Національний університет «Львівська політехніка»

Кафедра програмного забезпечення

**КУРСОВА РОБОТА**

**з дисципліни «Бази даних»**

на тему:

«База даних для обліку комп’ютерних пристроїв на підприємстві»

Виконав: студент

напряму 6.050103«Програмна інженерія»

групи ПІ-31

Павленчик М. М.

Керівник:

асистент кафедри програмного забезпечення, к.т.н. Квятковський Б.О.

Оцінка:

Національна шкала\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів\_\_\_\_\_\_\_Оцінка ECTS\_\_\_\_

Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_ Павич Н.Я.

(підпис)

\_\_\_\_\_\_\_\_\_\_\_\_ Квятковський Б.О.

(підпис)

Львів – 2018 рік

ЗМІСТ

РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ ОБЛІКУ КОМП’ЮТЕРНИХ ПРИСТРОЇВ НА ПІДПРИЄМСТВІ

* 1. Опис предметної області
  2. Постановка завдання
  3. Специфікація вимог до системи обліку комп’ютерних пристроїв
     1. Вступ
     2. Загальний опис
     3. Характеристики системи
     4. Вимоги зовнішніх інтерфейсів
     5. Нефункціональні вимоги до проекту
  4. Вибір засобів розроблення системи

РОЗДІЛ 2. ПРОЕКТУВАННЯ БАЗИ ДАНИХ

2.1. Концептуальне моделювання предметної області

2.2. Логічне проектування бази даних

2.3. Проектування типових запитів і транзакцій

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОГРАМИ РОБОТИ З БАЗОЮ ДАНИХ

3.1. Реалізація доступу до даних

3.2. Реалізація функціональних характеристик системи

3.3. Опис роботи програми

ВИСНОВКИ

СПИСОК ЛІТЕРАТУРИ

Додаток А. Скрипт створення БД

# РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ ОБЛІКУ КОМП’ЮТЕРНИХ ПРИСТРОЇВ НА ПІДПРИЄМСТВІ

## **1.1. Опис предметної області**

Важко уявити сучасне життя без комп'ютерів, а тим більше важко уявити без них роботу підприємств. Проте комп'ютер – це не тільки досягнення цивілізації, але й об'єкт обліку.

Діяльність будь-якого підприємства, що використовує комп’ютерну техніку, обов’язково буде включати в себе облік цих пристроїв, підтримку їх справності, покращення їх характеристик, їх переміщення та інші дії. Також у процесі обліку постають питання щодо методу обліку самих пристроїв. Наприклад, якщо підприємство придбало стаціонарний комп’ютер, то в обліку це можна відобразити як придбання одного суцільного пристрою, або ж як придбання різних комплектуючих (системний блок, монітор, клавіатура, контролер миші).

Відслідковування ці всі активності не використовуючи додаткових, автоматизованих засобів може виявитись доволі трудомістким процесом. А також в певні моменти це може вимагати від бухгалтерів чи менеджерів фірми певного рівня знань щодо архітектури предметів обліку.

Цих всіх незручностей можна позбавитись використовуючи для обліку пристроїв певну автоматизовану систему. Отже, розроблення бази даних для обліку комп’ютерних пристроїв на підприємстві дозволить автоматизувати облік, облегшити документування будь-яких аспектів діяльності фірми, що пов’язані із комп’ютерними пристроями, а також полегшити працю бухгалтерів та менеджерів підприємства.

## **1.2. Постановка завдання**

Необхідно розробити базу даних та застосунок для обліку комп’ютерних пристроїв на підприємстві.

За допомогою застосунку користувач повинен мати змогу слідкувати за наявними пристроями на підприємстві, їх справністю та характеристиками. Також користувач повинен мати змогу вести записи щодо будь-якої діяльності, що пов’язана із комп’ютерними пристроями на підприємстві. Такої як: придбання нових пристроїв, ремонтування наявних, списання непрацюючих пристроїв, зміна комплектацій та інше.

Застосунок повинен підтримувати різні типи користувачів, з різними рівнями повноважень.

## **1.3. Специфікація вимог до системи обліку комп’ютерних пристроїв на підприємстві**

### **1.3.1. Вступ**

Розроблений проект призначений для полегшення ведення обліку комп’ютерних пристроїв на підприємстві завдяки введенню системи обліку та управління діяльностями підприємства, що пов’язані із вказаними пристроями.

Нижче подано специфікацію вимог до програмного продукту, де описано функціональні та не функціональні вимоги, вимоги до зовнішніх інтерфейсів та технології, застосовані для розробки системи.

