**Лабораторна робота №1**

**з дисципліни «Бази даних»**

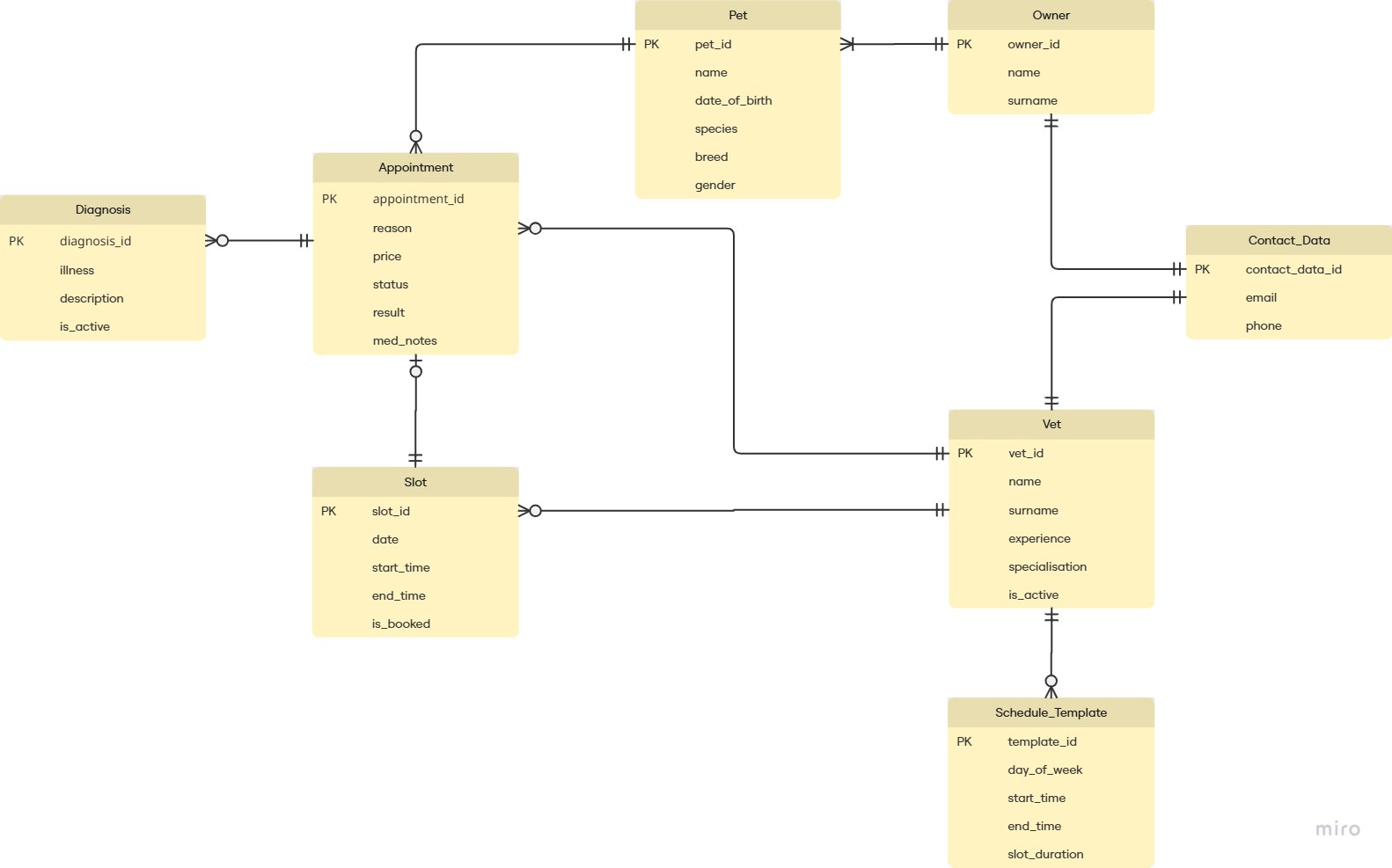
**Виконали:**

Олексійчук Станіслав ІМ-43,

Хубеджева Єлизавета ІМ-43

Київ 2025

**ER-діаграма**

****

**Вимоги**

1. **Суть системи**

Система побудована у вигляді структури ветеринарної клініки з функцією запису на прийоми. Домашні улюбленці можуть з допомогою своїх власників записуватися на вільні «слоти» часу ветеринара задля прийому. Після прийому пацієнт (тварина) отримує обов’язково результат та короткий опис прийому, а в разі негативного результату – ще й діагноз.

1. **Потреби зацікавлених сторін**

Pet (домашній улюбленець):

* Пошук та опис ветеринара,
* Пошук вільних слотів для запису,
* Запис на прийом,
* Отримання та перегляд результатів прийому.

Vet (ветеринар):

* Можливість складати та змінювати шаблон розкладу,
* Приймати у вільні «слоти» нових пацієнтів,
* Встановлювати ціну прийому,
* Отримувати дані тварини та його хазяїна,
* Встановлювати результати прийому та писати діагноз.

