Содержание

1. [Введение 2](#_Toc481591807)

[1.1. Актуальность 3](#_Toc481591808)

1.2 Обзор и анализ существующих систем [3](#_Toc481591809)

1.3. Цель работы

[Глава 1](#_Toc481591810)

1. [Анализ и разработка требовании 3](#_Toc481591810)
   1. [Обоснование необходимости разработки 3](#_Toc481591811)
   2. [Анализ требовании 3](#_Toc481591812)
   3. [Функциональные требования 3](#_Toc481591813)

[Глава 2 4](#_Toc481591814)

1. [Конструкторские работы 4](#_Toc481591815)
   1. [Структурный подход 4](#_Toc481591816)
      1. [Диаграмма функций IDEF0 5](#_Toc481591819)

3.1.1.a [Перечень функций в соответствии с функциональными блоками в диаграмме IDEFO 8](#_Toc481591820)

* + 1. Диаграмма потоков данных DFD [11](#_Toc481591821)

3.1.3. Диаграмма сущность связей ERD [11](#_Toc481591822)

[3.2. Объектно-ориентированный подход 11](#_Toc481591824)

[3.2.1. Диаграмма вариантов испольования USECASE 12](#_Toc481591825)

[3.2.2. Диаграмма деятельностей 12](#_Toc481591826)

[3.2.3. Диаграмма классов 12](#_Toc481591827)

[3.2.4. Диаграмма последовательностей 13](#_Toc481591828)

1. [Интерфейс системы 13](#_Toc481591829)

**Введение**

* 1. **Актуальность**

Сейчас почти во всех заведениях общепита используют традиционный метод принятия заказа. Клиент, либо должен прийти и заказать, либо пользоваться услугами предварительного заказа, которые являются не очень удобными: звонить, искать сайт заведения или качать отдельное приложение. Если клиент сделал предварительный заказ через сайт или звонок, у заведения существует риск, что готовый заказ не заберут. В этом случае заведение несет убыток. А качать приложение отдельного заведения согласны лишь 3-5% клиентов.

Клиентам в свою очередь, приходиться стоять в очереди, ждать заказ. Это особенно заметно в часах пик (обеденный перерыв и т.п.), когда в спешке нужно искать заведение сделать и получить заказ – это может занять не мало времени.

Даже если представить, что никаких очередей нет, все равно приходится ждать заказ, и терять драгоценное время, которое можно было потратить на более важные вещи.

Также имеется проблема со сторон заведений, многие из них все еще предпочитают традиционный подход “ручки и бумаги” при принятии заказов, что значит сотрудникам приходится запоминать клиентов на лицо, а это не гарантирует правильной очередь клиентов.

* 1. **Обзор и анализ существующих систем**

Аналогом такого приложения являются:

1. **Мобильное приложение сети кофеен Starbucks’а** запущенное в декабре 2014, позволяет найти ближайшее их заведение, оформить заказ и оплатить. Оно позволило существенно сократить очереди, ускорить обслуживание и повысить эффективность работы заведений в целом. За несколько месяцев 20% всех заказов поступало от пользователей приложения это составляет 9млн$.
2. **Мобильное приложение сети ресторанов Taco Bell** тоже было запушено в 2014 году и примерно через полгода 3,5 млн посетителей этого заведения активно пользовались приложением и она получила положительные отзывы.

Главными недостатками этих приложений является то, что они принадлежат одному заведению. В исследовании проведенным сетью ресторанов OpenTable было установлено, что клиенты предпочли бы установить приложение, в котором собрана информация сразу о нескольких ресторанах.

И в нынешнее время приложение на территории СНГ не имеет аналогов в, есть только приложения для доставки заказов.

* 1. **Цель работы**
* Организация онлайн предварительного заказа блюд заведениям общественного питания.
* Устранение возникновения больших очередей
* Увеличение прибыли
* Экономия времени клиентов

Глава 1   
 Анализ и разработка требовании

* 1. Обоснование необходимости разработки

Учитывая вышесказанные проблемы и недостатки аналогичных систем можно отметить актуальность разработки данного ПО.

