Санкт-Петербургский Национальный Исследовательский Университет Информационных Технологий, Механики и Оптики

ПИиКТ

Курсовая работа

по дисциплине

«Информационные системы и базы данных»

Выполнили: Студенты группы P33113

Мансуров Б.Б.

Кадыров А.Х.

Преподаватель: Николаев В.В.

     Санкт-Петербург

2020г

Задание

Производство и продажа экзотических колбасных изделий

Что из себя представляет наша предметная область

Представим, что нам нужно проследить путь колбасных изделий начиная от цеха по производству до прилавок магазинов и разных мясных лавок.  
Что из себя представляет наша предметная область: Есть много цехов где производят разные колбасные изделия у каждого цеха разный список продукции и каждый цех имеет свои фермы на разных городах и странах. В этих фермах есть разные животные включая экзотических животных. В зависимости от страны и города в фермах будут разводится разные животные включая экзотических которые доступны только в этой местности. У каждого цеха имеется склад в которых хранятся готовые изделия в течение 5 суток. У каждого цеха есть специальные люди (сотрудники цеха) которые будут поставлять разные колбасные изделия клиентам, под клиентами подразумевается продавец в магазине или в мясной лавке или еще где ни будь. Есть специальные лавки где продаются только экзотическая продукция цеха. У каждого цеха будет свой график, в зависимости от графика будут поставляться колбасные изделия.  Сотрудник одновременно может быть сотрудников нескольких цехов. В зависимости от груза (колбасных изделий) каждому сотруднику будет предоставлена специальная машина для перевозки груза. Город разделен на участки, каждый сотрудник может поставлять колбасные изделия только на своих участках. Продавцы могут иметь несколько поставщиков из разных цехов. Бывают такие случаи, когда колбасные изделия могут иметь дефекты и в этом случае продавец может обратно возвратить эти товары поставщикам, и поставщики в свою очередь цеху.  Клиент не всегда платежа способен, и поставщики чтобы не потерять клиентов могут отдать в долг товары. Клиенты могут сделать специальные заказы которые будут обрабатываться отдельно. Для сотрудников цехов и постоянных клиентов будут предоставлены скидки и льготные услуги на продукцию. Есть каталог подписок для клиентов в зависимости от подписки клиент может составить список продукции, которых может заказать. Для VIP клиентов у нас будет VIP подписки заказы которых будут обрабатываться отдельно.

Процессы

Проследим путь заказа: Клиент приобретает подписку -> заполняет список продуктов для заказа -> заказ добавляется в график заказов -> проверяется наличия продукции на складе -> в зависимости от местоположения заказчика будет выбран поставщик -> заказ будет принят поставщиком -> и в зависимости от веса будет предоставлена машина для перевозки груза -> товар будет доставлен.

Проследим путь возврата: У клиента есть дефект -> он заполняет список дефектных продуктов, и он проверяется с его заказом и если в заказе есть такой продукт, то дефект принимается -> и в зависимости кто поставил этот продукт будет выбран поставщик -> поставщик принимает этот дефект -> и дефект предоставляется цеху.

Проследим путь VIP заказа: клиент вне зависимости от местоположения может сделать заказ -> и этот заказ попадает в отдельную таблицу спец заказов -> и если в складе каких страны, то есть городов в этой стране есть этот продукт -> то заказ будет принят ближайшим поставщиком -> и товар будет доставлен.

Покупка товара исп. скидку: клиент заполняет список продуктов для заказа -> проверяется есть ли скидка и тому подобное -> и если есть скидка зачитывается -> и дальше обрабатывается как обычный заказ.

1. Имеются цехи которые производят колбасные изделия  
2. У каждого цеха имеется список колбасных изделий  
3. Есть разные страны и разные города.  
4. Имеется разные фермы где будут разводится разные животные включая экзотических  
5. У каждого цеха есть свои сотрудники которые будут поставлять колбасные изделия клиентам.  
6. Есть список клиентов   
7. Клиент может заказать колбасные изделия  
8. Из заказов клиентов строится график доставок  
9. В зависимости от заказа будет предоставлена машина для перевозки груза  
10. Города разделены на участки, участки разделены между поставщиков, и поставщик может поставлять колбасные изделия только на своем участок.  
11. Если товар имеет дефекты клиент имеет право возвратить товар

