Министерство науки и высшего образования Российской Федерации

Федеральное государственное образовательное учреждение

высшего образования

**ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра систем автоматизации управления

**Дисциплина: Проектирование информационного обеспечения**

**Тема: «**Проектирование базы данных**»**

|  |  |  |  |
| --- | --- | --- | --- |
| Выполнил студент  группы ИТб2302-02-00 |  | Плюснин Д.А. |  |
|  |  |

Киров 2021

# Кейс 12. Учет медикаментов и изделий медицинского назначения в стоматологии

Для ответственного за материальные запасы в стоматологической клинике необходимо в любой момент знать актуальное количество медикаментов и расходных материалов на складе и в кабинетах врачей, отслеживать сроки годности и иметь возможность расчета необходимого минимального остатка с учетом среднего расхода.  Проведение постоянного контроля над складскими остатками позволяет избежать сложности при инвентаризации.

**Требования**

Разработать автоматизированную систему учета лекарственных препаратов (ЛП) и изделий медицинского назначения (ИМН) для стоматологической клиники, включающую в себя следующие возможности:

* Ведение каталогов ЛП и ИМН ;
* Учёт остатков ЛП и ИМН;
* Возможность заносить поступление новых товаров;
* При списании ЛП и ИМН на пациента предусмотреть возможность списания базового перечня препаратов, которые пригодятся при выполнении определенной стоматологической услуги (шаблон списания);
* Возможность редактирования выбранного шаблона для персонализированного списания на пациента.

**Бизнес-правила и ограничения.**

* Нельзя принимать и списывать ЛП и ИМН с истёкшим сроком годности.
* При наличии на складе нескольких партий ЛП или ИМН с разными сроками годности на складе списанию подлежит позиция с наименьшим сроком годности.

