети доступа. Система должна выводить пользователю сообщение о подтверждении деятельности после того, как пользователь отослал информацию системе.
* *Требования к безопасности.* Обеспечение защиты системы от базовых атак на веб-приложения, использование шифрования по протоколу https. Использование anti-forgery токенов при отправке контента.
* *Требования к доступности.* Система должна быть в постоянном доступе за исключением времени планового обслуживания. Сбои в системе должны логироваться для дальнейшего исследования и устранения.
* *Требования к надежности.* При преждевременном разрыве соединения между пользователем и системой уже внесенная информация не должна нарушать целостность транзакции при повторном подключении. Система должна иметь постоянно обновляющуюся резервную копию БД. Необходимо предусмотреть возможность подключения дополнительных средств развёртывания для сглаживания нагрузки.
* *Требования к визуальному оформлению.* Ограничения дизайна интерфейсов и реализациисистемы должны быть заранее обговорены с заказчиком. Интерфейс должен быть удобен пользователю в любое время: отсутствие необоснованно ярких картинок, резких и интуитивно непонятных переходов и загруженных страниц.
* *Требования к документации.* Документация информационной системы по дизайну, коду и сопровождению должна соответствовать установленным стандартам и пожеланиям заказчика.

# 4. Формирование образа проекта в целом

На основе имеющихся сведений о предметной области проекта и требований к нему можно определить задачи, выполнение которых приведёт к успешному завершению проекта. Информационная система службы инкассации представляет собой распределённую систему обработки пользовательского взаимодействия, позволяющую автоматизировать бизнес-процессы и получить из этого выгоду для всех пользователей системы. Осуществление проекта заключается в реализации нового функционала согласно пожеланиям заказчика. Для этого требуется провести анализ предметной области, направленный на выявление взаимодействующих сущностей и процессов между ними, которые будут соответствовать определённым действиям со стороны всех субъектов системы. Далее следует выбрать оптимальные средства ведения проекта, разработки и тестирования функционала, а также провести начальное внедрение и обеспечение сопровождения информационной системы. После выполнения этих пунктов проект будет считаться завершённым.

# 5. Выполнение анализа бизнес-процессов. Определение сущностей и необходимых атрибутов.

Анализ бизнес-процессов позволяет смоделировать базовое поведение информационной системы, выделяя основные сущности системы (как связь взаимодействующих субъектов), действия, выполняемые этими сущностями, и влияние между ними.

В процесс изучения предметной области выбранной тематики работы, были выявлены следующие существительные: банковская система, банк-эмитент, клиент в виде юридического лица, служба инкассации, дневная выручка, купюры, сумма размена и им подобные.

Для построения логической модели необязательно использовать абсолютно все существительные. Определим сущность для каждого существительного, который будет участвовать в нашей логической модели.

Первой сущностью для системы инкассации будет административный объект, в котором будет проводиться инкассация. Назовем этот объект – Object. Объект будет обладать следующими характеристиками, такими как, описание, адрес, тип данного объекта (например, пользовательский или банкомат), уникальным кодом и прочими характеристиками. Определим следующие сущности.

- Сущность заявки на плановую инкассацию – Order. В бизнес-процессе пользователи могут создавать заявки на инкассацию, в которых указывают необходимый адрес и время, количество размена или доставку каких-либо ценных ресурсов. Например, доставка золота в магазин ювелирных украшений.

- Сущность пользователя для инкассации – User. Представляет наиболее важную часть бизнес-процесса. Пользователями являются абсолютно все участники процесса: непосредственно пользователи, менеджеры, инкассаторы.

- Сущность отзыва о приложении – Appeal. После проведения инкассации или в случае поломки, любой пользователь может оставить жалобу или предложение, а также оценить работу системы.

- Сущность адреса – Address. Представление адреса, которое содержит регион, город, улицу, и прочие необходимые атрибуты для адреса.

Остальные сущности будут представлены в логической модели в DataModeler.

Принцип бизнес-процесса следующий – пользователь регистрируется в приложении, после его до и пытается оформить заявку на инкассацию в свою фирму. Предположим, что этой фирмой будет магазин. Далее ему необходимо иметь номер договора с банком, который будет проводить инкассацию. Вместе с договором банка пользователь получает номер своего объекта. После этого пользователь, авторизовавшись в приложении может добавить свой объект в список объектов и подать заявку на плановую инкассацию этого объекта. Также может подать заявку на временный отказ от плановой инкассации. После этого пользователь может оставить оценку о работе приложения. Схематичное описание сущностей представлено в таблице 1.