### **1.3.2. Загальний опис**

Основне призначення системи – це полегшення обліку комп’ютерних пристроїв на підприємстві. Отже основний функціонал системи повинен бути пов’язаний із обліком пристроїв, а саме документування їх переміщень та будь-яких інших подій що їх безпосередньо стосуються. Також система повинна вести документування характеристик пристроїв та їх зміни.

### **1.3.3. Характеристики системи**

Система повинна надавати такий функціонал:

REQ-1: Надання переліку пристроїв підприємства в залежності від рівня доступу користувача.

REQ-2: Надання переліку параметрів як всіх пристроїв, так і кожного окремо.

REQ-3: Надання переліку всіх переміщень пристроїв за останній час, згідно із правами доступу користувача.

REQ-4: Занесення інформації в БД щодо переміщення пристрою.

REQ-5: Занесення інформації в БД щодо ремонту пристрою.

REQ-6: Занесення інформації в БД щодо зміни характеристик пристрою.

### **1.3.4. Вимоги зовнішніх інтерфейсів**

Для коректної роботи застосунку необхідно:

* ОС Windows 7, або вище;
* Microsoft .NET Framework 4.5.2, або вище;

MS SQL Server 2017

### **1.3.5. Нефункціональні вимоги до проекту**

У застосунку реалізована система користувачів. Кожен користувач має певну роль та рівень доступу, відповідний до ролі.

У системі передбачено такі типи користувачів:

* SuperAdmin – має абсолютний доступ до функціоналу застосунку. Даний тип користувачів передбачено для власника компанії, його заступника та працівників, що мають найвищий рівень доступу.
* Адміністратор – має повний доступ до ведення обліку пристроїв у межах його відділу. Даний тип користувачів передбачено для керівників відділів, бухгалтерів та інших менеджерів.
* Користувач – має доступ до обмеженого функціоналу застосунку, а саме до реєстрації переміщення пристрою, інформування керівника відділу чи менеджера про поломку пристрою та інше. Даний тип користувачів передбачено для працівників.

На рис. 1.1. зображено діаграму прецедентів системи, для описаних типів користувачів.

Система реалізована таким чином, що після входу в систему користувач має доступ до тих функцій, що відповідають його рівню доступу. Всі інші пункти меню для нього приховані.