**Модель предметной области:**

**«Поступление товара на склад»**

* Наименование ЛП или ИМН
* Серия
* Срок годности
* Цена
* Количество

**«Товарные остатки»**

* Наименование ЛП или ИМН
* Серия
* Срок годности
* Цена
* Количество

**«Списание»**

* Медицинская услуга
* Пациент
* Наименование ЛП или ИМН
* Количество

**«Шаблон списания»**

* Медицинская услуга
* Наименование ЛП или ИМН
* Количество

Диаграмма IDEF1x с названиями cущностей и полей на русском языке

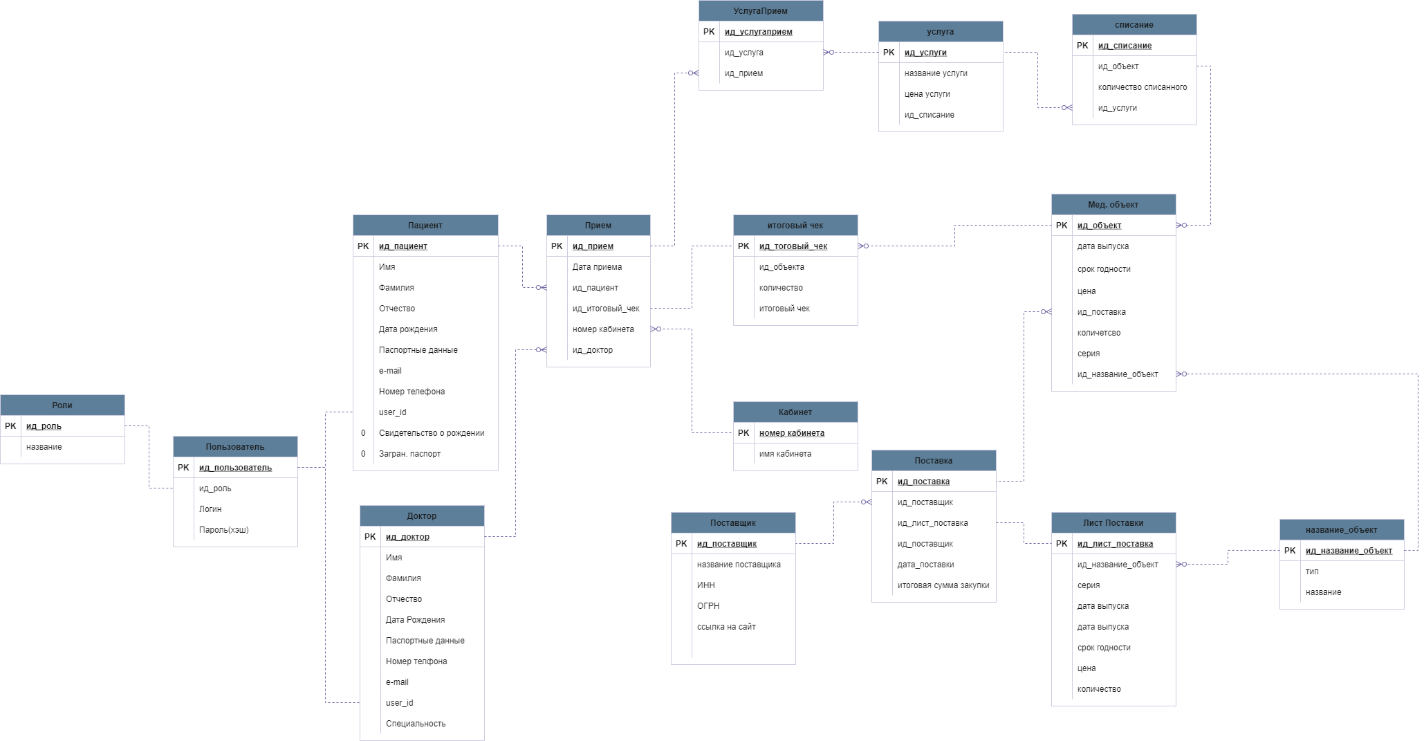
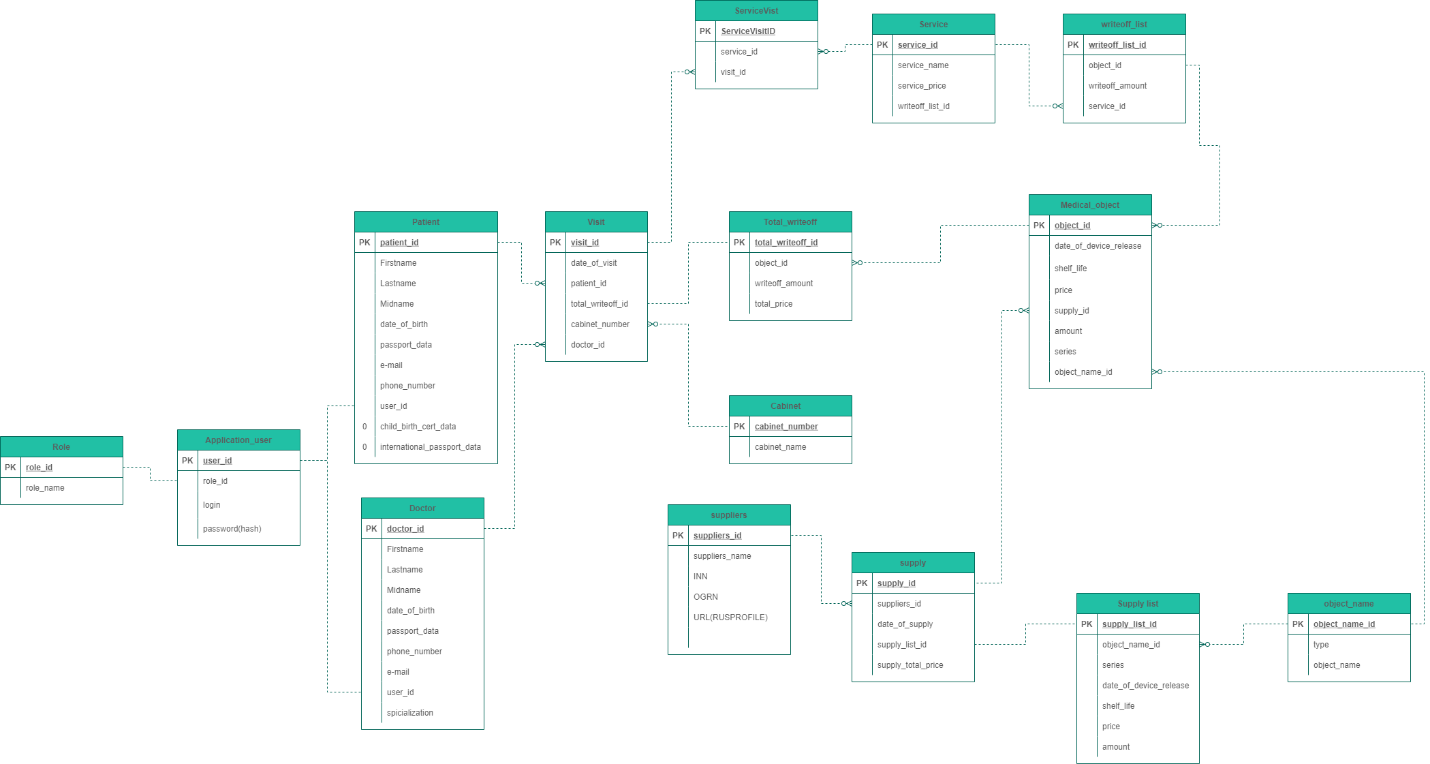