* 1. Анализ требовании

Основными требованиями к данной системе являются:

* Просмотр меню и заказ блюд клиентами
* Оплата заказа клиентов через приложение
* Формирование меню заведениями
  1. Функциональные требования

**2.3.1 Клиенты:**

1.Регистрация / авторизация

2.Возможность видеть расположение заведений

3.Возможность просмотра меню выбранного заведения

4.Возможность заказывать ассортименты

5.Возможность оплатить заказ.

6.Возможность получить код заказа после оплаты заказа.

7.Возможность получить уведомление о принятие и оплате заказа.

**2.3.2 Заведение:**

1.Регистрация / авторизация

2.Возможность получить личный кабинет

3.Возможность редактировать личный кабинет

4.Возможность получить уведомление о заказе

5.Подтверждение о выдаче заказа

6.Возможность формировать меню

**2.3.3 Система:**

1.Аудентификация

2.Обработка заявок(заказов)

3.Давать обратную связь потребителям

4.Обработка платежа потребителя

5.Генерация уникального кода заказа

6.Отправка заказа заведению

7.Уведомление об успешной оплате потребителя и заведения

8.Отправить уникальный код заказа потребителю и заведению

9.Деактивировать код после выдачи и подтверждения заказа

Глава 2  
Конструкторские работы.

* 1. **Структурный подход**
     1. Диаграмма функции IDEF0

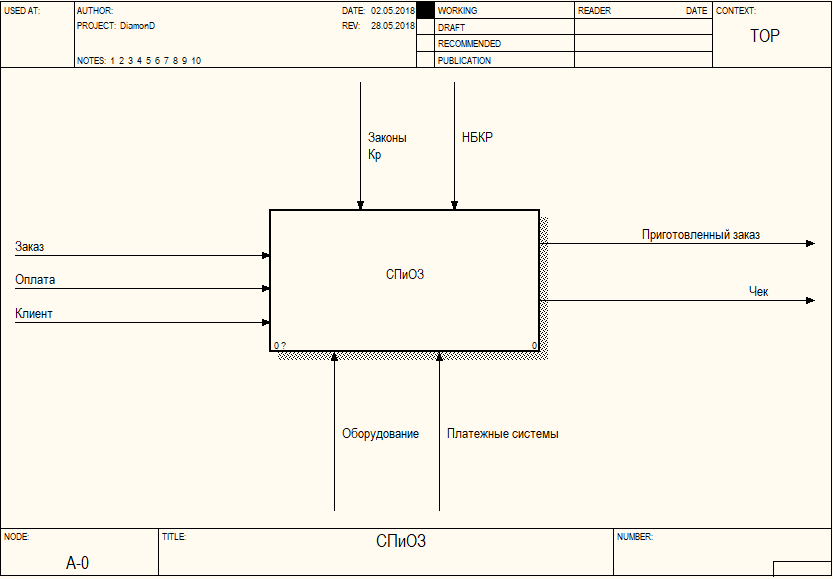


Рис 1. IDEF0 –контекстная диаграмма

**Таблица 1. Описание стрелок**

|  |  |  |
| --- | --- | --- |
| **Название стрелки** | **Тип стрелки** | **Описание** |
| Заказ | Входная | Заказ клиента |
| Оплата | Входная | Оплата клиента за определенный заказ. |
| Клиент | Входная | Данные клиента |
| Законы КР | Управление | Законы и стандарты, указанные государством. |
| НБКР | Управление | Законы и правила Национального Банка Кыргызской Республики |
| Оборудование | Механизм | ПК для заведения или планшет |
| Платежные системы | Механизм | Система для осуществления онлайн оплаты за заказ |
| Чек | Выходная | Выдается чек для клиента |
| Приготовленный заказ | Выходная | Приготовленный заказ заведением |

Декомпозиция первого уровня.

