|  |
| --- |
| https://www.mirea.ru/bitrix/templates/unlimtech/images/logo.png |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГООБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |

КАФЕДРА ПРИКЛАДНОЙ ИНФОРМАТИКИ (ПИ)

|  |  |
| --- | --- |
| Отчет подисциплине «Проектирование баз данных» | |
| |  | | --- | |  | | |
|  | |
| Выполнил студент | Ляшенко В. А. |
| группа | ИКБО-01-20 |
| Принял | Королев Ф. А. |

Москва 2022

Содержание

[1. Проектирование диаграмм в методологиях IDEF0, IDEF3, DFD в предметной области «Гостиница» 3](#_Toc101130007)

[1.1. Задание 1 3](#_Toc101130008)

[1.2. Выполнение задания 1 3](#_Toc101130009)

[2. Проектирование UML диаграмм предметной области «Гостиница» 13](#_Toc101130010)

[2.1. Задание 2 13](#_Toc101130011)

[2.2. Выполнение задания 1 13](#_Toc101130012)

[Выводы 26](#_Toc101130013)

[Список информационных источников 26](#_Toc101130014)

# 1. Проектирование диаграмм в методологиях IDEF0, IDEF3, DFD в предметной области «Гостиница»

## 1.1. Задание 1

Создать контекстную диаграмму в методологиях IDEF0, IDEF3, DFDв предметной области «Гостиница».

## 1.2. Выполнение задания 1

Гостиничный бизнес - сфера предпринимательской деятельности, целью которой является получение прибыли за счет предоставления услуг размещения гостей в специальных помещениях и определенного сервиса. В его состав могут входить: земельные участки, здания, оборудование, инвентарь, товарный знак, бренд, персонал и др.

Гостиничный бизнес на рынке в условиях высокой конкуренции должен предоставлять лучшие, передовые условия для того, чтобы клиенты выбирали именно данную организацию. Необходимо учитывать нужды клиентов, меняться в зависимости от отзывов и результатов самообследования, реагировать на изменения, например, сезонный приток посетителей или же крупное событие в жизни города (страны).

Экономическая выгода должна получаться превышением доходов от аренды номеров на их обслуживание, предоставлением дополнительных услуг. Причем автоматизация и оптимизация должна производиться для сокращения издержек. Сотрудничество с компаниями-партнерами позволяет увеличить удобство клиентов за счет более развитой инфраструктуры.

Таким образом, рассматривая основной блок диаграммы IDEF0 (Рисунок 1), можно выделить:

* Входные данные – клиенты, пользующиеся гостиницей, плата от них; информация об исследованиях, анализах в данной сфере; поставки, необходимые для функционирования
* Выходные данные – довольные гости и прибыль
* Управлением являются бюджет, устав гостиницы, законодательство страны
* К механизму системы относятся сотрудники, поставщики, помещение

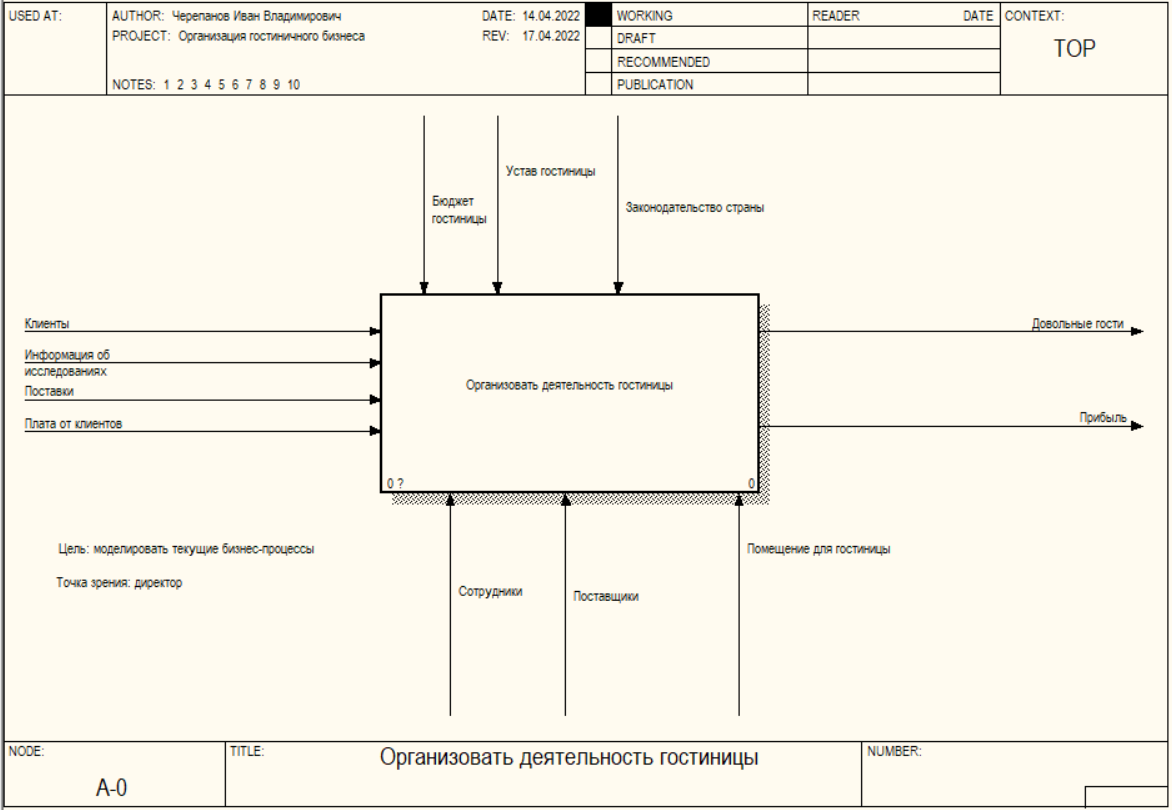


Рисунок 1 – Диаграмма IDEF0

Данную диаграмму можно декомпозировать на следующие активности (Рисунок 2):

* Осуществить управление гостиницей
* Предоставить гостиничные услуги
* Предоставить дополнительные услуги