Диаграмма IDEF1x с названиями сущностей и полей на английском языке



Листинг базы данных Dentistry

*CREATE TABLE* Role\_list  
(  
 role\_id *SERIAL NOT NULL PRIMARY KEY*,  
 role\_name *varchar*(20) *NOT NULL*);  
  
*CREATE TABLE* Application\_user  
(  
 user\_id *SERIAL NOT NULL PRIMARY KEY*,  
 *login varchar*(20) *NOT NULL UNIQUE*,  
 *password varchar*(20) *NOT NULL*,  
 *role varchar*(20) *NOT NULL*,  
 role\_id *int NOT NULL*);  
  
*CREATE TABLE* Patient  
(  
 patient\_id *SERIAL NOT NULL PRIMARY KEY*,  
 firstname *varchar*(30) *NOT NULL*,  
 midname *varchar*(30) *NOT NULL*,  
 lastname *varchar*(30) *NOT NULL*,  
 date\_of\_birth *date NOT NULL*,  
 passport\_data *varchar*(10) *UNIQUE*,  
 email *varchar*(40) *NOT NULL*,  
 phone\_number *varchar*(15) *NOT NULL*,  
 child\_birth\_cert\_data *varchar*(20) *UNIQUE*,  
 international\_passport\_data *varchar*(20) *UNIQUE*,  
 user\_id *INT NOT NULL*);  
  
*CREATE TABLE* Doctor  
(  
 doctor\_id *SERIAL NOT NULL PRIMARY KEY*,  
 firstname *varchar*(30) *NOT NULL*,  
 midname *varchar*(30) *NOT NULL*,  
 lastname *varchar*(30) *NOT NULL*,  
 date\_of\_birth *date NOT NULL*,  
 passport\_data *varchar*(10) *UNIQUE*,  
 email *varchar*(40) *NOT NULL*,  
 phone\_number *varchar*(15) *NOT NULL*,  
 specialization *varchar*(40) *NOT NULL*,  
 user\_id *int NOT NULL*);  
  
*CREATE TABLE* Cabinet  
(  
 cabinet\_number *varchar*(10) *NOT NULL PRIMARY KEY UNIQUE*,  
 cabinet\_name *varchar*(30) *NOT NULL*);  
  
