ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ПУТЕЙ

СООБЩЕНИЯ

(ФГБОУ ВО ИрГУПС)»

Факультет «Управление на транспорте и информационные технологии»

Кафедра «Информационные системы и защита информации»

Дисциплина «Инструментальные средства информационных систем»

Отчет по лабораторной работе №2

Тема: «Проектирование и разработка архитектуры»

ЛР.420200.09.03.02.086.ПЗ

|  |  |
| --- | --- |
| Выполнил:  Студент гр. ИС. 1-21-1  Ерохина Е.В.  Дата\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Подпись:\_\_\_\_\_\_\_\_\_\_\_\_ | Проверил:  Старший преподаватель  Береснева Н.М.  Дата\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Подпись:\_\_\_\_\_\_\_\_\_\_\_\_ |

Иркутск 2024

Оглавление

[Цель работы: 3](#_Toc179593473)

[Порядок выполнения работы: 3](#_Toc179593474)

1. [Диаграмма вариантов использования. 3](#_Toc179593475)

[2. Диаграмма последовательности. 4](#_Toc179593476)

[3. Диаграмма деятельности. 5](#_Toc179593477)

[4. Диаграмма классов. 6](#_Toc179593478)

[5. Построение физической модели базы данных. 7](#_Toc179593479)

[Вывод: 8](#_Toc179593480)

**Лабораторная работа №2**

**Проектирование и разработка архитектуры**

# Цель работы:

Сделать диаграмму Use-Case, диаграмму последовательности, деятельности, классов. Спроектировать БД для своего проекта.

# Порядок выполнения работы:

### Диаграмма вариантов использования.

Диаграмма вариантов использования (Use Case) — это диаграмма, описывающая, какой функционал разрабатываемой программной системы доступен каждой группе пользователей.

Действующие лица данной диаграммы:

1) Администратор — может заносить информацию об итогах конференций, формировать положения конференций и осуществлять сбор данных о сотрудниках в информационной системе «Отделы», а также собирать данные о начальнике и об отделе в системе ЭИС, которая заносит информацию в ИС «База данных»;

2) Начальник отдела — может получать данные об отделах из ИС «База данных», изменять список отдела (формирование, расформирование, корректировка отдела), распределять нагрузку по сотрудникам, заносить информацию о сотрудниках на основании данных, предоставленными сотрудниками, составлять отчёты (отчёты по рабочему плану, по конференциям и итоговый отчёт). Кроме того, начальник может выполнять некоторые действия, что и администратор;

3) ИС «Отделы» — может получать информацию из ИС «База данных» об отделах, и на её основании создавать отделы и распределять по ним сотрудников. После этого, с этими сотрудниками взаимодействует начальник отдела;

4) ИС «База данных» — может быть доступна только администратору компании, который заносит в БД всю необходимую информацию.

Исходя из потребностей действующих лиц, выделяются следующие варианты использования:

1. Изменение списка отдела;

2. Занесение информации о сотруднике;

3. Сбор данных о сотрудниках.

Диаграмма вариантов использования показана на рисунке 1.