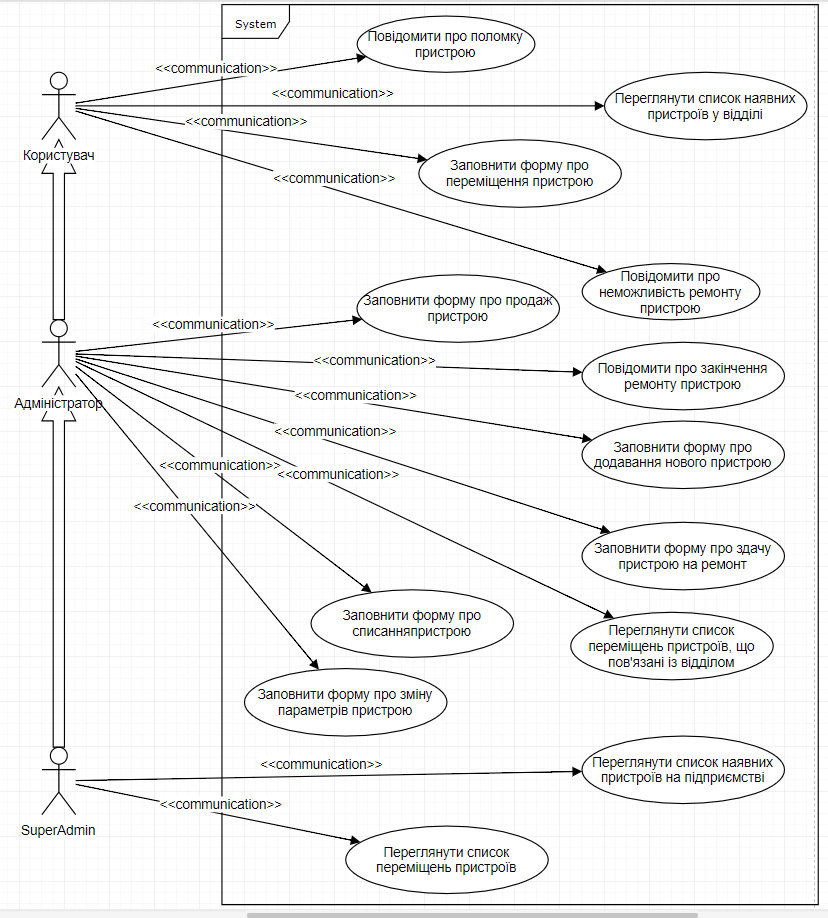


Рис. 1.1. Діаграма прецедентів розробленої системи

## **1.4. Вибір засобів розроблення системи**

Систему було розроблено із використанням MS SQL Server 2017, для роботи із базою даних. Для проектування системи було використано програмний продукт ERwin Data Modeler. Клієнтських застосунок було розроблено за допомогою мови C# та технології WPF. Для зручності роботи із вище вказаною технологією було вирішено використовувати фреймворк Caliburn.Micro. Робота із сервером БД на стороні клієнтського застосунку була реалізована за допомогою використання технології ADO.NET.

# РОЗДІЛ 2. ПРОЕКТУВАННЯ БАЗИ ДАНИХ

## **2.1. Концептуальне моделювання предметної області**

Відповідно до сформованих вимог до системи було визначено основні сутності у БД, їх атрибути та зв’язки між ними.

Пристрій (device) – комп’ютерний пристрій, що є одиницею обліку. Він має певний тип та параметри, що відповідають даному типу. Також вказується його статус, що вказує на справність пристрою, на наявність його на підприємстві та інше.

Параметр пристрою (attribute) – певна технічна характеристика пристрою. Параметр характеризується значенням, типом та одиницею виміру.

Переміщення (transfer) – сутність, що вказує на зміну розміщення пристрою. Переміщення має власний тип: «Переміщення», «Купівля», «Списання», тощо. За допомогою даних сутностей у базі даних зберігається вся інформація про будь-які дії на підприємстві, що пов’язані із пристроями.

Відділ (department) – певна структурна одиниця підприємства, що має власні ресурси, персонал, менеджерів та пристрої.

Працівник (employee) – працівник підприємства, що здійснює вхід в систему за відповідним йому користувачем.

Користувач (user) – користувач системи, що має певну роль та відповідний рівень доступу.

Сутності мають такі зв’язки між собою:

Пристрій – Параметри пристрою (1 : n). Один пристрій має багато параметрів. Параметри можна вказувати при додаванні пристрою, та змінювати їх значення, у випадку зміни характеристик пристрою (покращення/погіршення пристрою).

Пристрій – Переміщення (1 : n). Із одним пристроєм можна провести, фактично, безліч операцій переміщення, але кожне переміщення, інформація про яке зберігається у БД, може стосуватись лише одного пристрою.

Відділ – Пристрої (1 : n). У відділі знаходяться багато пристроїв, та в той час кожен пристрій може знаходитись лише в одному відділі.

Відділи – Переміщення (m : n). Для кожного переміщення записуються відділи, що прийняли участь в цьому переміщенні, а також роль кожного відділу в даному переміщенні.

Відділ – Працівники (1 : n). В одному відділі можуть працювати одночасно багато працівників, проте один працівник в один момент часу може працювати виключно в одному відділі.

Користувач – Працівник (1 : 1). Кожен працівник може входити до системи лише під одним користувачем, що дає змогу однозначно його ідентифікувати.

Користувачі – Ролі (m : n). За одним користувачем може бути закріплено декілька різних ролей.

Як результат проектування було створення концептуальну модель бази даних (рис. 2.1).