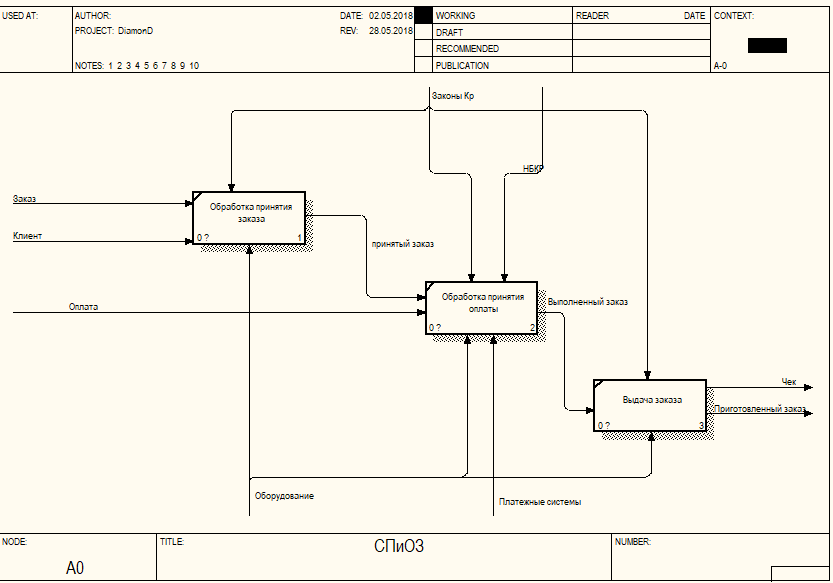


Рис.2 IDEF0-декомпозиция

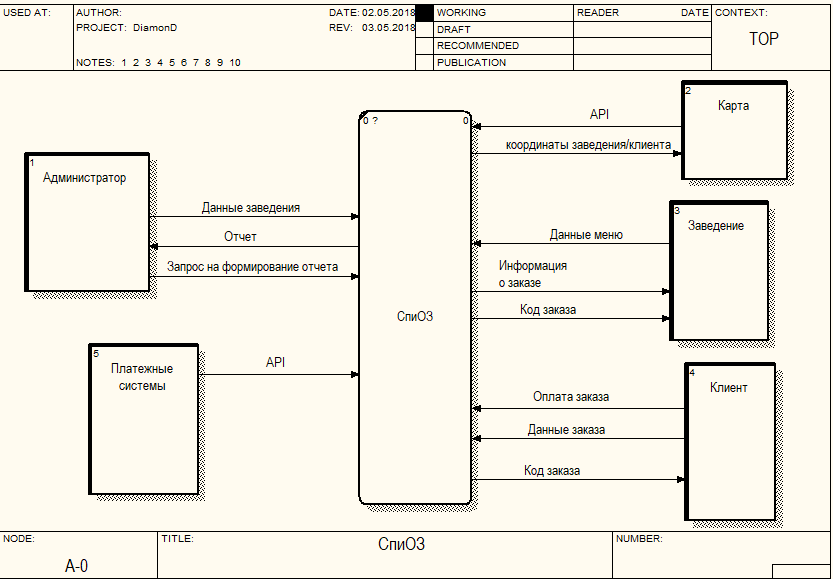
**Таблица 2.**

|  |  |  |
| --- | --- | --- |
| **Название стрелки** | **Тип стрелки** | **Описание** |
| Принятый заказ | Промежуточные(Входная) | Заказ клиента |
| Выполненный заказ | Промежуточные(Входная) | Готовый заказ |

**Таблица 3. Описание процессов.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Название процесса** | **Входные стрелки** | **Выходные стрелки** | **Описание** |
| Обработка принятия заказа | Заказы и клиент | Принятый заказ | Процесс принятия и регистрации заказа |
| Обработка принятия оплаты | Оплата, принятый заказ | Выполненный заказ | Процесс принятия оплаты за заказ |
| Выдача заказа | Выполненный заказ | Приготовленный заказ | Процесс выдачи заказа |

* + 1. Диаграмма потоков данных DFD

Рис.3 Контекстная диаграмма DFD

**Таблица 4. Описание сущностей**

|  |  |  |  |
| --- | --- | --- | --- |
| **Название сущности** | **Входные потоки** | **Выходные потоки** | **Описание** |
| Администратор | Отчет | Данные заведения,  Запрос на формирования отчета | Директор предприятия, может запросить отчёт. |
| Клиент | Код заказа | Данные заказа,  Оплата заказа | Менеджер может вносить все данные о заказе, о комплектующих в систему, а так же изменять их. Запросить отчёт. |
| Заведение | Информация о заказе  Код заказа | Данные меню |  |
| Карта | Координаты заведения/клиента | API |  |
| Платежная система |  | API |  |