*CREATE TABLE* Total\_writeoff\_list  
(  
 total\_writeoff\_list\_id *SERIAL NOT NULL PRIMARY KEY*,  
 writeoff\_ammount *int*,  
 total\_price *int*,  
 object\_id *int NOT NULL*);  
*CREATE TABLE* Visit  
(  
 visit\_id *SERIAL NOT NULL PRIMARY KEY*,  
 date\_of\_visit *date NOT NULL*,  
 patient\_id *INT NOT NULL*,  
 total\_writeoff\_list\_id *INT*,  
 cabinet\_number *varchar*(10) *NOT NULL*,  
 doctor\_id *INT NOT NULL*);  
  
*CREATE TABLE* Service  
(  
 service\_id *SERIAL NOT NULL PRIMARY KEY*,  
 service\_name *varchar*(20) *NOT NULL*,  
 service\_price *int NOT NULL*,  
 writeoff\_list\_id *int NOT NULL*);  
  
*CREATE TABLE* Writeoff\_list  
(  
 writeoff\_list\_id *SERIAL NOT NULL PRIMARY KEY*,  
 writeoff\_amount *int*,  
 object\_id *int NOT NULL*,  
 service\_id *int not null*);  
*CREATE TABLE* Service\_visit  
(  
 ServiceVisit\_id *SERIAL NOT NULL PRIMARY KEY*,  
 service\_id *int NOT NULL*,  
 visit\_id *int NOT NULL*);  
  
*CREATE TABLE* Medical\_object  
(  
 object\_id *SERIAL NOT NULL PRIMARY KEY*,  
 date\_of\_object\_release *date NOT NULL*,  
 shelf\_life *date NOT NULL*,  
 price *int NOT NULL*,  
 ammount *int NOT NULL*,  
 series *varchar*(40),  
 object\_name\_id *INT NOT NULL*,  
 supply\_id *INT NOT NULL*);  
  
*CREATE TABLE* Object\_name  
(  
 object\_name\_id *SERIAL NOT NULL PRIMARY KEY*,  
 *type varchar*(10) *NOT NULL*,  
 object\_name *varchar*(40) *NOT NULL*);  
  
*CREATE TABLE* Suppliers  
(  
 suppliers\_id *SERIAL NOT NULL PRIMARY KEY*,  
 suppliers\_name *varchar*(50) *NOT NULL*,  
 INN *varchar*(10) *NOT NULL UNIQUE*,  
 OGRN *varchar*(13) *NOT NULL UNIQUE*,  
 URL *varchar*);  
  
*CREATE TABLE* Supply  
(  
 supply\_id *SERIAL NOT NULL PRIMARY KEY*,  
 suppliers\_id *INT NOT NULL*,  
 date\_of\_supply *date NOT NULL*,  
 supply\_list\_id *int NOT NULL*);  
  