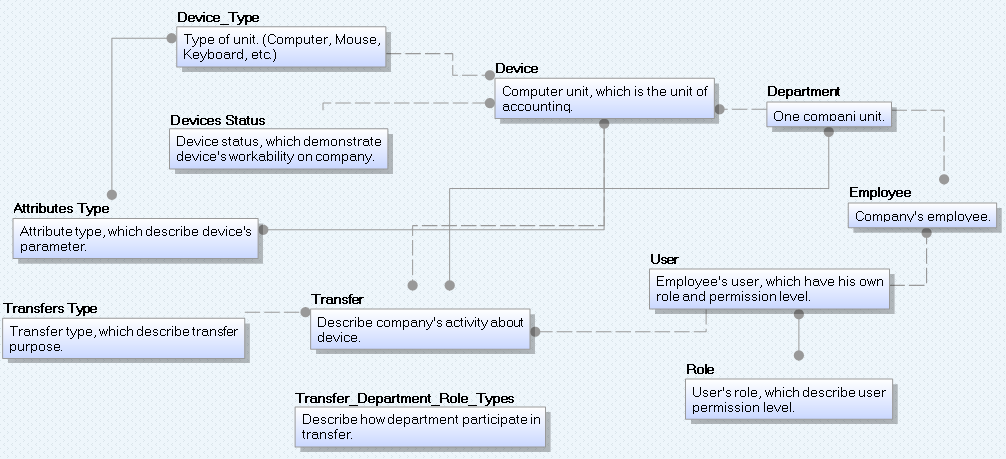


Рис. 2.1. Концептуальна модель бази даних

Після створення концептуальної моделі бази даних. Була створена логічна модель (рис. 2.2). До всіх сутностей було додано необхідні атрибути, зовнішні ключі та первинні ключі із автоінкрементом. Також було створено сутності-словники, що міститимуть відповідні переліки можливих типів, статусів та інше. Це такі сутності: тип пристрою (Device\_Type), статус пристрою (Devices Status), тип атрибуту (Attributes Type), тип переміщення (Transfer Type), роль відділу у переміщені (Transfer\_Department\_Role\_Types).

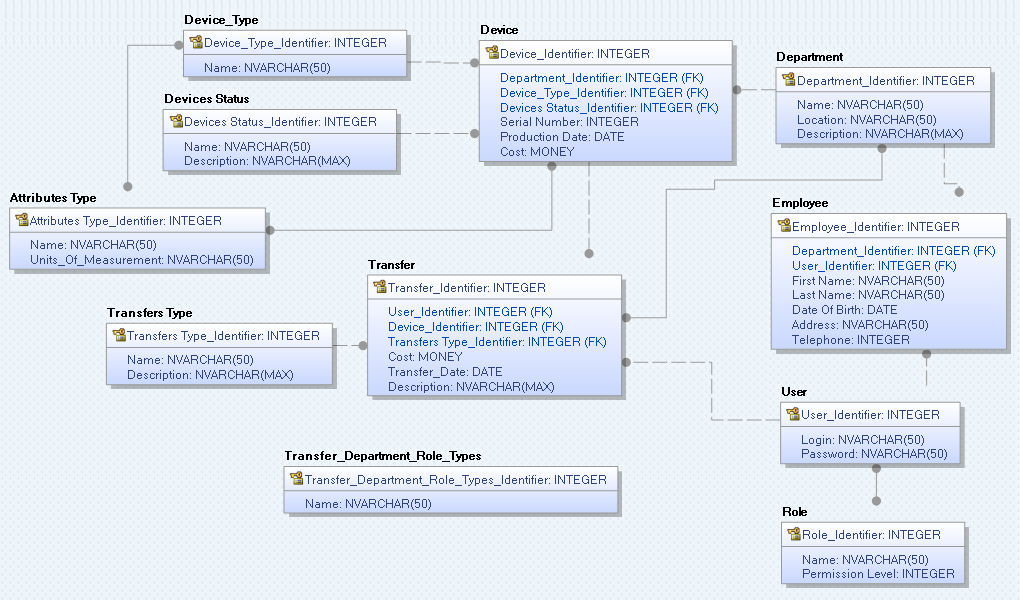


Рис. 2.2. Логічна модель бази даних

Після створення логічної моделі було створено фізичну модель бази даних (рис. 2.3). Всі логічні зв’язки типу багато до багатьох були перетворені, й представлені як додаткові таблиці, що містять первинні ключі на відповідні таблиці. Також було проведено перевірку створеної моделі засобами ERwin Data Modeler (рис. 2.4). Як показали результати дана модель є валідною.