**Таблица 5. Описание потоков данных.**

|  |  |  |
| --- | --- | --- |
| **Название** | **Тип** | **Описание** |
| Данные заведения | Входной поток | Администратор вводит данные заведения |
| Запрос на формирование отчета | Входной поток | Данные для отчета |
| Отчет | Выходной поток | Готовый отчет для отчетности филиалов/заведений |
| API(Карта) | Входной поток | Карта используется как модуль (Free Map) |
| Координаты заведений/клиента | Выходной поток | Координаты отправляемые системой в карту |
| Данные меню | Входной поток | Заведение формирует свое меню |
| Информация о заказе | Выходной поток | В заведение поступает данные о заказе |
| Код заказа | Выходной поток | Подтверждающий код заказа |
| API(Платежная система) | Входной поток | API платежных систем используется как модуль системы |
| Данные заказа | Входной поток | Выбранные ассортименты сохраняются в корзину |
| Оплата заказа | Входной поток | Оплата за заказ |
| Код заказа | Выходной поток | Подтверждающий код заказа |

* + 1. Диаграмма сущность связей ERD

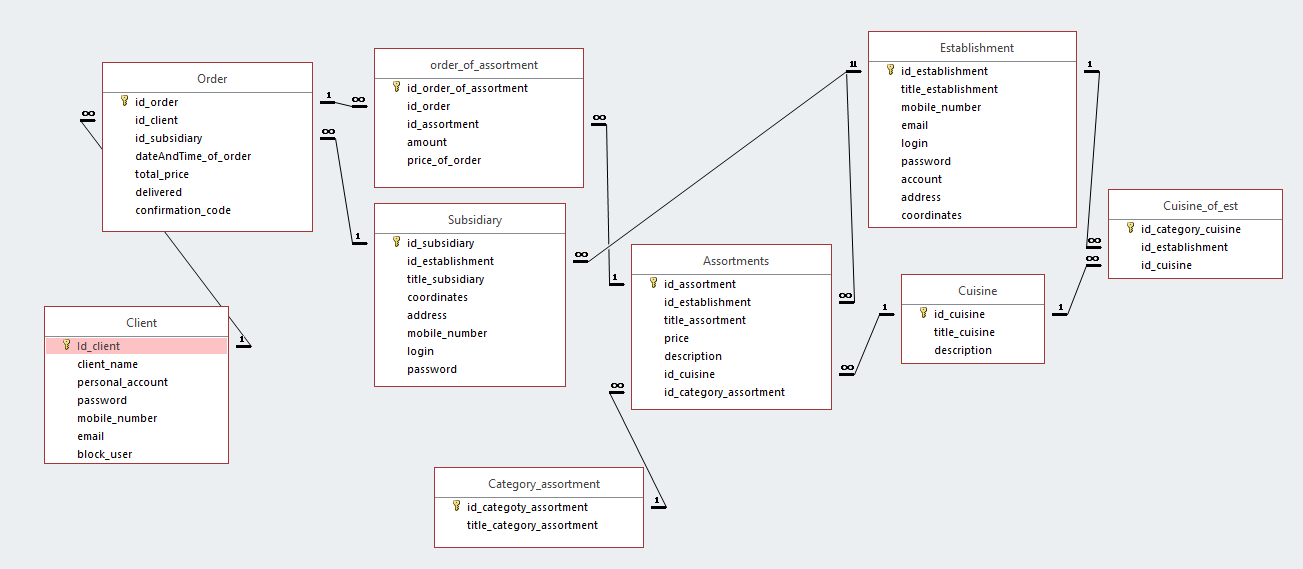


Рис 4. Диаграмма сущность связей ERD

**Таблица 6. Описание сущностей и связей**

|  |  |  |  |
| --- | --- | --- | --- |
| **Название сущности** | **Описание связей** | **Описание сущности** | **Описание атрибута** |
| Клиент | Клиент-Заказ  Клиент может сделать несколько заказов | Хранит данные клиента | id\_client-Целочисленный тип и я является первичным ключом |
| Заказ | Заказ-филиал  Заказ клиента поступает в один из филиалов заведения | Хранит данные всех заказов | Id\_subsidary-вторичный ключ, код филиала.  id\_client – вторичный ключ, код клиента |