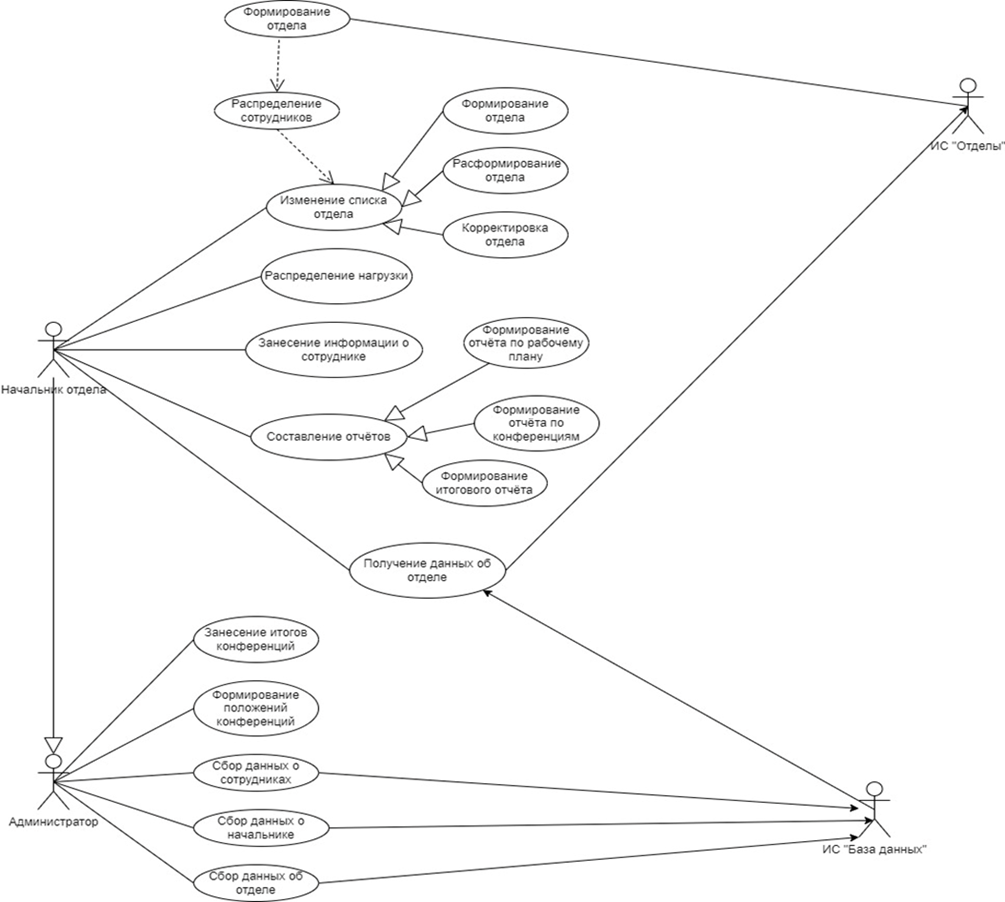


Рис. 1 Диаграмма вариантов использования

### 2. Диаграмма последовательности.

На рисунке 2 представлена диаграмма последовательностей (Sequence Diagram), которая описывает последовательность действий администратора. Так как у него нет прямого доступа к единой базе данных, то он пользуется информационной системой, разработанной специально для данной компании. Сперва ему нужно осуществить вход в саму ИС для последующих действий. После этого, администратору требуется получить список сотрудников. Затем, он уже приступает к редактированию формы внесения данных в ИС, которая потом автоматически внесёт новые данные в базу данных. Если редактирование будет успешно сохранено, то администратор получит доступ к списку конференций, но если система выдаст ошибку, то у администратора будет всего 2 действия: продолжить редактирование, либо сообщить о проблеме. При благополучном исходе событий, администратор получит данные как о конференциях, так и об их участниках. Конечная цель администратора — занести итоги конференций, что он и сделает после всех перечисленных действий.

Однако есть и второй вариант использования данной диаграммы: сразу начать редактирование в ИС, при условии, что администратор уже находится в ней. Это означает, что администратору не потребуется вновь входить в эту систему, и для внесения данных ему не понадобится список сотрудников. Тогда он сразу же может занести итоги конференций. Но если система выдаст ошибку, ему нужно будет продолжить вносить данные. При благополучном исходе будут получены данные об участниках конференций, при неблагополучном — произойдёт сбой системы.