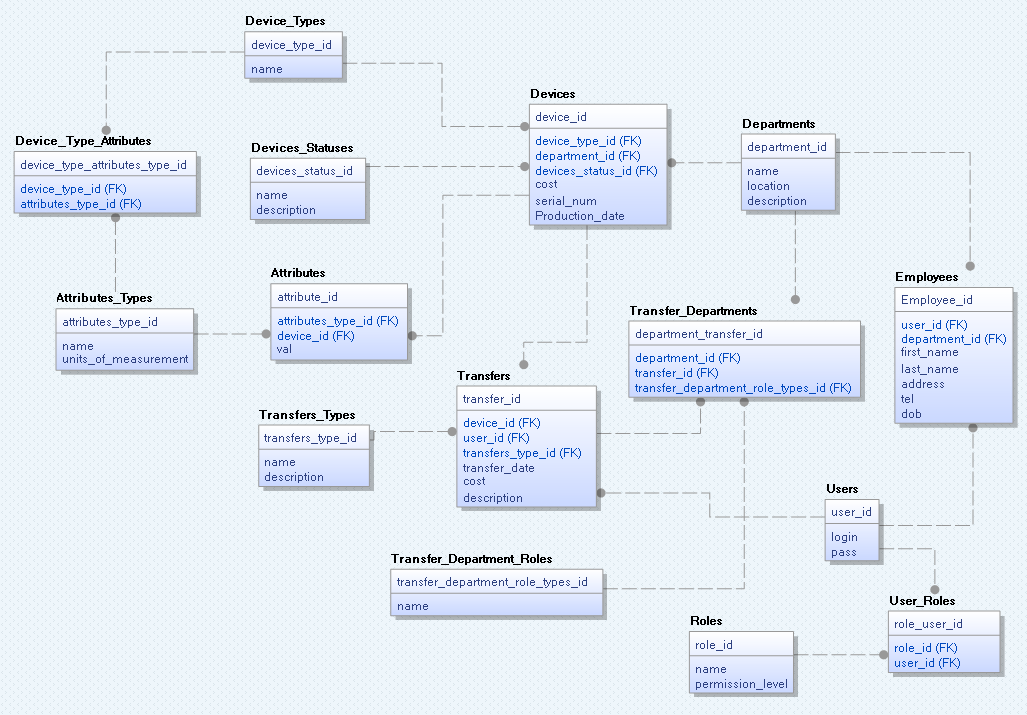


Рис. 2.3. Фізична модель бази даних

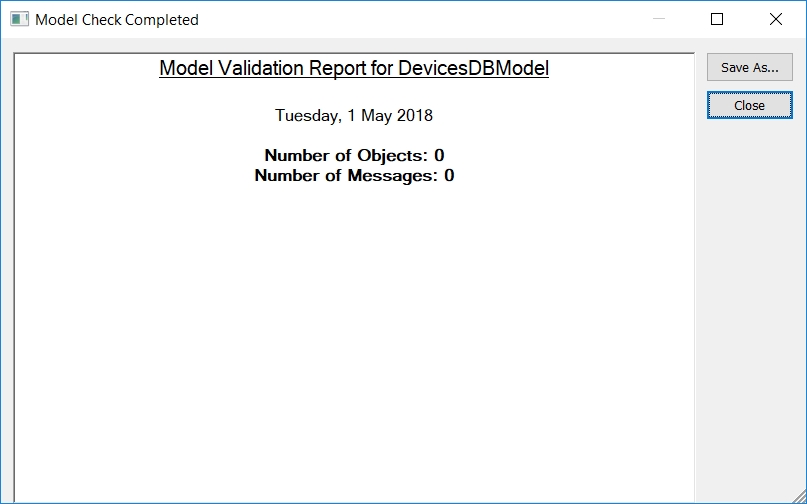


Рис. 2.4. Результати перевірки правильності створеної моделі

## **2.2. Логічне проектування бази даних**

Для всіх первинних ключів бази даних було організовано автоінкрементацію. Всі атрибути, що відповідають за імена, назви та інші текстові дані, що не потребують великого розміру, мають тип nvarchar(50). Поля, що відповідають за детальний опис мають тип nvarchar(max), для зберігання даних великого розміру. За допомогою ERwin Data Modeler було згенеровано скрипт на створення спроектованої бази даних. Код скрипту можна проглянути у додатку А.