| Ассортименты Заказа | Заказ-Детали заказа  В одном заказе может быть несколько блюд, и напитков | Хранит все данные об определенном заказе | id\_order\_assortment- первичный ключ  id\_order-код заказа  id\_assortment-код ассортимента |
| Заведение | Заведение – Филиал: Заведение может иметь несколько филиалов. Заведение – Ассортимент:  Каждое заведение имеет свои ассортименты | Хранит данные о заведении, контактные данные, местоположение, аккаунт и пароль для системы. | login-Аккаунт  password-Пароль  coordinates-местоположение заведения или офиса, для карты.  tittle-establishment –название заведения. |
| Филиал | Заведение – филиал: Заведение может иметь несколько филиалов.  Филиал – заказ: заказ поступает в филиал заведения | Хранит данные филиалов, и к какому заведению они принадлежат. | login-Аккаунт  password-Пароль  coordinates-местоположение филиала, для карты.  id\_establishment – вторичный ключ, код заведения |
| Категория ассортимента | Категория ассортимента-Ассортимент  Ассортимент имеет категорию. Например: Завтрак, салаты, Чизбургеры | Хранит название категории. | **id**\_category\_assortment-первичный ключ, код категория.  title\_category-название категория. |
| Кухня | Кухня-ассортимент:  Каждое блюдо относится к определенной кухне.  Кухня-Кухни заведения:  Каждое заведение имеет свой вид кухни, одну либо несколько. | Хранит название кухонь. |  |
| Кухня\_заведения | Заведение-кухня заведения:  Заведение может иметь несколько кухонь | Хранит кухни заведении. | id-establishment: Код заведения.  id-cuisine: Код кухни |
| Ассортименты | Ассортимент-Заказ ассортимента: в одном заказе может быть несколько ассортиментов. | Хранит данные о каждом ассортименте. |  |

* 1. **Объектно-ориентированный подход**
     1. Диаграмма вариантов использования

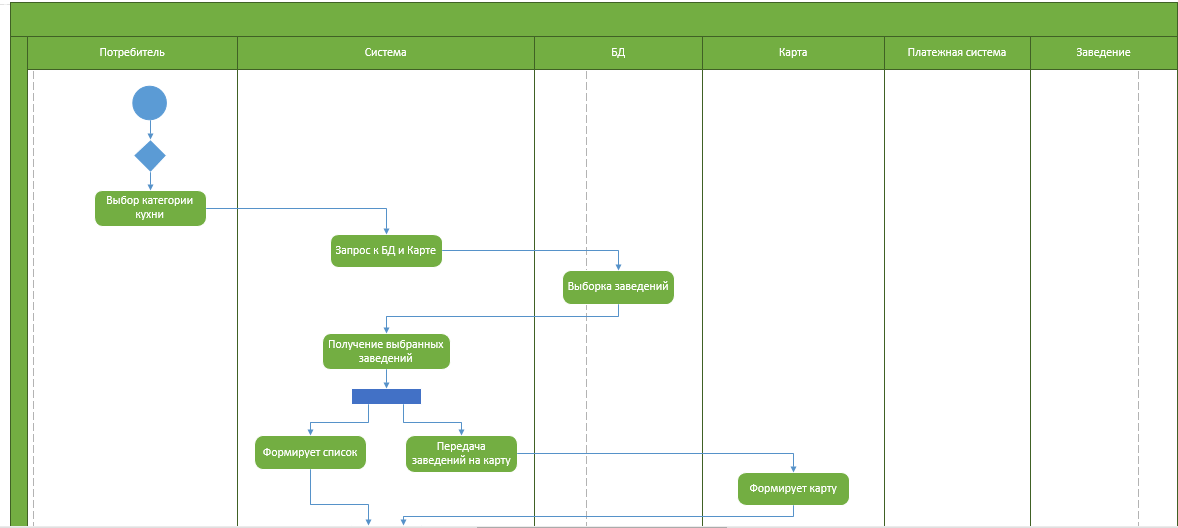
Актеры:

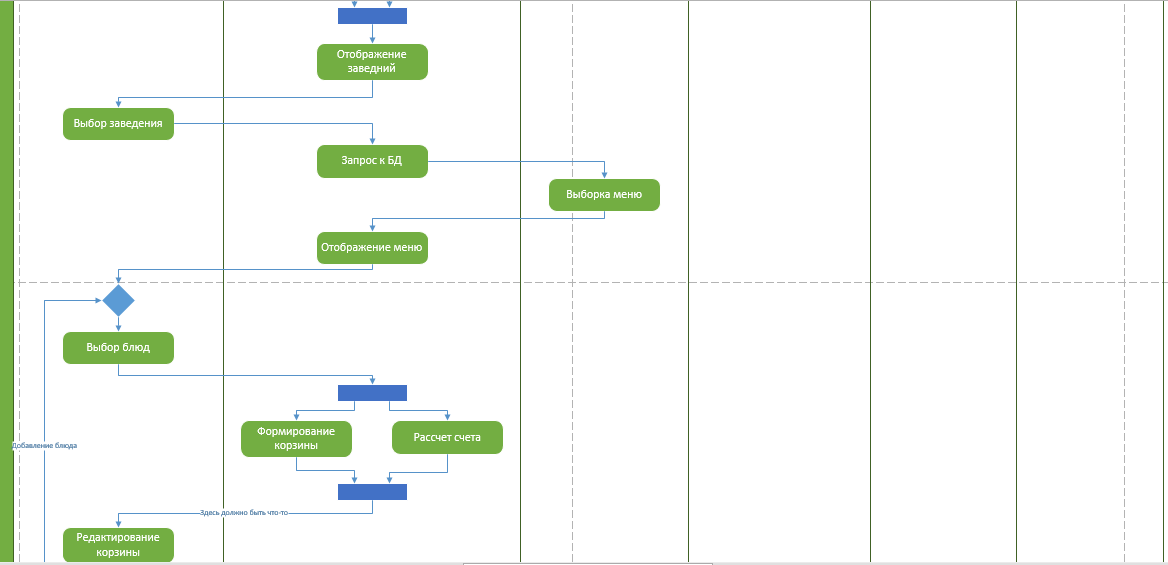
* Клиент
* Заведение
* Карта
* Администратор
* Платежная система