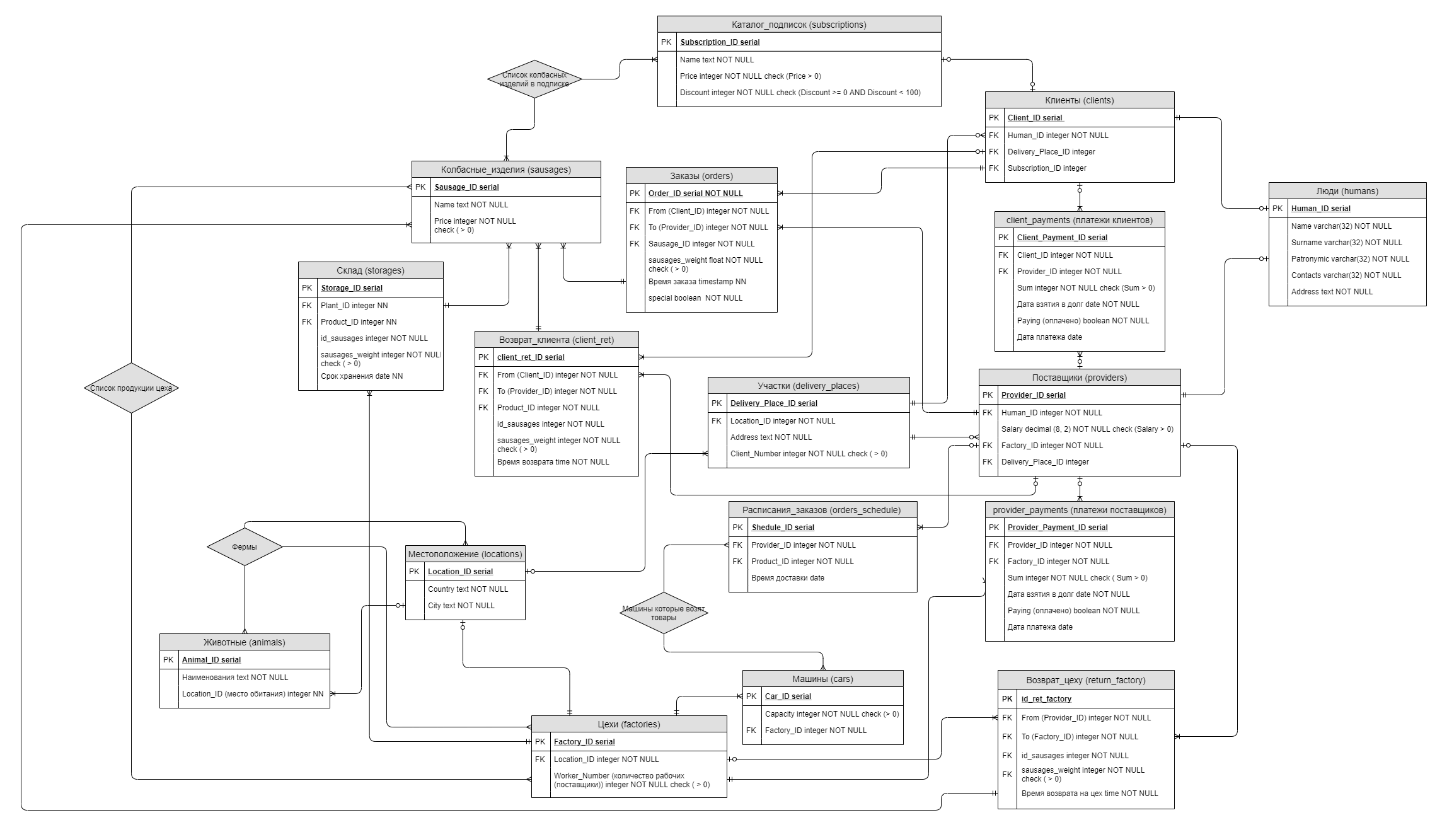
1. Возврат от клиента поставщику
2. Поставщик цеху