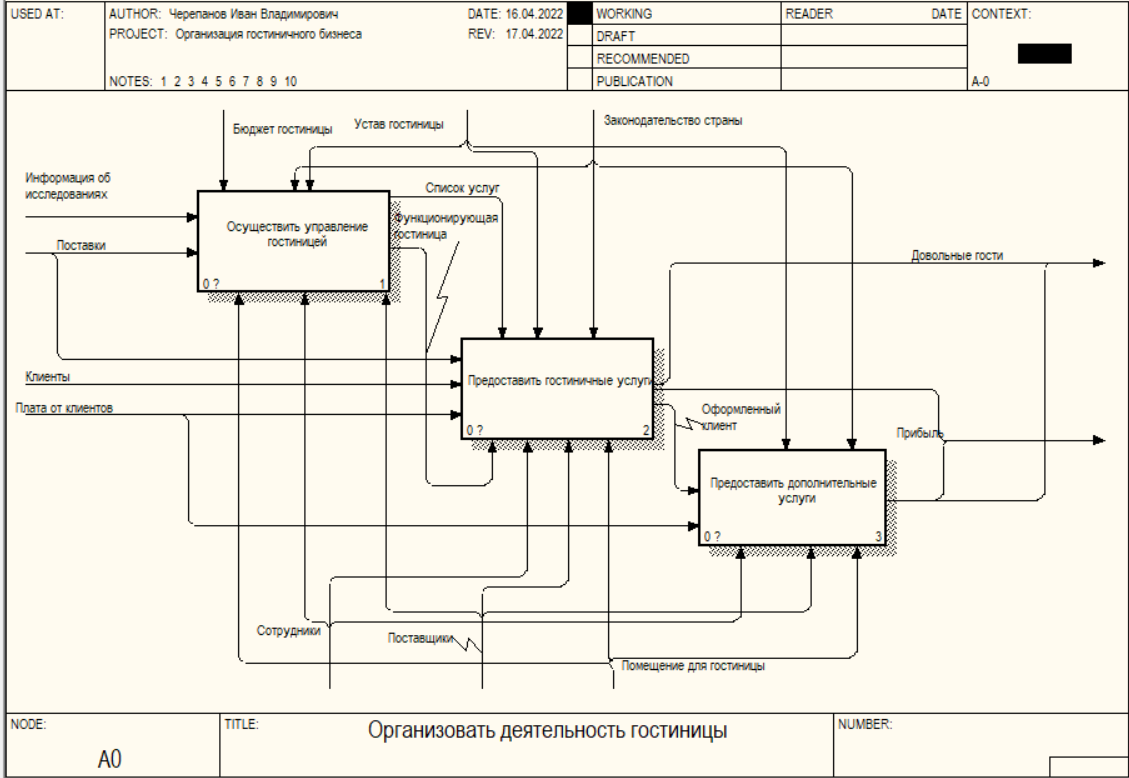


Рисунок 2 – Декомпозиция основного блока

Рассмотрим каждый из блоков по отдельности для лучшего понимания:

Декомпозиция A1 методологии IDEF0: Осуществить управление гостиницей

Для осуществления работы гостиницы необходимо грамотно и правильно осуществлять управление. Для этого необходимо отвечать требованиям времени, нуждам клиентов, для чего нужен блок «Исследование рынка гостиниц», в рамках которого получаем актуальный примерный список услуг и перечень товаров к поставкам. Блок «Составить список услуг» дает сам список и перечень товаров, необходимых для их осуществления. Что уже поступает в качестве управления для поставок во блок «Обеспечить поставки для гостиницы». Стоит отдельно выделить блок «Обеспечить охрану гостиницы», т.к данный блок важен для всей деятельности и хорошего настроя клиентов. Таким образом, разбиения представляет собой 4 пункта:

* Исследование рынка гостиниц
* Составить список услуг
* Обеспечить поставки для гостиницы
* Обеспечить охрану гостиницы

Результат моделирования представлен ниже (Рисунок 3).

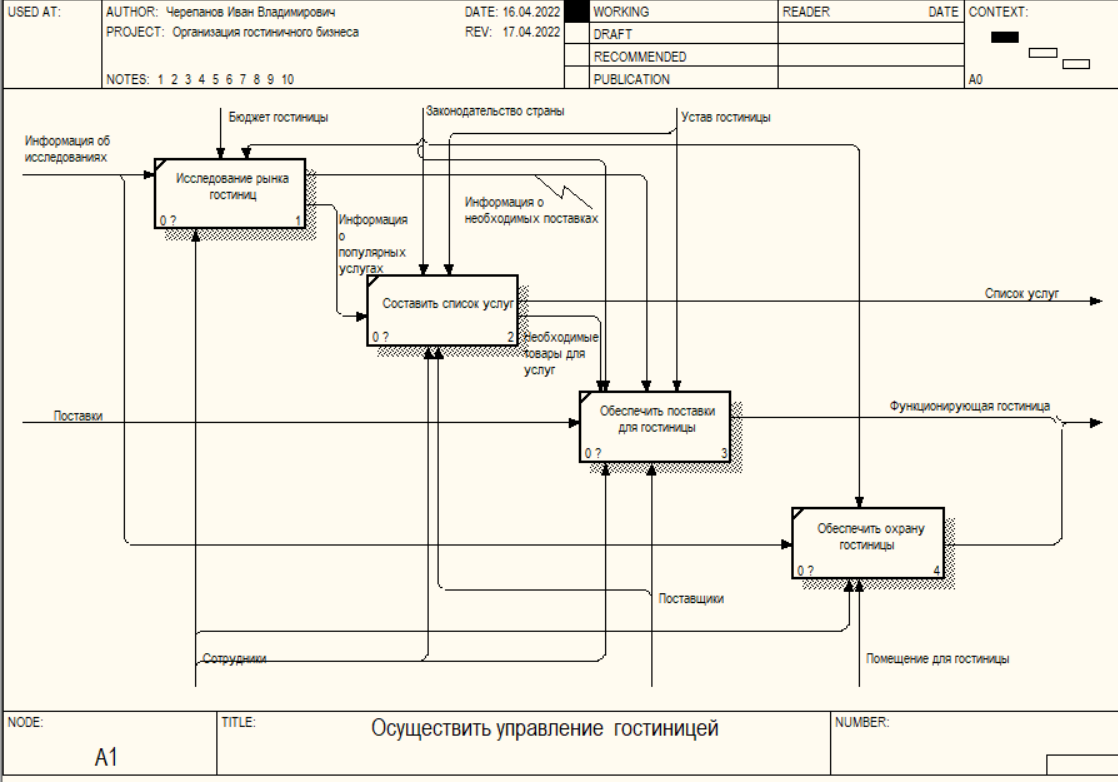


Рисунок 3 – Декомпозиция блока «Осуществить управление гостиницей»

Декомпозиция А2 в методологии IDEF0: Предоставить гостиничные услуги

Предоставление гостиничных услуг довольно размытый процесс, который может включать в себя множество этапов. Для более детального рассмотрения произведем декомпозицию на 3, после декомпозируем уже дочернюю. Получим следующие блоки:

* Предоставить номер
* Организовать питание
* Организовать уборку

Это разбиение помогает рассмотреть классические процессы, которые должны быть так или иначе представлены в гостинице (хостелы не включаем в разбор). Результат представим в виде диаграммы (Рисунок 4).

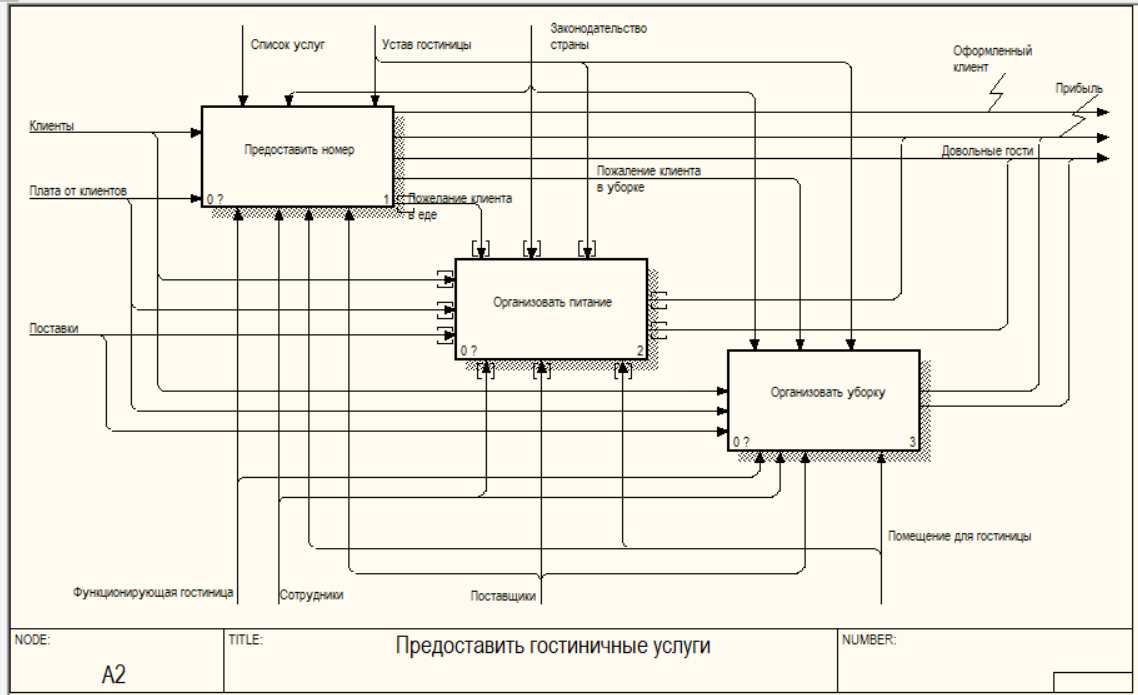


Рисунок 4 – Декомпозиция блока «Предоставить гостиничные услуги» в методологии IDEF0

Декомпозиция А21 в методологии IDEF0: Предоставить номер (Рисунок 5).