*CREATE TABLE* Supply\_list  
(  
 supply\_list\_id *SERIAL NOT NULL PRIMARY KEY*,  
 object\_name\_id *INT NOT NULL*,  
 series *varchar*(40),  
 date\_of\_device\_release *date NOT NULL*,  
 shelf\_life *date NOT NULL*,  
 price *int NOT NULL*,  
 ammount *int NOT NULL*);  
*ALTER TABLE* Application\_user  
 *ADD CONSTRAINT* fk\_role\_id *FOREIGN KEY* (role\_id) *REFERENCES* Role\_list (role\_id);  
*ALTER TABLE* Patient  
 *ADD CONSTRAINT* fk\_user\_id *FOREIGN KEY* (user\_id) *REFERENCES* Application\_user (user\_id);  
*ALTER TABLE* Doctor  
 *ADD CONSTRAINT* fk\_user\_id *FOREIGN KEY* (user\_id) *REFERENCES* Application\_user (user\_id);  
*ALTER TABLE* Total\_writeoff\_list  
 *ADD CONSTRAINT* fk\_object\_id *FOREIGN KEY* (object\_id) *REFERENCES* Medical\_object (object\_id);  
*ALTER TABLE* Total\_writeoff\_list  
 *ADD CONSTRAINT* check\_total\_price *CHECK* ( total\_price >= 0 );  
*ALTER TABLE* Total\_writeoff\_list  
 *ADD CONSTRAINT* check\_amount *CHECK* ( writeoff\_ammount >= 0);  
*ALTER TABLE* Service  
 *ADD CONSTRAINT* check\_service\_price *CHECK* ( service\_price >= 0 );  
*ALTER TABLE* Visit  
 *ADD CONSTRAINT* fk\_total\_writeoff\_list\_id *FOREIGN KEY* (total\_writeoff\_list\_id) *REFERENCES* Total\_writeoff\_list (total\_writeoff\_list\_id);  
*ALTER TABLE* Visit  
 *ADD CONSTRAINT* fk\_patient\_id *FOREIGN KEY* (patient\_id) *REFERENCES* Patient (patient\_id);  
*ALTER TABLE* Visit  
 *ADD CONSTRAINT* fk\_cabinet\_number *FOREIGN KEY* (cabinet\_number) *REFERENCES* Cabinet (cabinet\_number);  
*ALTER TABLE* Visit  
 *ADD CONSTRAINT* fk\_doctor\_id *FOREIGN KEY* (doctor\_id) *REFERENCES* Doctor (doctor\_id);  
*ALTER TABLE* Writeoff\_list  
 *ADD CONSTRAINT* fk\_service\_id *FOREIGN KEY* (service\_id) *REFERENCES* Service (service\_id);  
*ALTER TABLE* Writeoff\_list  
 *ADD CONSTRAINT* fk\_object\_id *FOREIGN KEY* (object\_id) *REFERENCES* Medical\_object (object\_id);  
*ALTER TABLE* Service\_visit  
 *ADD CONSTRAINT* fk\_service\_id *FOREIGN KEY* (service\_id) *REFERENCES* Service (service\_id);  
*ALTER TABLE* Service\_visit  
 *ADD CONSTRAINT* fk\_visit\_id *FOREIGN KEY* (visit\_id) *REFERENCES* Visit (visit\_id);  
*ALTER TABLE* Medical\_object  
 *ADD CONSTRAINT* fk\_object\_id *FOREIGN KEY* (object\_name\_id) *REFERENCES* Object\_name (object\_name\_id);  
*ALTER TABLE* Medical\_object  
 *ADD CONSTRAINT* fk\_supply\_id *FOREIGN KEY* (supply\_id) *REFERENCES* supply (supply\_id);  
*ALTER TABLE* Medical\_object  
 *ADD CONSTRAINT* check\_price *CHECK* ( price >= 0 );  
*ALTER TABLE* Medical\_object  
 *ADD CONSTRAINT* check\_amount *CHECK* ( ammount >= 0 );  
*ALTER TABLE* Supply  
 *ADD CONSTRAINT* fk\_suppliers\_id *FOREIGN KEY* (suppliers\_id) *REFERENCES* Suppliers (suppliers\_id);  
*ALTER TABLE* Supply  
 *ADD CONSTRAINT* fk\_supply\_list\_id *FOREIGN KEY* (supply\_list\_id) *REFERENCES* Supply\_list (supply\_list\_id);  
*ALTER TABLE* Supply\_list  
 *ADD CONSTRAINT* fk\_object\_name\_id *FOREIGN KEY* (object\_name\_id) *REFERENCES* Object\_name (object\_name\_id);  
*ALTER TABLE* Supply\_list  
 *ADD CONSTRAINT* check\_price *CHECK* ( price >= 0 );  
*ALTER TABLE* Supply\_list  
 *ADD CONSTRAINT* check\_ammount *CHECK* ( ammount >= 0 );