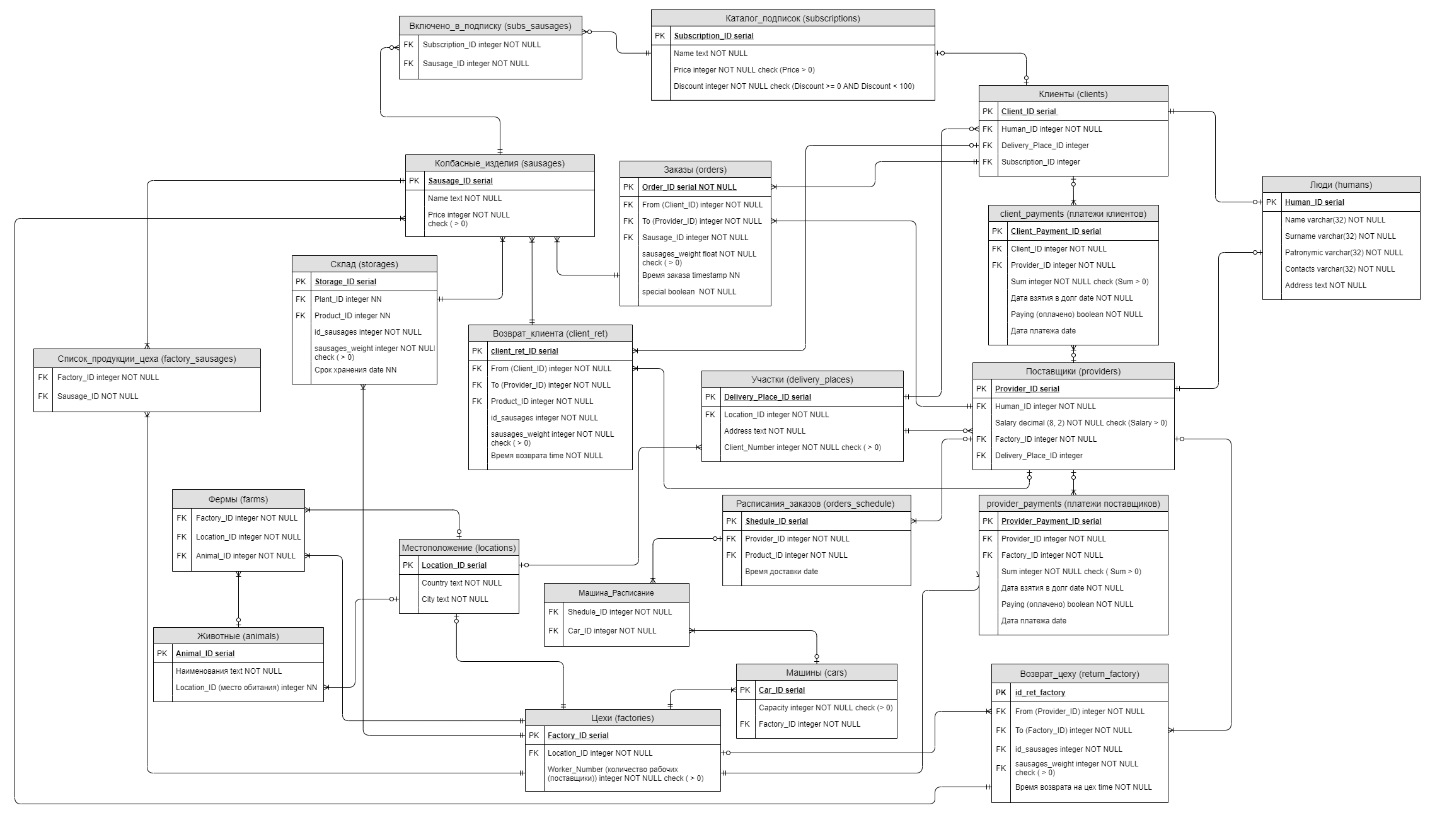
12. Клиент может взять товар в долг  
13. Есть специальные заказы которые будут обрабатываться отдельно  
14. Есть каталог подписок и в зависимости от подписки будут составлено список продукции для заказа.