Сначала нужно/можно забронировать номер, после чего или сразу, или с течением времени перейти к оформлению, куда будут включены такие моменты, как заполнение документов, предварительные пожелания о времени уборки, питании. После происходит оплата, а затем поселение, а с течением времени и выселение.

Поэтому произведем разбиение на 5 этапов:

* Бронировать номер
* Оформить клиента
* Принять оплату
* Поселить клиента
* Выселить клиента

Результат представим в виде диаграммы (Рисунок 5)

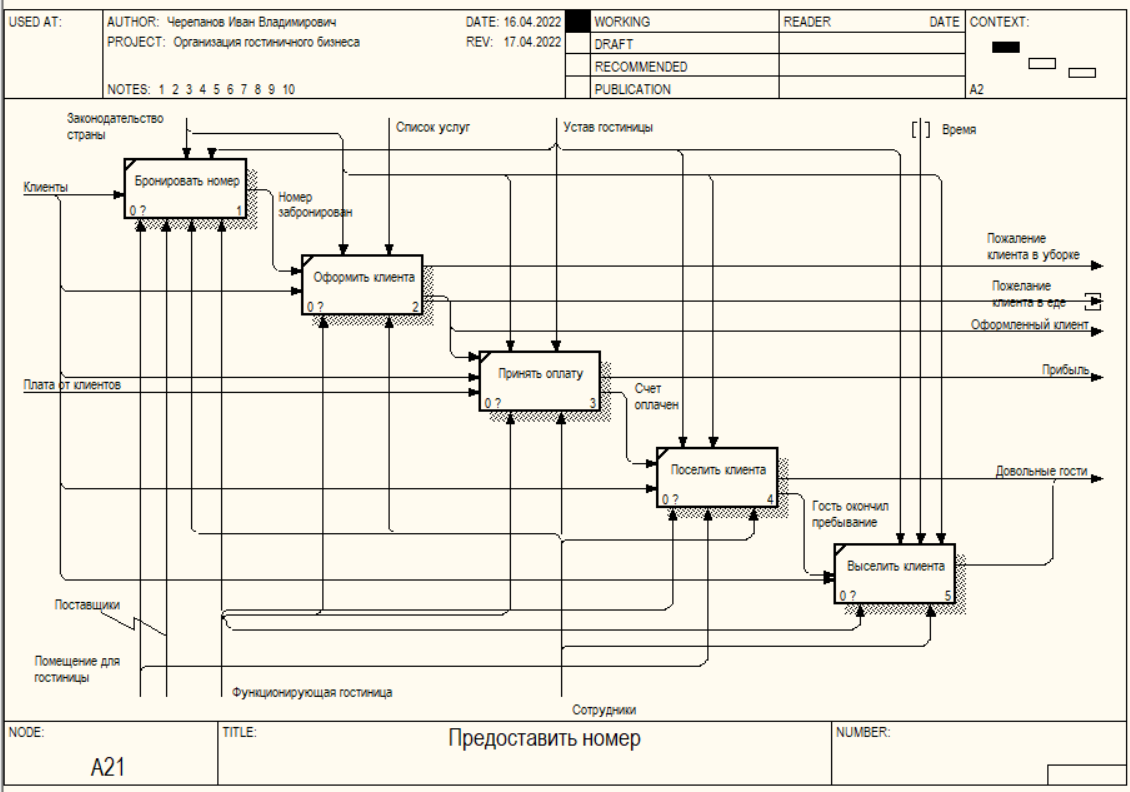


Рисунок 5 – Декомпозиция блока «Предоставить номер» в методологии IDEF0

Декомпозиция А22 в методологии DFD: Организовать питание

Только зарегистрированные клиенты могут заказать еду. Данный заказ принимается, обслуживается сотрудником, кроме того, плату принимает также сотрудник. Для приготовления используется обращение к поставщикам для готовой еды/ингредиентов для готовки в помещении. Таким образом, в данной диаграмме представлены (Рисунок 6):

* Внешние сущности: пользователи, менеджеры, технические консультанты.
* Хранилища: хранилища реквизитов пользователей для входа, общее хранилище пользователей.
* Потоки данных, описывающие движение объектов из одной части системы в другую.

Сам блок декомпозирован на 3 активности:

* Принять заказ на еду
* Подать заказ
* Принять плату

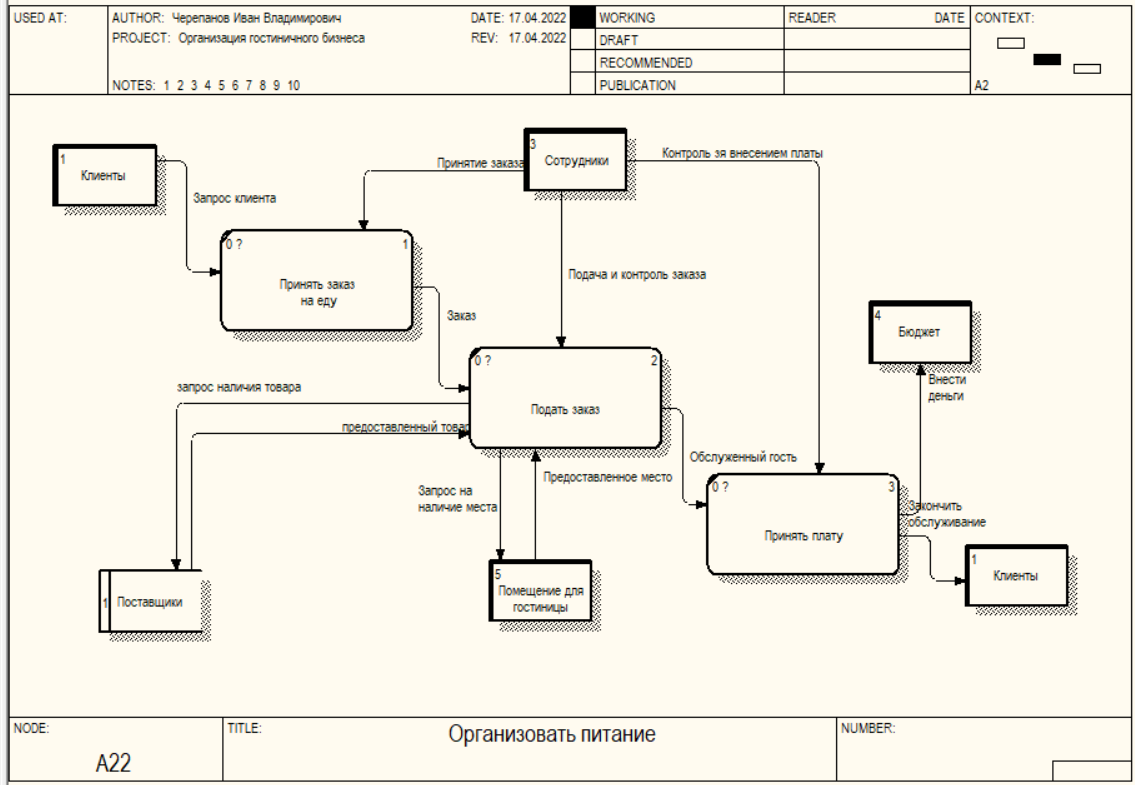


Рисунок 6 – Декомпозиция блока «Организовать питание» в методологии DFD

Декомпозиция А23 в методологии IDEF0: Организовать уборку

Сама функция похожа по многом на «Организовать питание». Принимается от клиента заказ на уборку (возможно только время, удобное для этого или специальный вызов), затем проводится уборка и взимается плата, либо же может быть внесена в отдельный чек. Выделим 3 пункта (Рисунок 7):

