Міністерство освіти і науки України  
Національний технічний університет України

«Київський політехнічний інститут ім. І. Сікорського»

Навчально-науковий інститут атомної та теплової енергетики

Кафедра цифрових технологій в енергетиці

Лабораторна робота №3

з дисципліни «Проектування інформаційних систем»

Тема «Розробка структури бази даних і контрольного прикладу для АРМ ІС»

Варіант №16

Виконав:

студент 4-го курсу, НН ІАТЕ

групи ТР-23

Ровний Григорій Олександрович

Перевірив: ст. вик. Гурін А. Л.

КИЇВ 2025

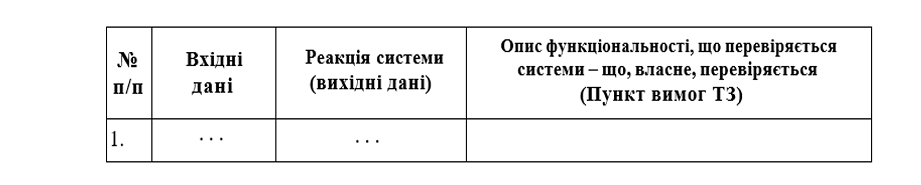
**Мета роботи:**

Закріпити навички створення структури бази даних

**Поставлене завдання:**

Розробити звіт, що містить структуру бази даних та контрольний приклад для АРМ ІС. Повинні бути визначені: - склад таблиць: з кожної таблиці – поля, розмірність полів, тип полів; - взаємозв’язок таблиць: ключові атрибути; - структура: намалювати структуру бази даних.

Контрольний приклад повинен забезпечити перевірку функціонування АРМ ІС, у тому числі дій, що виконуються користувачами в процесі експлуатації, та реакції АРМ на дії користувачів. Повинен бути у вигляді таблиці:



**Варіант завдання:**

**16. ІС обліку абонентів стільникового зв'язку (деякого провайдера)**

Зразковий зміст Бази даних:

— інформація про абонента, паспортні дані та ін. інформація;

— номер телефону, параметри номера, сім-карти...;

— параметри обладнання клієнта (марка телефону...);

— чи є договір на підтримку, ...;

— інше.

**Результат виконання роботи**

**Звязки між таблицями:**

Subscribers — Contracts (1 до багатьох (1:N))

Один абонент може мати декілька контрактів.

Contracts.subscriber\_id є зовнішнім ключем на Subscribers.subscriber\_id.

Subscribers — Payments (1 до багатьох (1:N))

Один абонент може здійснювати декілька оплат.

Payments.subscriber\_id є зовнішнім ключем на Subscribers.subscriber\_id.

Subscribers → SupportRequests (1 до багатьох (1:N))

Один абонент може створити декілька заявок у підтримку.

SupportRequests.subscriber\_id є зовнішнім ключем на Subscribers.subscriber\_id.

Subscribers — Equipment (1 до багатьох (1:N))

Equipment.subscriber\_id є зовнішнім ключем на Subscribers.subscriber\_id.

Contracts — Payments (непрямий зв’язок через абонента)

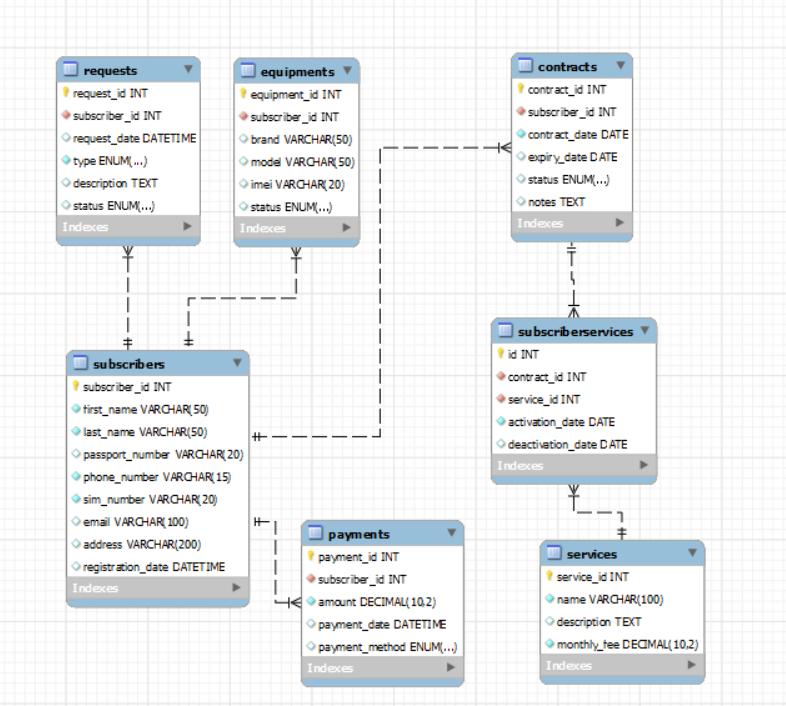
Оплати можуть відноситися до контрактів, але наразі ми зберігаємо їх лише через subscriber\_id.