Список сущностей и их классификация

1. Цехи
2. Колбасные изделия
3. Животные
4. Местоположения
5. Склады
6. Клиенты (связь)
7. Поставщики
8. Заказ
9. Участки
10. Расписание
11. Машины
12. Подписки
13. Долги клиентов
14. Долги поставщиков
15. Возврат поставщикам
16. Возврат клиентам
17. Продукт

Инфологическая модель

  
  
Дата логическая модель



Реализация

CREATE TABLE humans  
(  
 id\_human serial,  
 name varchar(32) NOT NULL,  
 surname varchar(32) NOT NULL,  
 patronymic varchar(32) NOT NULL,  
 contacts varchar(32) NOT NULL,  
 address text NOT NULL,  
 PRIMARY KEY (id\_human)  
);  
  
CREATE TABLE sausages  
(  
 id\_sausage serial,  
 name text NOT NULL,  
 price decimal(7, 2) NOT NULL CHECK (price > 0),  
 PRIMARY KEY (id\_sausage)  
);  
  
CREATE TABLE locations  
(  
 id\_location serial,  
 country text NOT NULL,  
 city text NOT NULL,  
 PRIMARY KEY (id\_location)  
);  
  
CREATE TABLE delivery\_places  
(  
 id\_delivery\_place serial,  
 id\_location integer NOT NULL,  
 address text NOT NULL,  
 client\_num integer NOT NULL CHECK ( client\_num > 0),  
 PRIMARY KEY (id\_delivery\_place),  
 FOREIGN KEY (id\_location) REFERENCES locations (id\_location)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE  
);  
  
CREATE TABLE factories  
(  
 id\_factory serial,  
 location integer NOT NULL,  
 worker\_num integer NOT NULL CHECK (worker\_num > 0),  
 PRIMARY KEY (id\_factory),  
 FOREIGN KEY (location) REFERENCES locations (id\_location)  
);  
  
CREATE TABLE animals  
(  
 id\_animal serial,  
 name text NOT NULL,  
 location integer NOT NULL,  
 PRIMARY KEY (id\_animal),  
 FOREIGN KEY (location) REFERENCES locations (id\_location)  
);  
  
CREATE TABLE cars  
(  
 id\_car serial,  
 capacity integer NOT NULL CHECK (capacity > 0),  
 id\_factory integer NOT NULL,  
 PRIMARY KEY (id\_car),  
 FOREIGN KEY (id\_factory) REFERENCES factories (id\_factory)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE  
);  
  
CREATE TABLE subscriptions  
(  
 id\_subscription serial,  
 name text NOT NULL,  
 price integer NOT NULL CHECK (price > 0),  
 discount integer NOT NULL CHECK (discount >= 0 AND discount < 100),  
 PRIMARY KEY (id\_subscription)  
);  
  
CREATE TABLE subs\_sausages  
(  
 id\_subscription integer NOT NULL,  
 id\_sausage integer NOT NULL,  
 PRIMARY KEY (id\_subscription, id\_sausage),  
 FOREIGN KEY (id\_subscription) REFERENCES subscriptions (id\_subscription)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_sausage) REFERENCES sausages (id\_sausage)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE  
);  
  
CREATE TABLE providers  
(  
 id\_provider serial,  
 id\_human integer NOT NULL,  
 salary decimal(8, 2) NOT NULL CHECK (salary > 0),  
 id\_factory integer NOT NULL,  
 delivery\_place integer,  
 PRIMARY KEY (id\_provider),  
 FOREIGN KEY (id\_human) REFERENCES humans (id\_human)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_factory) REFERENCES factories (id\_factory)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (delivery\_place) REFERENCES delivery\_places (id\_delivery\_place)  
 ON UPDATE CASCADE  
 ON DELETE SET NULL  
  
);  
  
CREATE TABLE clients  
(  
 id\_client serial,  
 id\_human integer NOT NULL,  
 id\_delivery\_place integer,  
 id\_subscription integer,  
 PRIMARY KEY (id\_client),  
 FOREIGN KEY (id\_human) REFERENCES humans (id\_human)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_delivery\_place) REFERENCES delivery\_places (id\_delivery\_place)  
 ON UPDATE CASCADE  
 ON DELETE SET NULL,  
 FOREIGN KEY (id\_subscription) REFERENCES subscriptions (id\_subscription)  
 ON UPDATE CASCADE  
 ON DELETE SET NULL  
);  
  
CREATE TABLE orders  
(  
 id\_order serial,  
 from integer NOT NULL,  
 to integer NOT NULL,  
 id\_sausage integer NOT NULL,  
 sausages\_weight float NOT NULL CHECK (sausages\_weight > 0),  
 ord\_time timestamp NOT NULL,  
 special boolean NOT NULL,  
 PRIMARY KEY (id\_order),  
 FOREIGN KEY (from) REFERENCES clients (id\_client)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (to) REFERENCES providers (id\_provider)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_sausage) REFERENCES sausages(id\_sausage)  
  