* Принять заказ об уборке
* Провести уборку
* Принять плату за уборку

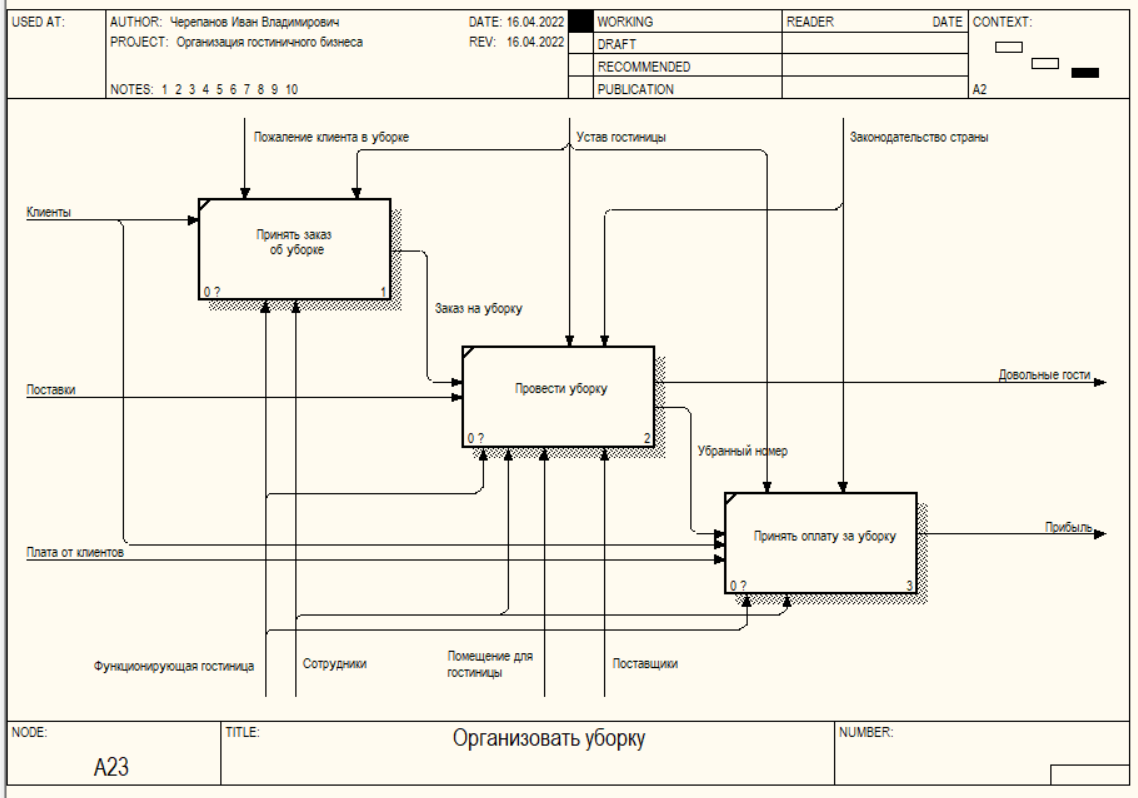


Рисунок 7 – Декомпозиция блока «Организовать уборку» в методологии IDEF0

Декомпозиция А3.1 в методологии IDEF3: Предоставить дополнительные услуги

Необходимо предоставить дополнительные услуги, список которых получаем из предыдущего пункта, так же участвуют сотрудники и опционально помещение для гостиницы (если услуга оказывается на стороне дружественной компании). Таким образом, данный блок представлен на диаграмме IDEF3 (Рисунок 8).

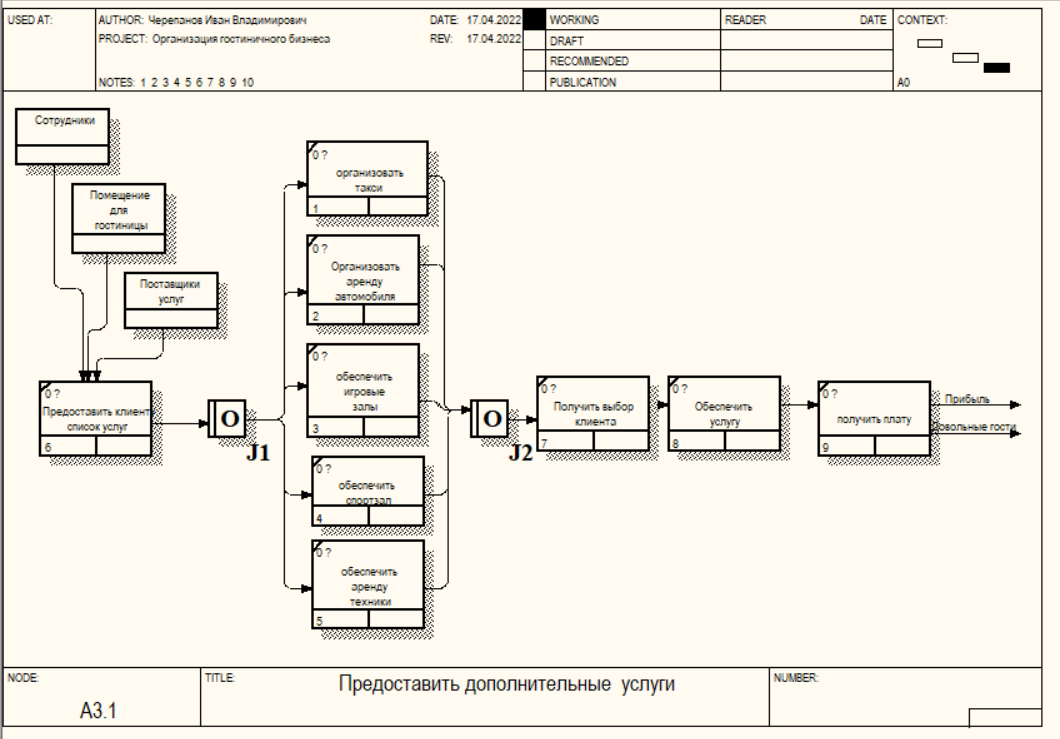


Рисунок 8 – Декомпозиция блока «Предоставить дополнительные услуги» в методологии IDEF3

# 2. Проектирование UML диаграмм предметной области «Гостиница»

## 2.1. Задание 2

Создать UMLдиаграммы по предметной области «Гостиница».