1. **Дані для зберігання**

* Інформація про пацієнта (Pet), хазяїна (Owner) та ветеринара (Vet);
* Шаблони розкладу та створені за цими шаблонами «слоти» прийому;
* Дані прийому – ціна, час проведення, статус, результати, сторони проведення.
* Відсутність чи наявність діагнозу (за наявності є назва хвороби, її відсутність чи присутність, опис діагнозу).

1. **Бізнес-правила**

***1) Правила Керування Розкладом та Доступністю***

* Генерація Слота: Доступні слоти (Slot) генеруються автоматично застосунком на основі правил, визначених у Schedule\_Template (Шаблоні Графіка) для кожного дня тижня.
* Унікальність Графіка Ветеринара: Один лікар (Vet) може мати лише одне правило (Schedule\_Template) на кожен день тижня. (Наприклад, не можна мати два різні графіки на Понеділок).
* Контроль Активності: Якщо поле Vet.is\_active встановлено як False, генерація нових Slot для цього лікаря не відбувається, незалежно від його шаблонів.

***2) Правила Бронювання та Цілісності Даних***

* Запобігання Подвійному Бронюванню: Один Slot може бути пов'язаний лише з нулем або одним Appointment (Прийомом). Після створення прийому, Slot.is\_booked змінюється на True.
* Визначення Часу Прийому: Фактичні дата та час прийому (Appointment) завжди витягуються з Slot, на який посилається Appointment.slot\_id.
* Цілісність Лікаря: Vet, призначений для Appointment, повинен збігатися з Vet, який є власником відповідного Slot.
* Обов'язковість Зв'язку: Кожен Appointment має бути прив'язаний до одного Slot, одного Vet й одного Pet.

***3) Правила Обліку Клієнтів та Медичної Історії***

* Власність Тварини: Кожна тварина (Pet) може належати лише одному власнику (Owner).
* Медична Документація: Кожен Diagnosis (Діагноз), якщо він існує, повинен бути пов'язаний з одним конкретним Appointment (Прийомом).
* Множинність Діагнозів: Один Appointment може мати один, кілька або жодного Diagnosis.
* Наявність Діагнозу: у залежності від статусу прийому та його результату Діагноз (Діагнози) існуватимуть або не існуватимуть.

**Сутності та їх зв’язки**

1. **Сутності**

***Contact\_Data (Контактні Дані)***

* **contact\_data\_id:** Унікальний ідентифікатор конкретного контакту. **(Первинний Ключ, PK)**.
* email: Електронна пошта для зв'язку.
* phone: Номер телефону для зв'язку.

***Owner (Власник)***

* **owner\_id:** Унікальний ідентифікатор власника. **(Первинний Ключ, PK)**.
* name: Ім'я власника.
* surname: Прізвище власника.
* contact\_data\_id: Посилання на контактні дані. **(Зовнішній Ключ, FK)**.

***Pet (Домашній улюбленець)***

* **pet\_id:** Унікальний ідентифікатор тварини (медична картка). **(Первинний Ключ, PK)**.
* name: Кличка тварини.
* date\_of\_birth: Дата народження тварини.
* species: Вид тварини.
* breed: Порода тварини.
* gender: Стать тварини.
* owner\_id: Посилання на власника тварини. **(Зовнішній Ключ, FK)**.

***Vet (Ветеринар)***

* **vet\_id:** Унікальний ідентифікатор лікаря. **(Первинний Ключ, PK)**.
* name: Ім'я лікаря.
* surname: Прізвище лікаря.
* experience: Досвід роботи (стаж).
* specialisation: Спеціалізація лікаря.
* is\_active: Статус активності; визначає, чи повинен генеруватися графік (True/False).
* contact\_data\_id: Посилання на контактні дані. **(Зовнішній Ключ, FK)**.

***Schedule\_Template (Шаблон Графіку)***

* **template\_id:** Унікальний ідентифікатор правила графіка. **(Первинний Ключ, PK)**.
* day\_of\_week: День тижня, до якого застосовується правило.
* start\_time: Типовий час початку зміни.
* end\_time: Типовий час закінчення зміни.
* slot\_duration: Стандартна тривалість одного прийому (у хвилинах).
* vet\_id: Посилання на лікаря, якому належить правило. **(Зовнішній Ключ, FK)**.

***Slot (Доступний Слот)***

* **slot\_id:** Унікальний ідентифікатор блоку часу. **(Первинний Ключ, PK)**.
* date: Дата слота.
* start\_time: Фактичний час початку слота.
* end\_time: Фактичний час закінчення слота.
* is\_booked: Статус: True (зайнято) або False (вільно).
* vet\_id: Посилання на лікаря, який обслуговує цей час. **(Зовнішній Ключ, FK)**.