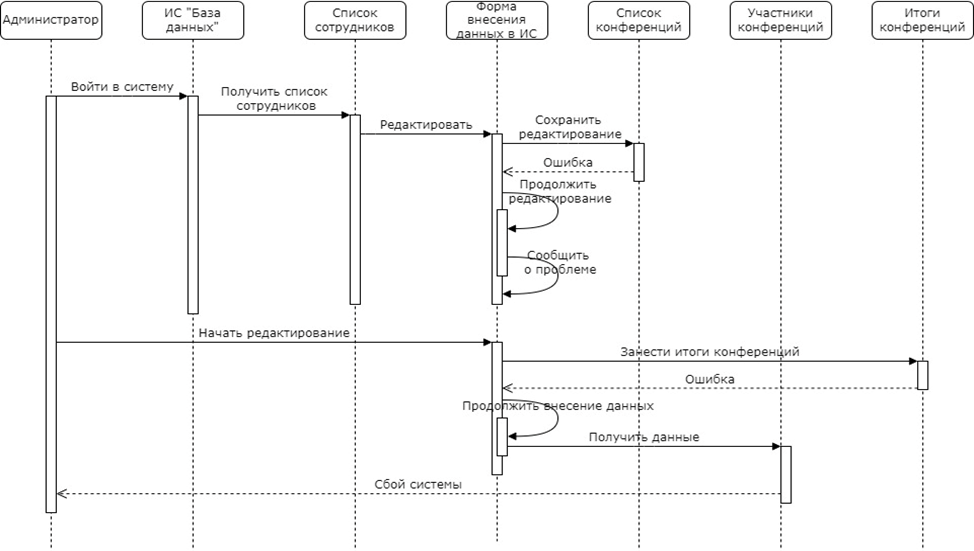


Рис. 2 Диаграмма последовательности

### 3. Диаграмма деятельности.

На рисунке 3 представлена диаграмма деятельности (Activity Diagram), которая описывает, какую деятельность осуществляет каждый из объектов для достижения единой цели. Сперва начальник какого-либо отдела получает список сотрудников. После этого, он должен выбрать сотрудника и внести о нём данные, как об участнике конференции. Если же по какой-то причине сотрудник больше не числится в этом отделе или вообще в компании, система не даст внести необходимые данные. В этом случае, начальник осуществит сортировку по нужным критериям, обновит систему и получит новый список сотрудников. Затем, он внесёт данные и отправит их администратору. Администратор сначала формирует положения конференций, а затем параллельно составляет список участников и вносит данные о них. Здесь также может быть и ошибка внесения, т.е. данные по какой-либо причине окажутся некорректны. Тогда администратор перепроверяет внесённые данные, изменяет их при необходимости, и только потом составляет список участников. Если список участников составлен правильно, администратор продолжит формировать положения конференций. Если же список был составлен неверно, то администратор повторит внесение данных в ИС. Тогда система обновится и данные будут автоматически внесены в базу данных. Результат достигнут.

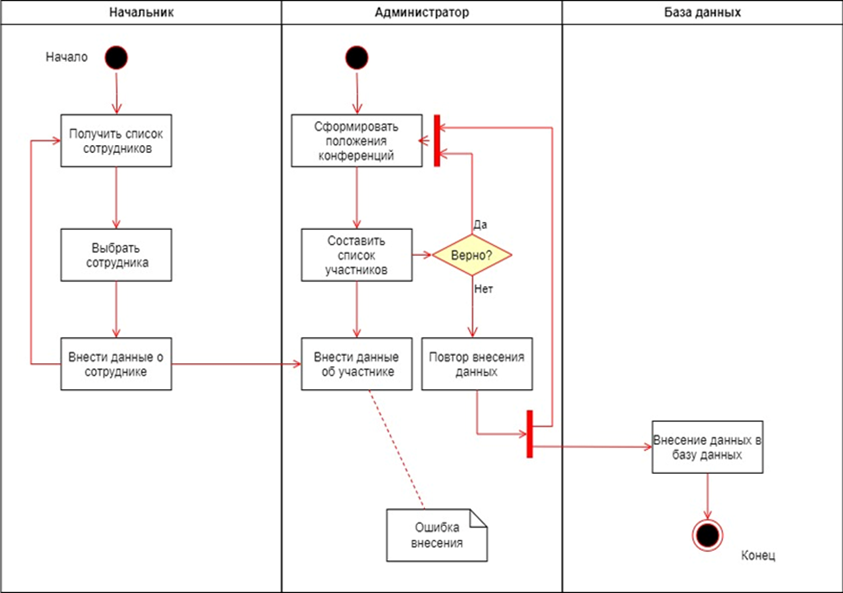


Рис. 3 Диаграмма деятельности

### 4. Диаграмма классов.

На рисунке 4 представлена диаграмма классов (Class Diagram). Она была выполнена в онлайн-программе «draw.io», предназначенная для построения различных типов диаграмм. Рассмотрим каждую связь между сущностями подробно. Один сотрудник имеет одну должность (1:1), один сотрудник оформляет много заказов (1:М), один сотрудник создаёт много закупок (1:М); один заказ включает в себя один статус (1:1), один заказ содержит много товаров (1:М); один клиент делает много заказов (1:М); на одном складе находится много товаров (1:М); одна фирма-дистрибьютор осуществляет много поставок (1:М), одна фирма-дистрибьютор взаимодействует с множеством закупок (1:М). Таким образом, мы детально рассмотрели диаграмму классов.