+ можна додати прямий зв’язок (Payments.contract\_id).

Contracts — SupportRequests (необов’язковий зв’язок)

Запити підтримки можуть стосуватися конкретного контракту.

**Схема Бази Даних в MySQL**



**Структура таблиць:**

|  |  |  |  |
| --- | --- | --- | --- |
| Subscribers | subscriber\_id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| first\_name | VARCHAR(50) | NOT NULL |
| last\_name | VARCHAR(50) | NOT NULL |
| passport\_number | VARCHAR(20) | NULL |
| phone\_number | VARCHAR(15) | UNIQUE, NOT NULL |
| sim\_number | VARCHAR(20) | UNIQUE, NOT NULL |
| email | VARCHAR(100) | NULL |
| address | VARCHAR(200) | NULL |
| registration\_date | DATE | DEFAULT (CURRENT\_DATE) |
| Contracts | contract\_id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| subscriber\_id | INT | FOREIGN KEY → Subscribers(subscriber\_id), NOT NULL |
| contract\_number | VARCHAR(50) | UNIQUE, NOT NULL |
| start\_date | DATE | NOT NULL |
| end\_date | DATE | NULL |
| status | ENUM('active','inactive','terminated') | DEFAULT 'active' |
| Payments | payment\_id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| subscriber\_id | INT | FOREIGN KEY → Subscribers(subscriber\_id), NOT NULL |
| amount | DECIMAL(10,2) | NOT NULL |
| payment\_date | DATETIME | DEFAULT CURRENT\_TIMESTAMP |
| payment\_method | ENUM('card','cash','bank\_transfer','other') | DEFAULT 'card' |
| SupportRequests | request\_id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| subscriber\_id | INT | FOREIGN KEY → Subscribers(subscriber\_id), NOT NULL |
| request\_date | DATETIME | DEFAULT CURRENT\_TIMESTAMP |
| issue\_description | TEXT | NOT NULL |
| status | ENUM('open','in\_progress','closed') | DEFAULT 'open' |
| Equipment | equipment\_id | INT | PRIMARY KEY, AUTO\_INCREMENT |
| subscriber\_id | INT | FOREIGN KEY → Subscribers(subscriber\_id), NOT NULL |
| device\_type | VARCHAR(50) | NOT NULL |
| brand | VARCHAR(50) | NULL |
| model | VARCHAR(50) | NULL |
| serial\_number | VARCHAR(50) | UNIQUE, NOT NULL |
| purchase\_date | DATE | NULL |

**Контрольні приклади:**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Вхідні дії** | **Реакція системи** | **Функціональність яка перевіряється** |
| **1** | Користувач вводить нові дані абонента (ім’я, прізвище, телефон, SIM-номер, паспортні дані) та натискає “Зберегти” | Дані успішно додаються в таблицю Subscribers | Реєстрація нового абонента |
| **2** | Користувач реєструє нове обладнання з вказанням типу, марки та моделі, прив’язує до абонента | Запис зберігається в таблиці Equipment, створюється зв’язок із абонентом | Облік обладнання абонентів |
| **3** | Оператор вносить договір (дата, умови, прив’язка до абонента) | У таблиці Contracts зберігається запис із зовнішнім ключем на Subscribers | Облік договорів |
| **4** | Абонент здійснює платіж через картку | У таблиці Payments зберігається запис із сумою, методом оплати та датою | Облік та контроль платежів |
| **5** | Адміністратор виконує пошук абонента за номером телефону | Система повертає дані з таблиці Subscribers, а також пов’язані договори, обладнання та платежі | Пошук і перегляд даних абонентів |
| **6** | Користувач формує звіт по платежах за місяць | Система генерує агреговані дані з таблиці Payments | Формування звітів |