Таблица 1 – Сущности, их атрибуты и описание

|  |  |  |
| --- | --- | --- |
| Сущность | Атрибуты | Описание |
| Object | - code  - state  - object\_type  - description  - address\_id  - pass | Сущность объекта, в котором будет проводиться инкассация |
| Order | - id  - object\_code  - name  - phone\_number  - date  - description  - user\_id  - delivery\_attributes\_id | Сущность заказа. Содержит себе важную информацию о времени и месте, также о необходимом количестве размена, если таковое потребуется |
| User | - id  - name  - surname  - passport  … | Сущность пользователя, которая описывает его тип по отношению к системе и прочие характеристики, необходимые для работы системы инкассации |
| Appeal | - id  - user\_id  - type  - description  - rating  - created\_date  - object\_code | Сущность обращения пользователя в жалобой или отзывом в систему |
| Address | - id  - postcode  - city  … | Сущность адреса, которая предоставляет всю информацию по заданному объекту |

# 6. Определение отношений между сущностями, представление их графически.

С целью определения отношений между сущностями воспользуемся матрицей отношений. В таблице 2 представлена матрица отношений между сущностями.

Таблица 2 – матрица отношений между сущностями

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Объект** | **Пользователь** | **Заявка** | **Обращение** | **Адрес** |
| **Объект** |  | **+** | **+** |  | **+** |
| **Пользователь** | **+** |  | **+** | **+** |  |
| **Заявка** | **+** | **+** |  |  | **+** |
| **Обращение** |  | **+** |  |  |  |
| **Адрес** | **+** |  | **+** |  |  |

Данная модель представляет собой сущности, наделённые атрибутами, и взаимосвязанные между собой различными типами связей, которые определяют взаимоотношения различных сущностей.

# 7. Определение атрибутов, которые будут являться уникальными идентификаторами для каждой сущности.

Для каждой сущности определены стандартные уникальные идентификаторы в качестве дополнительного атрибута, значение которого позволит однозначно определить ту или иную реализацию типа сущности. Также наличие уникального идентификатора упростит создание моделей в дальнейшем.

# 8. Определение основной функциональности системы и формирование ключевых функций системы

Для определения основной функциональности системы и формирование ключевых функций системы распишем ключевые функции, определенные пунктом выше, через основную составляющую функциональность.

Таблица 3 – Функционал «Оформление заявки»

|  |  |
| --- | --- |
| **Оформление заявки** | **Описание** |
| Регистрация | Клиент должен иметь возможность зарегистрировать. |
| Добавление желаемого объекта | Клиент должен иметь возможность добавить существующий объект в систему |
| Составление заявки | Должно быть правильно указано необходимое место и время, доступное для доставки размена |
| Отправка заявки | Должна быть проверка заявки на максимальное количество и правильно указанные контактные данные |

Таблица 4 – Функционал «Регистрация объекта»

|  |  |
| --- | --- |
| **Регистрация объекта** | **Описание** |
| Пароль | Система должна уметь проверять совпадение пароля |
| Адрес | Система должна проверять наличие адреса и его корректность |
| Пользователь | Система должна иметь возможность проверить что пользователь, регистрирующий объект является сотрудником |

Таблица 5 – Функционал «Составление отзыва»

|  |  |
| --- | --- |
| **Составление отзыва** | **Описание** |
| Пользователь | Система должна уметь определять пользователя |
| Оценка | Система должна давать выбор пользователю ставить или не ставить оценку |
| Тип отзыва | Система должна давать выбор пользователю между отзывом или жалобой |

Таблица 6 – Функционал «Просмотр отзывов пользователей»

|  |  |
| --- | --- |
| **Просмотр отзывов пользователей** |  |
| Отзывы | Сотрудник должен иметь возможность получить последние отзывы для составления отчета по работе системы |

# 9. Разработка модели информационной системы в виде диаграммы потоков данных

Модель потоков данных основа на инфологической модели и реализуется при помощи диаграммы потоков данных. На диаграмме, представленной на рисунке 3, указаны основные действующие лица системы, порождающие и принимающие потоки данных.