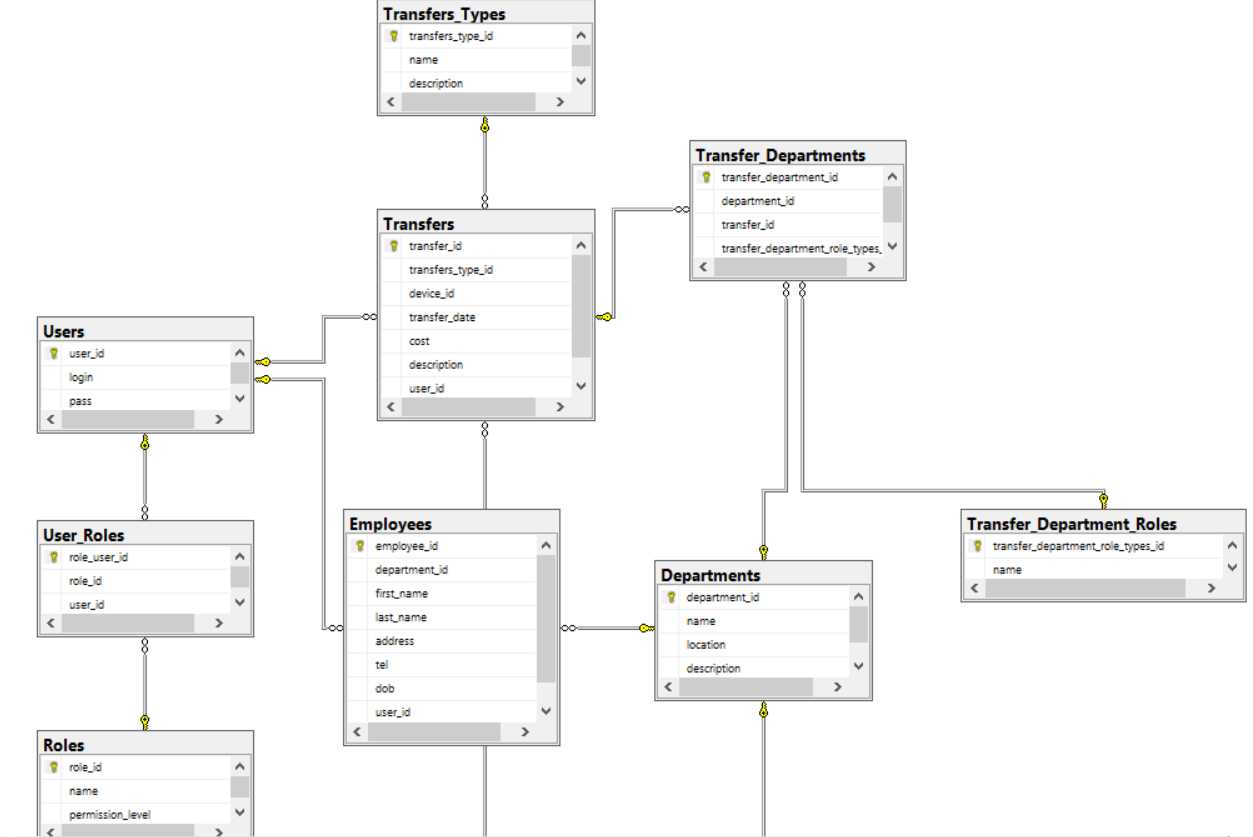


Рис. 2.5. Діаграма бази даних у MS SQL Server ч.1

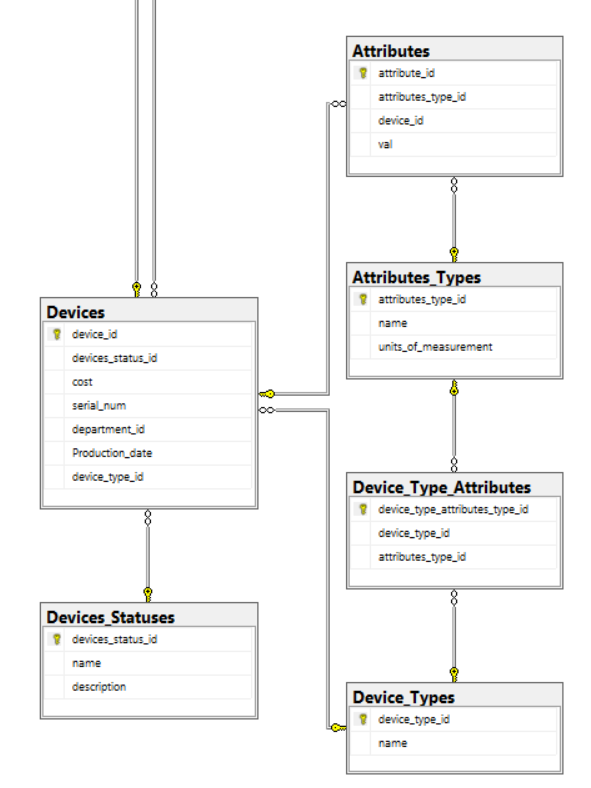


Рис. 2.6. Діаграма бази даних у MS SQL Server ч. 2

## **2.3. Проектування типових запитів, транзакцій та представлень**

Для бази даних були створені також збережені процедури, представлення та функції.