## 2.2. Выполнение задания 1

Диаграмма 1: диаграмма вариантов использования

Диаграмма вариантов использования — это графическое изображение возможных взаимодействий пользователя с системой. Моделируя работы гостиницы, в качестве системы мы подразумеваем гостиницу, а в качестве пользователя – клиента. Пользователь при взаимодействии с системой после авторизации (рассмотрено на следующих диаграммах) может забронировать номер, куда включен выбор номера, пожеланию по питанию, уборке, личные данные, затем оформить въезд, куда обязательно включено оплата и выбор услуг, которые не являются обязательными (такси, игровые залы, спортзал, аренда техники, аренда автомобиля), далее въезд и выезд из номера. За каждым важным шагом следит специализированный сотрудник (Рисунок 9)

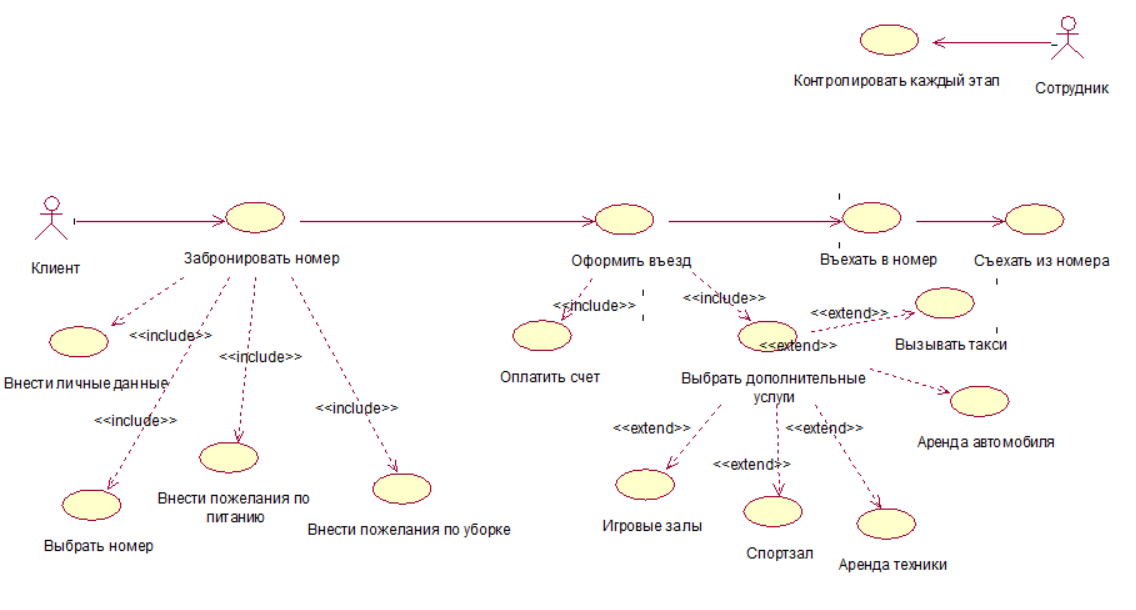


Рисунок 9 – Диаграмма прецедентов

Диаграмма 2: Диаграмма классов

При моделировании данной диаграммы необходимо выделить основные сущности системы, определить их свойства и методы. Опишем объекты системы следующим образом, выделив их операции и атрибуты (Рисунок 10):

* Интерфейс гостиницы
* Показать форму бронирования() – метод выдачи формы бронирования
* Представить список дополнительных услуг() – метод выдачи списка дополнительных услуг
* Показать варианты питания() – метод, отображающий для пользователя варианты питания
* Показать варианты уборки() – метод, отображающий для пользователя варианты уборки
* Оформить въезд() – метод для оформления въезда в гостиницу
* Оформить выезд() – метод для оформления выезда из гостиницы
  + Запрос к БД
  + Текст запроса: String
  + Создать запрос() – создание нового запроса к БД
  + Закрыть запрос() – закрытие запроса к БД
  + Карточка бронирования
  + id: Long
  + Список основных услуг: List<Maintenance>
  + Список дополнительных услуг: List<Maintenance>
  + Плата: int
  + Дата приезда: Data
  + Дата выезда: Data
  + Ввести личные данные() – ввод данных о госте
  + Выбрать условия основных услуг() – выбор основных услуг
  + Выбрать условия дополнительных услуг() – выбор дополнительных услуг
  + Выбрать номер() – метод для выбора номера из доступных
  + IСервер гостиницы
  + Осуществить вход() – производит вход в интерфейс заявки
  + Создать новую заявку() – создание заявки
  + Зарегистрировать пользователя() – производит регистрацию заявки
  + Продолжить заполнение заявки() – метод позволяет продолжить заполнять существующую забронированную заявку
  + Оформить полную заявку() – отмечает заявку как законченную
  + Создать запрос к БД() – создает новый запрос к базе данных
  + База данных гостиницы
  + Получить список доступных номеров() – метод возвращает список доступных для заселения номеров
  + Получить условия услуг() – метод возвращает список условий для основных услуг
  + Получить список дополнительных услуг()– метод возвращает список условий для основных услуг
  + Обновить информацию() – обновление информации в БД
  + Получить информацию о наличии товара() – метод проверяет наличие товара/услуги в БД
  + Получить заявку() – метод возвращает заявку для редактирования/заполнения
  + Интерфейс сотрудника
  + Проверить этап() – проверка заполненности данных на наличие ошибок (в том числе и программных)

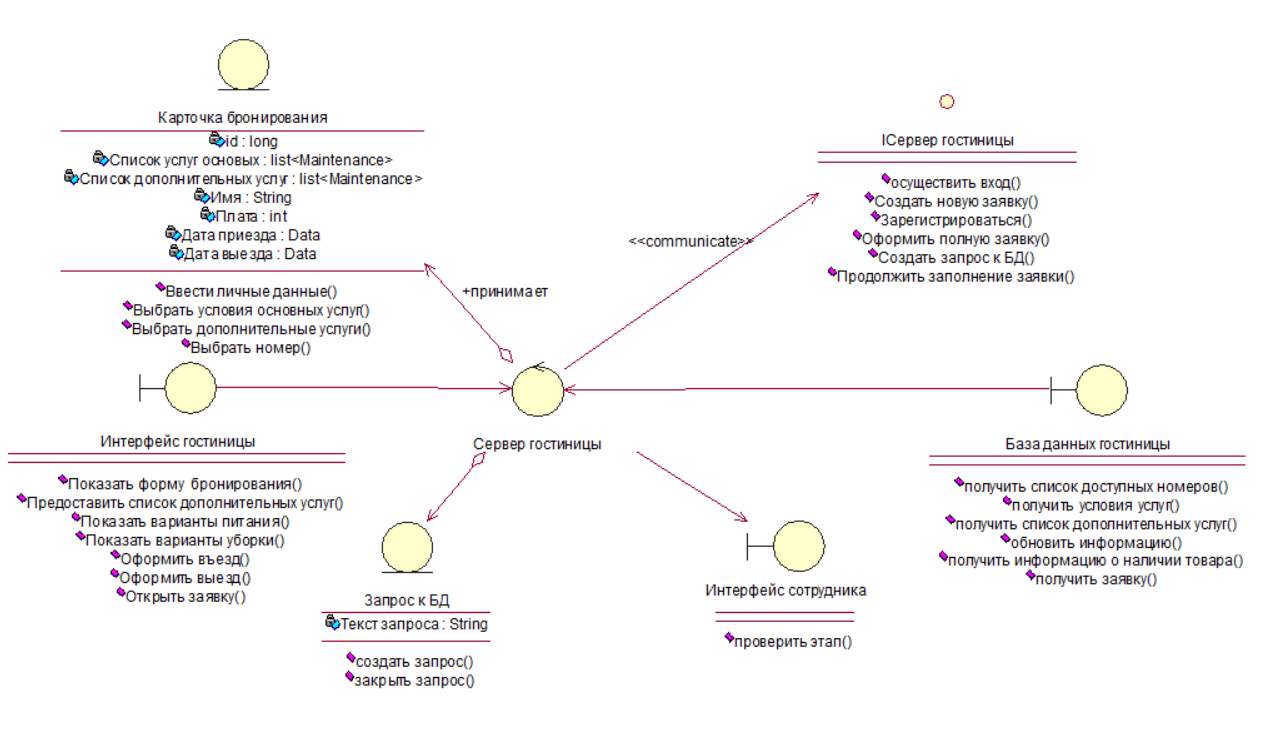


Рисунок 10 – Диаграмма классов

Диаграмма 3: Диаграмма кооперации

Кооперация (collaboration) служит для обозначениямножества взаимодействующих с определенной целью объектов в общем контексте моделируемой системы. Цель самой кооперации состоит в том, чтобы специфицировать особенности реализации отдельных наиболее значимых операций в системе. Кооперация определяет структуру поведения системы в терминах взаимодействия участников этой кооперации. Эти диаграммы отображают поведение конкретного варианта использования и определения роли каждого объекта.