***Appointment (Прийом)***

* **appointment\_id:** Унікальний ідентифікатор факту прийому. **(Первинний Ключ, PK)**.
* reason: Описана причина запису.
* price: Загальна вартість послуг.
* status: Поточний стан прийому (SCHEDULED, IN\_PROGRESS, COMPLETED).
* result: Загальний медичний висновок.
* med\_notes: Детальні нотатки лікаря.
* pet\_id: Посилання на тварину, яка прийшла/прийде на прийом. **(Зовнішній Ключ, FK)**.
* vet\_id: Пряме посилання на лікаря, який проводив/проводитиме прийом. **(Зовнішній Ключ, FK)**.
* slot\_id: Посилання на заброньований блок часу. **(Зовнішній Ключ, FK)**.

***Diagnosis (Діагноз)***

* **diagnosis\_id:** Унікальний ідентифікатор поставленого діагнозу. **(Первинний Ключ, PK)**.
* illness: Назва встановленої хвороби або стану.
* description: Детальний опис стану.
* is\_active: Статус діагнозу (True – хвороба з’явилася/триває, False – хвороба вилікувана).
* appointment\_id: Посилання на прийом, під час якого поставлено діагноз. **(Зовнішній Ключ, FK)**.

1. **Зв’язки**

* **Owner ↔ Contact\_Data (1 : 1):** Один власник може мати один номер телефону та одну електронну пошту.
* **Owner ↔ Pet (1 : 1..N):** Один власник може мати одного або багато домашніх улюбленців.
* **Pet ↔ Appointment (1 : 0..N):** Один домашній улюбленець може не мати жодного прийому або мати їх багато.
* **Appointment ↔ Diagnosis (1 : 0..N):** Після одного прийому може бути не виданий діагноз або видані їх кілька.
* **Vet ↔ Contact\_Data (1 : 1):** Один ветеринар може мати один номер телефону та одну електронну пошту.
* **Vet ↔ Schedule\_Template (1 : 0..N):** Один ветеринар може не мати шаблон розкладу або мати їх декілька (до 7 – кількість днів тижня).
* **Vet ↔ Slot (1 : 0..N):** Один ветеринар може не мати «слотів» для прийомів (vet.is\_active = False) або мати їх багато.
* **Slot ↔ Appointment (1 : 0/1):** Один «слот» може бути незаброньованим або заброньованим лише одним прийомом.
* **Vet ↔ Appointment (1 : 0..N):** У одного ветеринара може не бути записів/прийомів або він має їх декілька.

**Припущення та обмеження**

* Ми припускаємо, що кожна Тварина (Pet) має лише одного основного Власника (Owner) у системі. Модель не підтримує ідею спільного володіння або другого контактного номера чи електронної пошти.
* Ми припускаємо, що Slot.date, Slot.start\_time і Slot.end\_time є основою для часу прийому, тому використовуватимемо Slot.slot\_id як FK (Foreign Key) у таблиці Appointment, спростивши модель.
* Ми припускаємо, що всі часові дані (start\_time, end\_time) зберігаються в одному часовому поясі і немає необхідності керувати міжрегіональними часовими поясами.
* Ми припускаємо, що сутність Schedule\_Template існує виключно як конфігурація для програмного забезпечення (шаблон для подальших «слотів»). Вона не має прямого FK до Slot, оскільки ми припускаємо, що це б ускладнило зміну графіка.
* Ми припускаємо, що Vet.is\_active = False є достатнім критерієм, щоб програмне забезпечення припинило генерацію нових слотів для цього лікаря, навіть якщо в Schedule\_Template є активні записи.
* Ми припускаємо, що атрибути Appointment змінюватимуться в залежності від статусу прийому. Якщо статус SCHEDULED або IN\_PROGRESS, то result і med\_notes дорівнюють NULL, і відповідно якщо COMPLETED – мають якісь значення NOT NULL.
* Ми припускаємо, що Appointment має статус COMPLETED. У залежності від атрибуту result (DIAGNOSED/RECOVERED/HEALTHY) залежатиме існування діагнозу. Якщо
  + result = HEALTHY, діагнозу не буде, лише med\_notes;
  + result = DIAGNOSED, діагноз буде з is\_active = True, але med\_notes теж залишаються;
  + result = RECOVERED, діагноз буде з is\_active = False, але med\_notes теж залишаються.
* Ставимо обмеження, що Schedule\_Template.slot\_duration повинен бути більшим за 15 хвилин. Це запобігає створенню недійсних або нульових слотів.
* У сутностях Schedule\_Template і Slot, end\_time має бути строго пізнішим, ніж start\_time.
* Згенеровані Slot.date (доступні для бронювання) повинні бути більшими або рівними поточній даті (CURRENT\_DATE), щоб запобігти бронюванню в минулому.
* Appointment.price (ціна прийому) має DEFAULT значення, рівне 0. Виставлена ціна може бути відповідно більшою або рівною за нуль.
* Уся інформація про пацієнта (окрім breed), власника та ветеринара (окрім experience), контактні дані, інформація про діагноз (якщо він існує), інформація про «слот» та певний шаблон, згадана інформація про прийоми (окрім певних умов, вказаних при статусі та результаті прийому) має бути NOT NULL.