Збережені процедури:

* Для додавання записів про новий пристрій у базу даних. Дана процедура в якості параметрів приймає дані про новий пристрій, список атрибутів пристрою, що передається у вигляді таблиці користувацького типу даних, а також дані про саме переміщення, що описує придбання даного пристрою підприємством. Виконується запис всієї інформації однією транзакцією із рівнем ізоляції READ COMMITTED.
* Для додавання записів про переміщення пристрою. Дана процедура в якості параметрів приймає ідентифікатор пристрою, ідентифікатор відділу, куди було перенесено пристрій, а також додаткову інформацію про переміщення: опис, дату переміщення, ідентифікатор користувача, що заніс дані. Виконується запис всієї інформації однією транзакцією із рівнем ізоляції READ COMMITTED.
* Для додавання запису про здачу пристрою в ремонт. Дана процедура в якості параметрів приймає ідентифікатор пристрою та вартість передоплати ремонту, а також додаткову інформацію про переміщення: опис, дату переміщення, ідентифікатор користувача, що заніс дані. Виконується запис всієї інформації однією транзакцією із рівнем ізоляції READ COMMITTED.
* Для додавання запису про закінчення ремонту пристрою. Дана процедура в якості параметрів приймає ідентифікатор пристрою та вартість ремонту, а також додаткову інформацію про переміщення: опис, ідентифікатор користувача, що заніс дані. Виконується запис всієї інформації однією транзакцією із рівнем ізоляції READ COMMITTED.
* Для додавання запису про продаж пристрою. Дана процедура в якості параметрів приймає ідентифікатор пристрою та суму коштів за які було його продано, а також додаткову інформацію про переміщення: опис, дату переміщення, ідентифікатор користувача, що заніс дані. Виконується запис всієї інформації однією транзакцією із рівнем ізоляції READ COMMITTED.
* Для додавання запису про списання пристрою. Дана процедура в якості параметрів приймає ідентифікатор пристрою, а також додаткову інформацію про переміщення: опис, дату переміщення, ідентифікатор користувача, що заніс дані. Виконується запис всієї інформації однією транзакцією із рівнем ізоляції READ COMMITTED.
* Для встановлення статусу пристрою «Потребує ремонту». Дана процедура в якості параметрів приймає ідентифікатор пристрою.
* Для встановлення статусу пристрою «Не підлягає ремонту». Дана процедура в якості параметрів приймає ідентифікатор пристрою.
* Для додавання запису про зміну певного параметра пристрою. Дана процедура в якості параметрів приймає ідентифікатор пристрою, ідентифікатор атрибуту та нове значення атрибуту, а також додаткову інформацію про переміщення: вартість, опис, дату переміщення, ідентифікатор користувача, що заніс дані. Виконується запис всієї інформації однією транзакцією із рівнем ізоляції READ COMMITTED.

Представлення:

* Для перегляду інформації про переміщення. Дане представлення об’єднує дані із таблиць, що містять інформацію про переміщення, та формує вихідну таблицю. У вихідній таблиці міститься інформація про тип переміщення, пристрій та його серійний номер, відділи, що прийняли участь у переміщенні, вартість переміщення, дату та інше.
* Для перегляду інформації про пристрої. Дане представлення об’єднує дані із таблиць, що містять інформацію про пристрої, за виключенням їх параметрів. У вихідній таблиці міститься інформація про тип пристрою, його статус, відділ, в якому зараз знаходиться пристрій, серійний номер пристрою, а також дату виробництва та вартість.
* Для перегляду інформації про параметри пристрою. Дане представлення об’єднує дані із таблиць, що містять інформацію про параметри пристроїв. У вихідній таблиці міститься інформація про пристрій, серійний номер пристрою, відділ, в якому зараз знаходиться пристрій, а також тип атрибуту, значення даного параметра для конкретного приладу та одиниці виміру даного параметру.

Функції:

* Для вибору інформації із про пристрої відповідно до рівня доступу користувача. Функція використовує описане вище представлення, та на основі нього вибирає інформацію, що стосується конкретного користувача. Параметром приймає ідентифікатор користувача.
* Для вибору інформації про параметри пристроїв відповідно до рівня доступу користувача. Функція використовує описане вище представлення, та на основі нього вибирає інформацію, що стосується конкретного користувача. Параметром приймає ідентифікатор користувача.
* Для вибору інформації про переміщення пристроїв відповідно до рівня доступу користувача. Функція використовує описане вище представлення, та на основі нього вибирає інформацію, що стосується конкретного користувача. Параметром приймає ідентифікатор користувача.
* Для вибору параметрів конкретного пристрою. Функція використовує описане вище представлення для вибору параметрів пристроїв, та повертає таблицю із даними про параметрами вказаного пристрою.

# РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОГРАМИ РОБОТИ З БАЗОЮ ДАНИХ

## **3.1. Реалізація доступу до даних**

Доступ до бази даний клієнтського застосунку реалізовано за допомогою технології ADO.NET. Дана технологія була обрана по причині того, що вона дає змогу програмісту чіткіше контролювати та опрацьовувати запити, що надходять до серверу бази даних. А також вона забезпечує зручну роботу із користувацькими типами даних, що були створені в базі даних.