**Висновок:** В результаті виконання лабораторної роботи було досягнуто та на практиці закріплено навички створення простої бази даних. Було розроблено структуру бази даних для заданої АPM. Структура включає таблиці з описом полів, а також визначено взаємозв’язки між ними. Кожна таблиця має відповідні атрибути та первинні ключі, а також встановлені зв’язки між ними, що забезпечує ефективну обробку та збереження даних. Також додатково описано кілька тестових сценаріїв із вхідними діями користувачів та реакцією системи, що дає можливість оцінити працездатність системи та її реакцію на реальні дії користувачів.

Код БД (MySQL):

CREATE DATABASE IF NOT EXISTS subscriber\_management

CHARACTER SET utf8mb4

COLLATE utf8mb4\_unicode\_ci;

USE subscriber\_management;

-- Абоненти

CREATE TABLE Subscribers (

subscriber\_id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

passport\_number VARCHAR(20),

phone\_number VARCHAR(15) UNIQUE NOT NULL,

sim\_number VARCHAR(20) UNIQUE NOT NULL,

email VARCHAR(100),

address VARCHAR(200),

registration\_date DATETIME DEFAULT CURRENT\_TIMESTAMP

);

-- Договори

CREATE TABLE Contracts (

contract\_id INT AUTO\_INCREMENT PRIMARY KEY,

subscriber\_id INT NOT NULL,

contract\_date DATE NOT NULL,

expiry\_date DATE,

status ENUM('active','expired','terminated') DEFAULT 'active',

notes TEXT,

FOREIGN KEY (subscriber\_id) REFERENCES Subscribers(subscriber\_id)

ON DELETE CASCADE

);

-- Послуги

CREATE TABLE Services (

service\_id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100) NOT NULL,

description TEXT,

monthly\_fee DECIMAL(10,2) NOT NULL

);

-- Абонентські послуги (зв’язок договорів з послугами)

CREATE TABLE SubscriberServices (

id INT AUTO\_INCREMENT PRIMARY KEY,

contract\_id INT NOT NULL,

service\_id INT NOT NULL,

activation\_date DATE NOT NULL,

deactivation\_date DATE,

FOREIGN KEY (contract\_id) REFERENCES Contracts(contract\_id)

ON DELETE CASCADE,

FOREIGN KEY (service\_id) REFERENCES Services(service\_id)

ON DELETE CASCADE

);

-- Обладнання

CREATE TABLE Equipments (

equipment\_id INT AUTO\_INCREMENT PRIMARY KEY,

subscriber\_id INT NOT NULL,

brand VARCHAR(50),

model VARCHAR(50),

imei VARCHAR(20) UNIQUE,

status ENUM('active','lost','broken') DEFAULT 'active',

FOREIGN KEY (subscriber\_id) REFERENCES Subscribers(subscriber\_id)

ON DELETE CASCADE

);

-- Платежі

CREATE TABLE Payments (

payment\_id INT AUTO\_INCREMENT PRIMARY KEY,

subscriber\_id INT NOT NULL,

amount DECIMAL(10,2) NOT NULL,

payment\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

payment\_method ENUM('card','cash','bank\_transfer','other') DEFAULT 'card',

FOREIGN KEY (subscriber\_id) REFERENCES Subscribers(subscriber\_id)

ON DELETE CASCADE

);

-- Звернення (сапорт)

CREATE TABLE Requests (

request\_id INT AUTO\_INCREMENT PRIMARY KEY,

subscriber\_id INT NOT NULL,

request\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

type ENUM('technical','billing','general') NOT NULL,

description TEXT,

status ENUM('open','in\_progress','closed') DEFAULT 'open',

FOREIGN KEY (subscriber\_id) REFERENCES Subscribers(subscriber\_id)

ON DELETE CASCADE

);