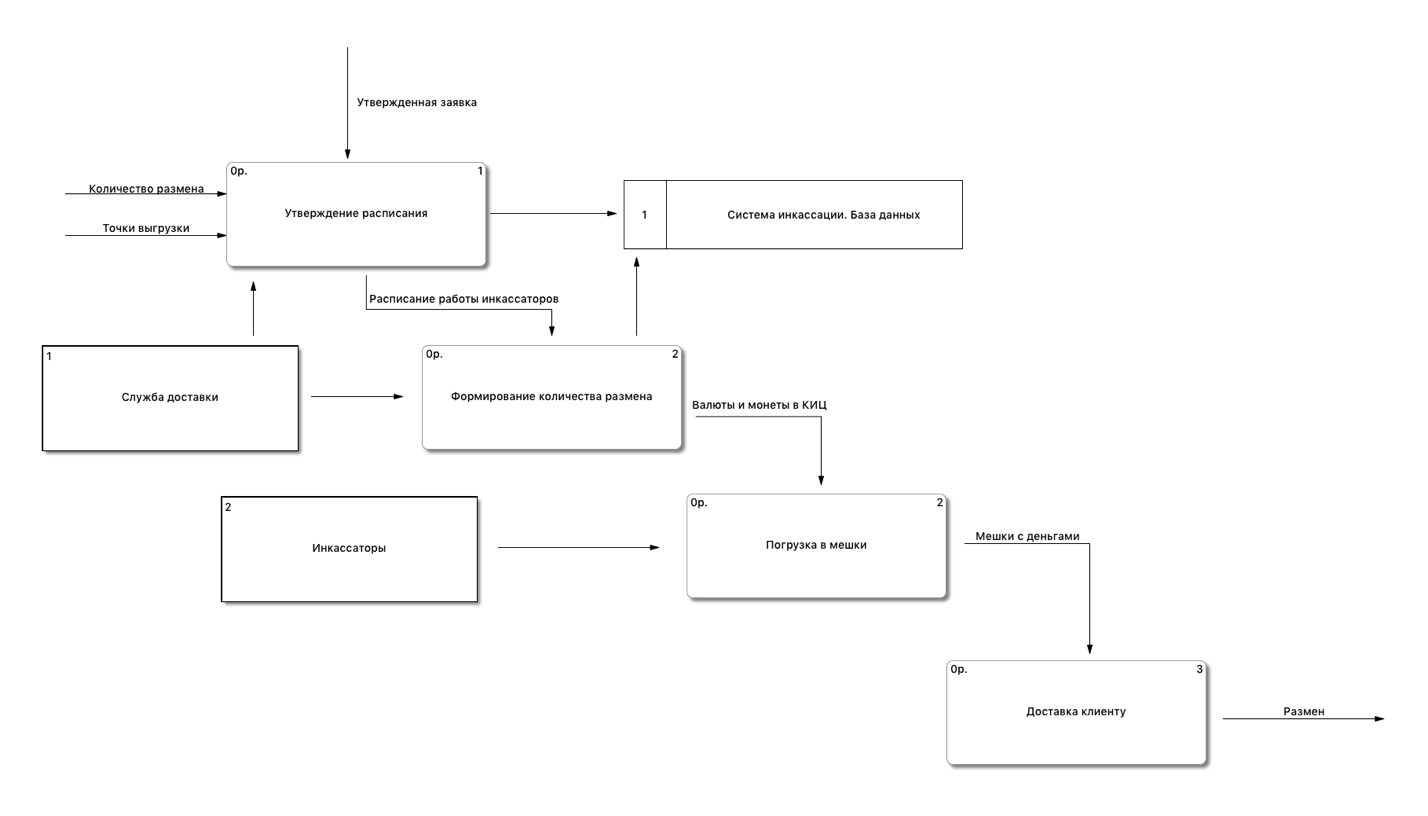
****

Рисунок 3 – Диаграмма потоков данных

# Заключение

В процессе выполнения лабораторной работы по проектированию архитектуры приложения было получено видение приложения, которое решает поставленную перед ним задачу, предоставляет достаточный набор функциональных и нефункциональных требований. Такой вид проектирования позволяет избегать отношения многие-ко-многим, грамотно избегать дублирования сущностей и данных.