Якщо порівнювати дану технологію із іншими, наприклад, Entity Framework, то ADO.NET надає більше контролю та можливість простіше використовувати збережені процедури, функції та представлення, особливо у випадку, якщо вони використовують користувацькі типи, що були створені в базі даних. А для організації бази даних, як було описано вище, було створено саме такий користувацький тип даних.

Діаграма розгортання системи подана на рис. 3.1. За основну логіку клієнтського застосунку відповідає модуль Models, саме він за допомогою ADO.NET з’єднується із базою даних, що розміщено на MS SQL Server.

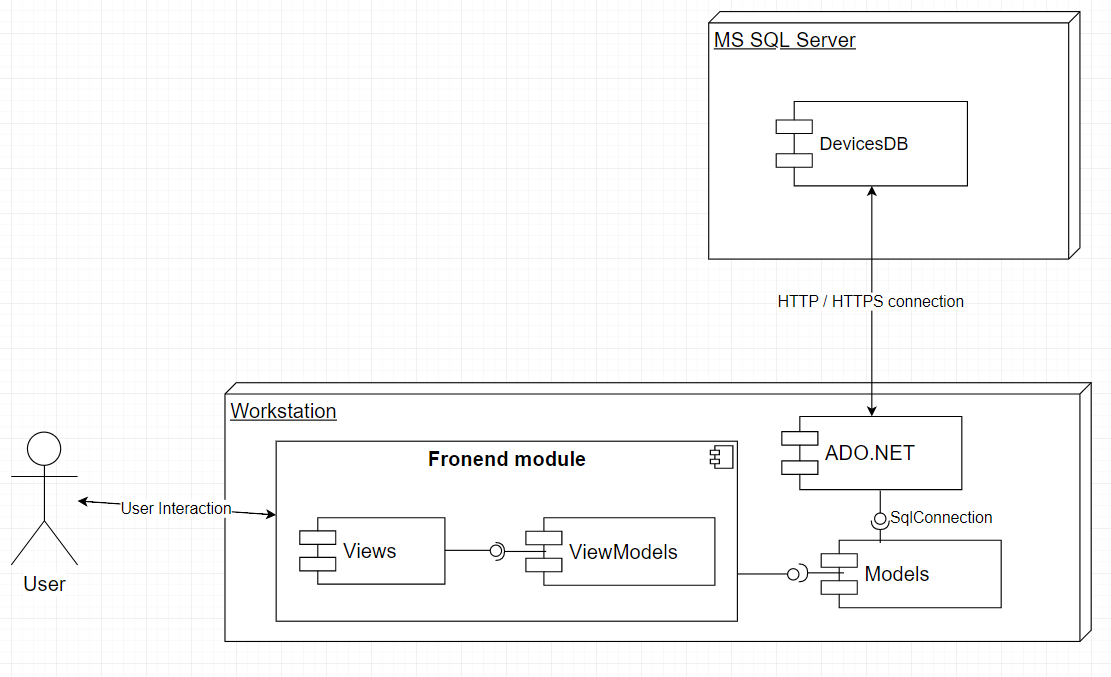


Рис. 3.1. Діаграма розгортання системи

## **3.2. Реалізація функціональних характеристик системи**

Для реалізації функціональних характеристик системи було використано вище описані збережені процедури, представлення та функції. Обмеження користувачів відповідно до їх рівнів доступу відбувається за допомогою як користувацького застосунку, так й логіки бази даних. Наприклад, створені функції повертають таблиці даних вміст яких відповідає рівню доступу користувача, але обмеження по використанню функціоналу системи відбувається безпосередньо на стороні клієнтського застосунку. Таким чином при вході в систему користувач, що має тип «Користувач», не матиме змоги зробити нічого, окрім відведених йому функцій по причині відсутності пунктів меню, із забороненим функціоналом.

Виконання функціональних вимог REQ-1, REQ-2 та REQ-3 реалізовано за допомогою використання створених представлень та функцій, а для виконання функціональних вимог REQ-4, REQ-5 та REQ-6 використано створені збережені процедури. Збережені процедури були створені за допомогою транзакцій, що забезпечує цілісність їх виконання. Також виконання створених функцій на стороні клієнта було організовано за допомогою транзакцій, з аналогічною метою.

## **3.3. Опис роботи системи**

.

# ВИСНОВКИ

.

# СПИСОК ЛІТЕРАТУРИ



# ДОДАТОК А. Скрипт створення БД