Пользователь делает запрос на новую заявку, происходит создание и выдача пользователю, который вводит первоначальную информацию, необходимую для бронирования. Происходит первичная регистрация заявки. Данный этап может быть пропущен, если бронирование было проведено ранее. Тогда происходит переход сразу к заполнению.

Выполняется запрос к БД, при наличии заявки открывается форма дополнения заявки и создается новый запрос для доступа к списку комнат, услуг, доп.услуг, обновлении информации. Второй запрос завершается. Происходит заполнение от пользователя: номер, услуги, доп.услуги. При завершении регистрации заявки происходит проверка сотрудником. Далее возможен въезд, где данные сверяются с документами сотрудником, так же при выезде происходит проверка.

Данную диаграмму представим в Rational Rose как и последующие (Рисунок 11).

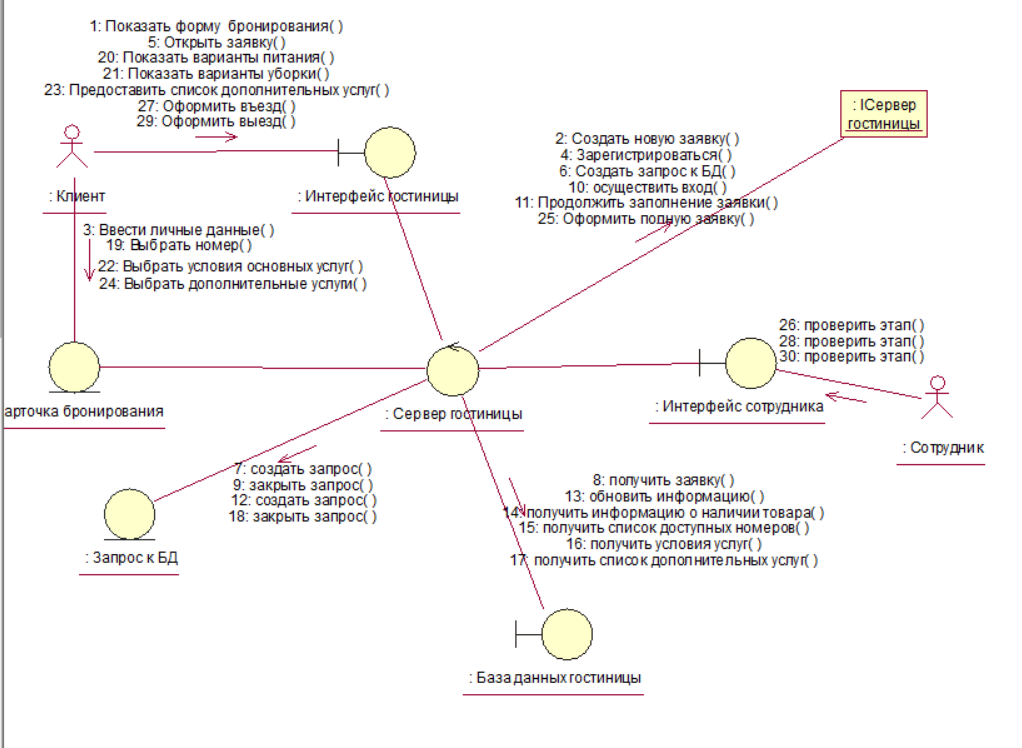


Рисунок 11 – Кооперативная диаграмма

Диаграмма 4: Диаграмма последовательности

Диаграмма последовательности — этот тип диаграммы описывает поведенческие аспекты системы во времени (отображают взаимодействие объектов в динамике, т.е. временные аспекты передачи и приемы сообщений). В данном случае диаграмма совпадает с кооперативной диаграммой и позволяет увидеть связь между объектами и количество бизнес-слоев (Рисунок 12).

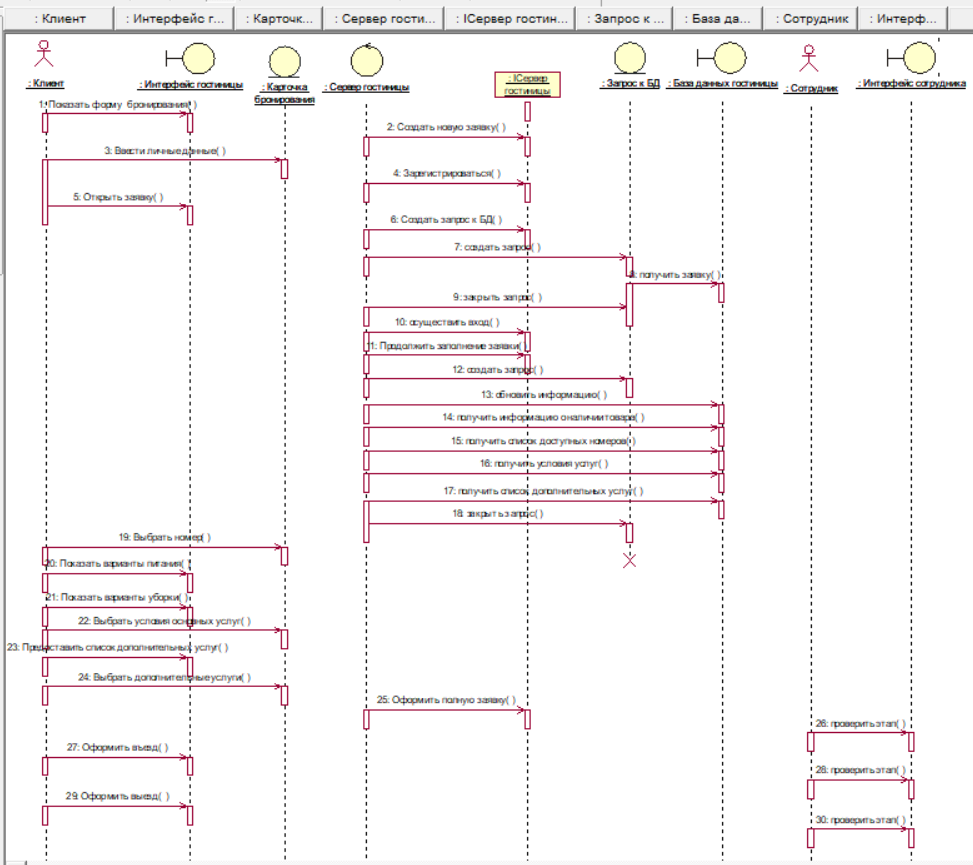


Рисунок 12 – Диаграмма последовательности

Диаграмма 5:Диаграмма состояний

Диаграммы состояний требуют, чтобы описываемая система состояла из конечного числа состояний. Состояние отображается в виде четырехугольника со скругленными углами, внутри которого обязательно записывается имя. С помощью диаграммы состояний можно описать процесс бронирования и последующие использование функционала (Рисунок 13).