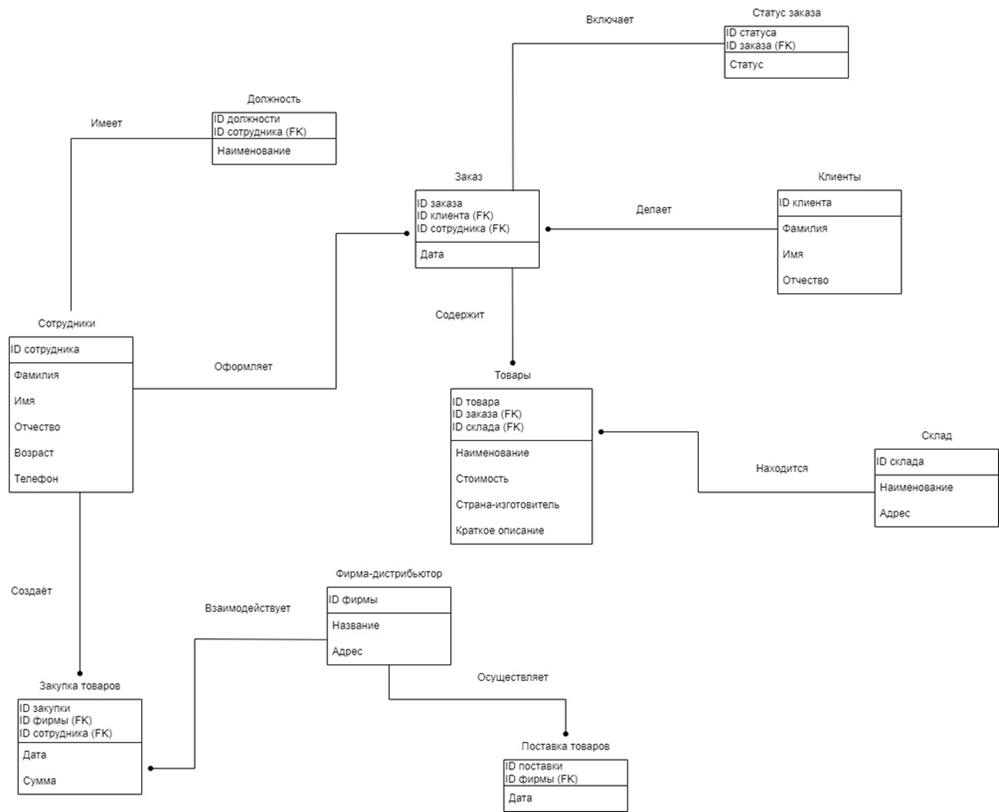


Рис. 4 Диаграмма классов

### 5. Построение физической модели базы данных.

На рисунке 5 представлена физическая модель базы данных типа «Сущность-связь». Она была выполнена в специальной программе для построения ER-диаграмм «DBDesigner». Данная диаграмма включает в себя основные сущности и связи с ними. Так, например, у таблицы «Сотрудники» есть 3 связи. Первая связь с таблицей «Должности» (каждый сотрудник имеет должность). Вторая связь с таблицей «Заказ» (какой-либо сотрудник оформляет заказ). Третья связь с таблицей «Закупка товаров» (один из сотрудников создаёт закупку товаров). Рассмотрим и другие связи и сущности диаграммы. К примеру, таблица «Заказ» содержит сразу несколько связей: заказ включает в себя статус заказа (таблица «Статус заказа»), заказ содержит товары (таблица «Товары»), заказ оформляет сотрудник (таблица «Сотрудники») и заказ делает клиент (таблица «Клиенты»). Теперь перейдём к оставшимся сущностям и их связям: товар находится на складе (таблица «Склад»), закупка товаров взаимодействует с фирмой-дистрибьютером (таблица «Фирма-дистрибьютор»), фирма-дистрибьютор осуществляет поставку товаров (таблица «Поставка товаров»).

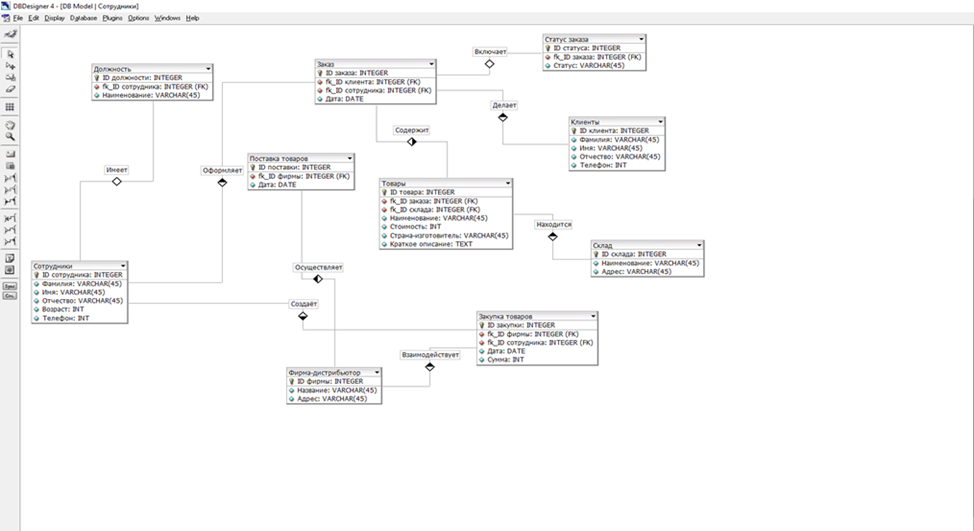


Рис. 5 Физическая модель базы данных

### Вывод:

В ходе выполнения данной лабораторной работы были сделаны диаграмма Use-Case, диаграмма последовательности, деятельности, классов, а также была спроектирована БД.