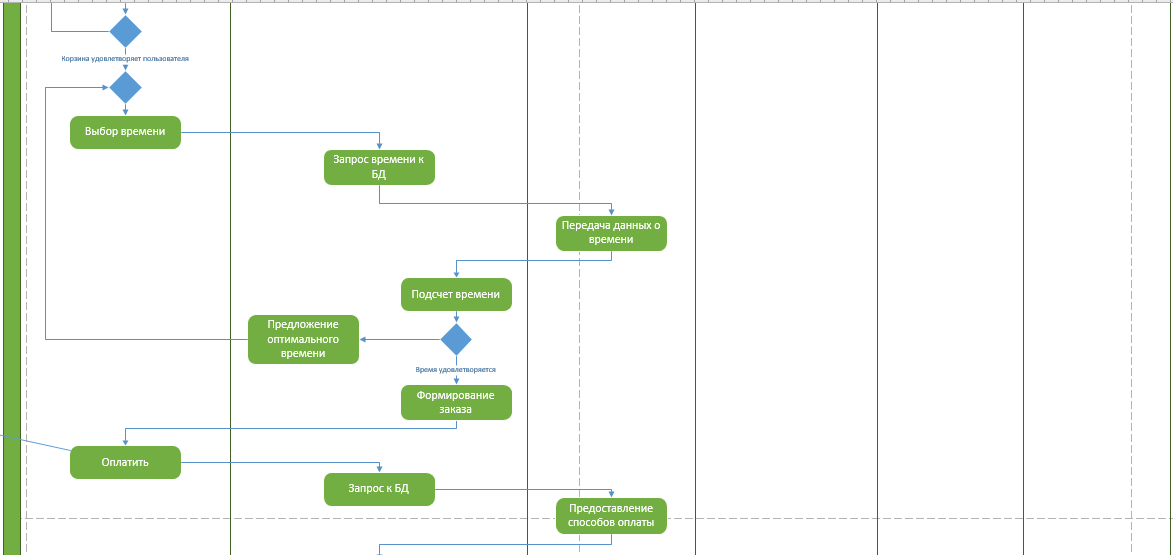


Рис. 5 Диаграмма вариантов использования

* + 1. Диаграмма деятельностей







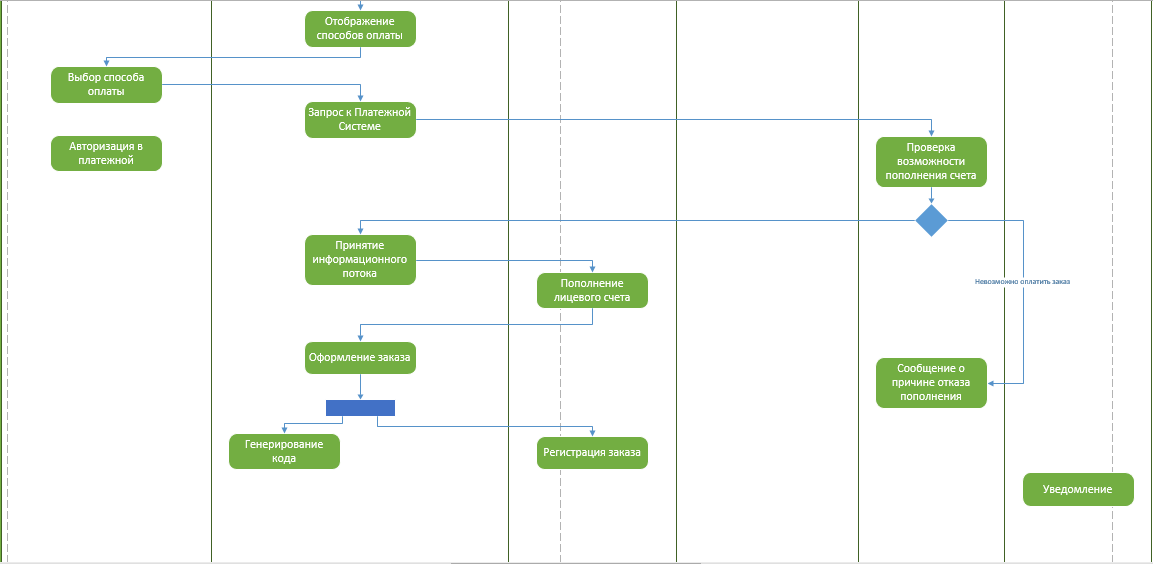


Рис. 6 Диаграмма деятельности

* + 1. Диаграмма классов

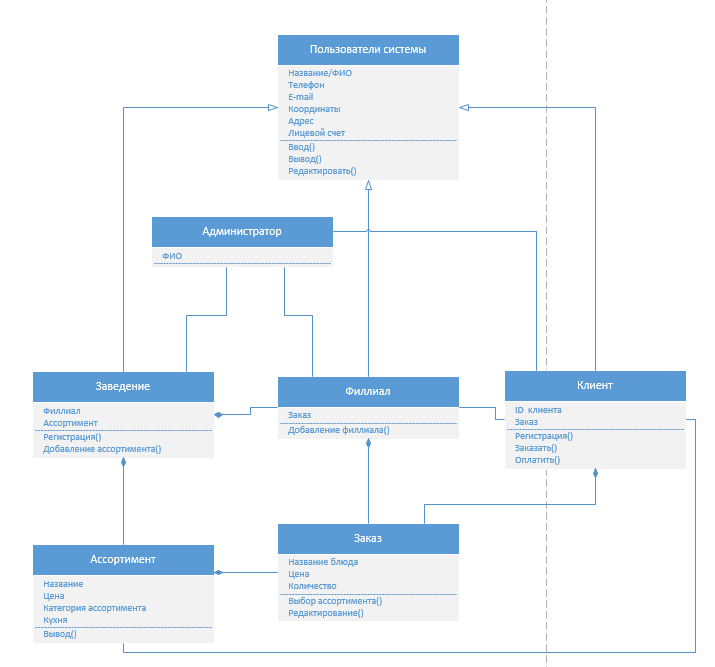


Рис.7 диаграмма классов

* + 1. Диаграмма последовательностей



Рис.8 Диаграмма последовательности заказ ассортимента