);  
  
CREATE TABLE client\_payments  
(  
 id\_client\_payment serial NOT NULL,  
 id\_client integer NOT NULL,  
 id\_provider integer NOT NULL,  
 sum integer NOT NULL CHECK (sum > 0),  
 dept\_date date NOT NULL,  
 paying boolean NOT NULL,  
 payment\_date date,  
 PRIMARY KEY (id\_client\_payment),  
 FOREIGN KEY (id\_client) REFERENCES clients (id\_client)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_provider) REFERENCES providers (id\_provider)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE  
);  
  
CREATE TABLE providers\_payments  
(  
 id\_provider\_payment serial,  
 id\_provider integer NOT NULL,  
 id\_factory integer NOT NULL,  
 sum integer NOT NULL CHECK (sum > 0),  
 dept\_date date NOT NULL,  
 paying boolean NOT NULL,  
 payment\_date date,  
 FOREIGN KEY (id\_provider) REFERENCES providers (id\_provider)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_factory) REFERENCES factories (id\_factory)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE  
);  
  
CREATE TABLE order\_shedule  
(  
 id\_shedule serial,  
 id\_provider integer NOT NULL,  
 id\_sausage integer NOT NULL,  
 sausages\_weight float NOT NULL CHECK (sausages\_weight > 0),  
 del\_time date NOT NULL,  
 PRIMARY KEY (id\_shedule),  
 FOREIGN KEY (id\_provider) REFERENCES providers (id\_provider)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_sausage) REFERENCES sausages(id\_sausage)  
);  
  
CREATE TABLE car\_shedule  
(  
 id\_shedule integer NOT NULL,  
 id\_car integer NOT NULL,  
 FOREIGN KEY (id\_shedule) REFERENCES order\_shedule (id\_shedule)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_car) REFERENCES cars (id\_car)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 PRIMARY KEY (id\_shedule, id\_car)  
  
);  
  
CREATE TABLE storages  
(  
 id\_storage serial,  
 id\_factory integer NOT NULL,  
 id\_sausage integer NOT NULL,  
 sausages\_weight float NOT NULL CHECK (sausages\_weight > 0),  
 stor\_life date,  
 FOREIGN KEY (id\_factory) REFERENCES factories (id\_factory)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_sausage) REFERENCES sausages(id\_sausage)  
);  
  
CREATE TABLE returns  
(  
 id\_return serial,  
 from integer NOT NULL,  
 to integer NOT NULL,  
 id\_sausage integer NOT NULL,  
 sausages\_weight float NOT NULL CHECK (sausages\_weight > 0),  
 ret\_time time NOT NULL,  
 PRIMARY KEY (id\_return),  
 FOREIGN KEY (from) REFERENCES clients (id\_client)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (to) REFERENCES providers (id\_provider)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_sausage) REFERENCES sausages(id\_sausage)  
  
);  
  
CREATE TABLE factory\_sausages  
(  
 id\_factory integer NOT NULL,  
 id\_sausage integer NOT NULL,  
 PRIMARY KEY (id\_factory, id\_sausage),  
 FOREIGN KEY (id\_factory) REFERENCES factories (id\_factory)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_sausage) REFERENCES sausages (id\_sausage)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE  
);  
  
CREATE TABLE farms  
(  
 id\_factory integer NOT NULL,  
 id\_location integer NOT NULL,  
 id\_animal integer NOT NULL,  
 PRIMARY KEY (id\_factory, id\_location, id\_animal),  
 FOREIGN KEY (id\_factory) REFERENCES factories (id\_factory)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_location) REFERENCES locations (id\_location)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_animal) REFERENCES animals (id\_animal)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE  
);  
  
CREATE TABLE return\_factory  
(  
 id\_ret\_factory serial,  
 from integer NOT NULL,  
 to integer NOT NULL,  
 id\_sausage integer NOT NULL,  
 sausages\_weight float NOT NULL CHECK (sausages\_weight > 0),  
 ret\_time time NOT NULL,  
 PRIMARY KEY (id\_ret\_factory),  
 FOREIGN KEY (from) REFERENCES providers (id\_provider)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (to) REFERENCES factories (id\_factory)  
 ON UPDATE CASCADE  
 ON DELETE CASCADE,  
 FOREIGN KEY (id\_sausage) REFERENCES sausages(id\_sausage)  
);