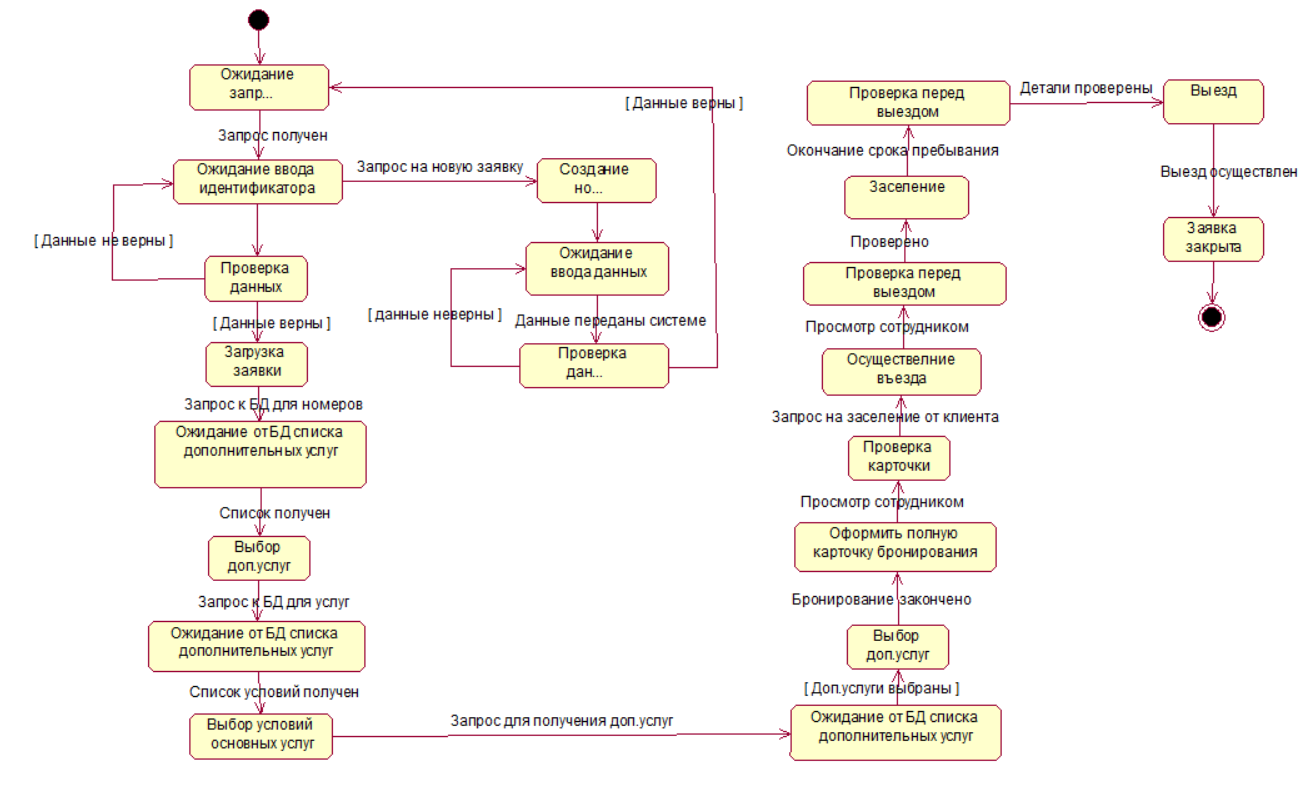


Рисунок 13 – Диаграмма состояний

Диаграмма 6:Диаграмма деятельности

Диаграмма деятельности акцентирует внимание на последовательности выполнения определенных действий, которые в совокупности приводят к получению желаемого результата Данную диаграмму можно построить, опираясь на диаграмму состояний (Рисунок 14).

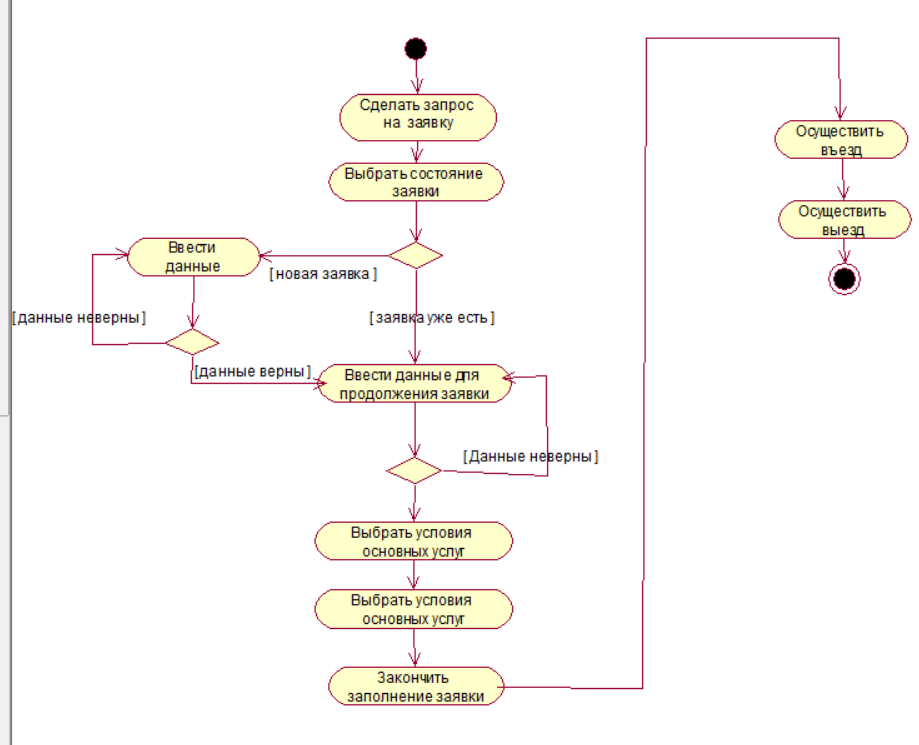


Рисунок 14 – Диаграмма деятельности

Диаграмма 7: Моделирование бизнес-процессов

Данный тип диаграмм крайне удобен для отображения последовательности выполнения бизнес-процессов, отделяемых дорожками. Опишем процесс выполнения услуги. Саму заявку будем рассматривать в качестве заказа. При приеме формируется заказ, который окончательно оформляется и передается в отдел предоставления услуг, счет же формируется и передается в отдел бухгалтерии (может быть фиктивным, а саму функцию исполнять часть системы, приложения). В отделе предоставления проверяется возможность оказать услугу, после чего поступает в следующий отдел: отдел исполнения услуг. Где при наличии чека и возможности, происходит оказание услуги (Рисунок 15).

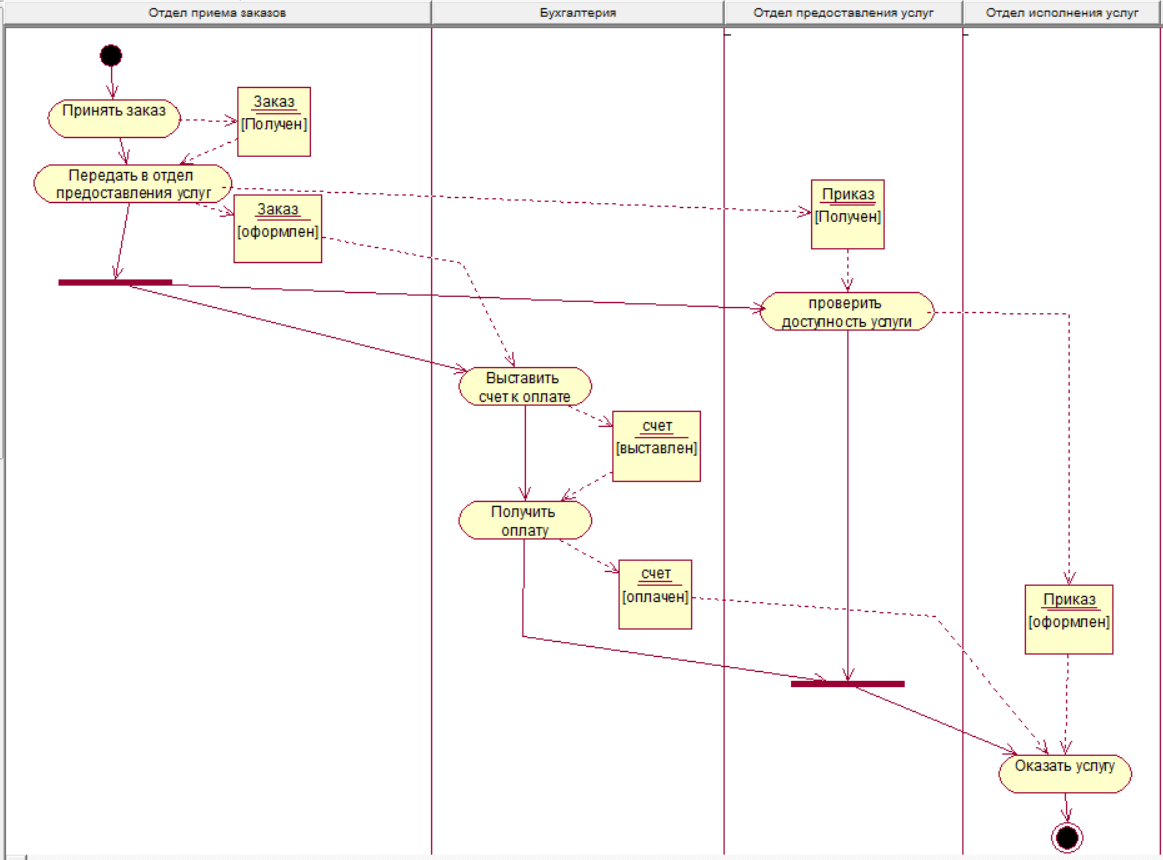


Рисунок 15 – Диаграмма деятельности с дорожками бизнес-процессов

Диаграмма 8: Диаграмма компонентов

Диаграммы компонентов используются для визуализации организации компонентов системы и зависимостей между ними. Они позволяют получить высокоуровневое представление о компонентах системы.Компонентами могут быть программные компоненты, такие как база данных или пользовательский интерфейс; или аппаратные компоненты, такие как схема, микросхема или устройство;

Одним из главных компонентов является приложение гостиницы, через которое пользователь взаимодействует с сервером. Сервер использует СУБД для работы с БД заявок и услуг. (Рисунок 16).

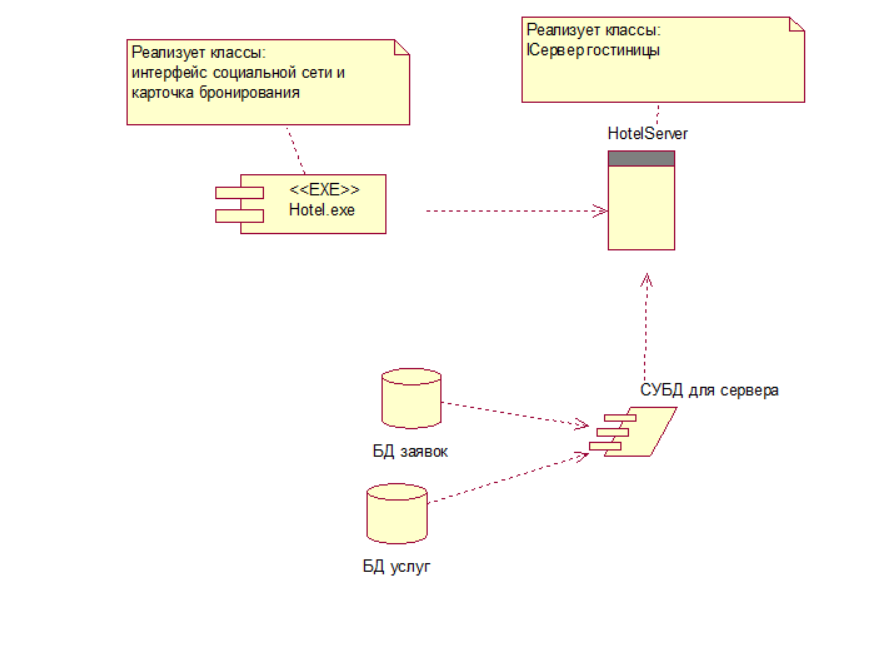


Рисунок 16 – Диаграмма компонентов

Диаграмма 9: Диаграмма развертывания

Диаграмма развертывания – это тип UML-диаграммы, которая показывает архитектуру исполнения системы, включая такие узлы, как аппаратные или программные среды исполнения, а также промежуточное программное обеспечение, соединяющее их.

Диаграммы развертывания обычно используются для визуализации физического аппаратного и программного обеспечения системы. Используя его, вы можете понять, как система будет физически развернута на аппаратном обеспечении.

В данной предметной области выделим следующее – приложение, установленное в различных гостиницах. Все передается через сервер на специализированный сервер (Рисунок 17).

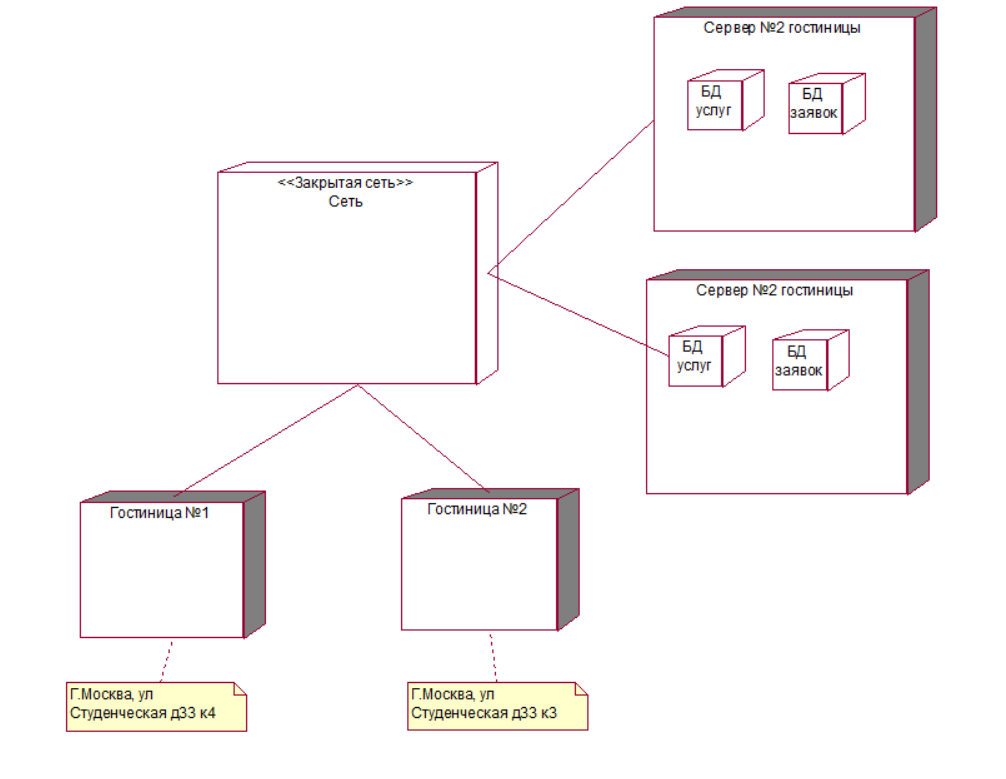


Рисунок 17 – Диаграмма развертывания

# Выводы

В данной практической работе были изучены методы и технологии проектирования баз данных. В первой части работы были выполнены задания по проектированию в методологиях IDEF0, IDEF3 и DFD: в методологии IDEF0 создана общая диаграмма, после последовательно декомпозировали на меньшие активности, представляя в методологиях DFD, IDEF3, IDEF0. Во второй части работы были выполнены задания по разработке различных типов UML- диаграмм в предметной области «Гостиница».

# Список информационных источников

1. Лекционный материал по дисциплине «Проектирование баз данных» Назаренко М. А.
2. DataFlowDiagrams // ScienceDirect [Электронный ресурс]. - [https://www.sciencedirect.com/science/article/pii/B0122272404000319](%20https://www.sciencedirect.com/science/article/pii/B0122272404000319)– (дата обращения: 16.04.2022)
3. Основы IDEF3 //CFIN [Электронный ресурс]. - <https://www.cfin.ru/vernikov/idef/idef3.shtml> – (дата обращения: 17.04.2022)