

Міністерство освіти і науки України  
Харківський радіотехнічний фаховий коледж  
Циклова комісія «Комп'ютерних наук»

## **КУРСОВА РОБОТА**

з навчальної дисципліни «БАЗИ ДАНИХ»  
на тему «ПРОЕКТУВАННЯ ТА РОЗРОБКА БД «ТУРИСТИЧНЕ БЮРО»

Виконала:  
студентка гр. ПІ-332  
Ширшова Марина Сергіївна  
Керівник роботи:  
Ахмедзянова О.А.

м. Харків – 2025 рік

# Харківський радіотехнічний фаховий коледж

Циклова комісія      КОМП'ЮТЕРНИХ НАУК  
Спеціальність      121 «Інженерія програмного забезпечення»  
Курс - 3      Група – ПІ – 332      Семестр - 6

## Завдання на курсову роботу

студентки *Ширшової Марини Сергіївни*

Тема курсової роботи: **«Проектування та розробка БД «Туристичне бюро»**

**ОСНОВНІ ДАНІ:** В БД необхідно зберігати та опрацьовувати інформацію про путівки, їх розклад, клієнтів та замовлення путівок клієнтами. Необхідно автоматизувати процес відображення інформації про путівки, фільтрацію цих даних за різними параметрами, наприклад напрямку, виду, дати; автоматизувати процес бронювання путівок з можливістю реєстрації нового клієнта, а також облік історії бронювання.

Склад пояснювальної записки:

Вступ

1 Аналіз предметної області і постановка задачі

2 Проектування реляційної БД

2.1 Побудова ER-діаграми

2.2 Побудова схеми реляційної моделі даних

2.3 Вибір СУБД та опис фізичної моделі даних

2.4 Ескіз інтерфейсу користувача

3 Опис програмної реалізації

3.1 Опис структури бази даних

3.2 Опис задач автоматизації та інтерфейсу користувача

Висновки

Перелік використаних джерел

Дата видачі завдання: 17 січня 2025р

Дата здачі виконаної роботи: 20 травня 2025

Керівник роботи \_\_\_\_\_ /О.А. Ахмедзянова /

Розглянуто на засіданні ЦК

Голова ЦК \_\_\_\_\_ /Федосєєва А.О./

## ЗАЛІКОВИЙ ЛИСТ

Харківський радіотехнічний фаховий коледж

Циклова комісія КОМП'ЮТЕРНИХ НАУК  
Спеціальність 121 «Інженерія програмного забезпечення»  
Курс - 3 Група – ПІ - 332 Семестр - 6

### Курсова робота

студентки *Ширшової Марини Сергіївни*

на тему «*Проектування та розробка БД «Туристичне бюро»*»

### ДОПУЩЕНИЙ ДО ЗАХИСТУ

Керівник роботи \_\_\_\_\_ /Ахмедзянова О.А. /

ОЦІНКА \_\_\_\_\_

Члени комісії: \_\_\_\_\_ /Ахмедзянова О.А./  
\_\_\_\_\_ /Радченко О.П. /  
\_\_\_\_\_ /Багацький Д.В./

« \_\_\_\_ » « \_\_\_\_\_ » 2025 р

## ЗМІСТ

|  |    |
|--|----|
| ВСТУП.....   | 5  |
| 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ.....         | 7  |
| 2 ПРОЄКТУВАННЯ БАЗИ ДАНИХ.....                               | 11 |
| 2.1 Побудова діаграм .....                                   | 11 |
| 2.2 Побудова реляційної моделі даних .....                   | 13 |
| 2.3 Вибір СУБД та опис фізичної моделі бази даних .....      | 16 |
| 2.4 Ескіз інтерфейсу користувача .....                       | 21 |
| 3 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ .....                           | 25 |
| 3.1 Опис структури бази даних .....                          | 25 |
| 3.2 Опис задач автоматизації та інтерфейсу користувача ..... | 29 |
| ВИСНОВКИ .....   | 36 |
| ПЕРЕЛІК ПОСИЛАНЬ .....                                       | 37 |

## ВСТУП

База даних (БД) — це впорядкована сукупність взаємопов'язаних даних, які зберігаються в електронному вигляді та організовані таким чином, щоб забезпечити ефективний доступ, управління, оновлення і пошук інформації. Бази даних застосовуються в найрізноманітніших сферах — від невеликих особистих проєктів до великих інформаційних систем підприємств, банків, медичних установ, освітніх платформ та державних структур. Дані в базі можуть представляти різні об'єкти: користувачів, товари, транзакції, документи, події тощо. Основна мета використання баз даних — це централізоване зберігання інформації та зручна робота з нею.

Для управління базами даних використовуються спеціальні програмні засоби, які називаються системами управління базами даних (СУБД). СУБД — це комплекс програмного забезпечення, який забезпечує створення, підтримку, обслуговування та захист бази даних. Вона дозволяє виконувати різноманітні операції з даними: зберігати, змінювати, вилучати, шукати, аналізувати та контролювати доступ до них. При цьому користувачеві або програмісту не потрібно піклуватися про технічні деталі зберігання чи структурування даних — усе це бере на себе СУБД.

Серед найбільш відомих і поширених систем управління базами даних можна назвати MySQL, PostgreSQL, Oracle Database, Microsoft SQL Server, SQLite та інші. Вони можуть мати різні характеристики залежно від призначення: одні краще підходять для великих корпоративних рішень, інші — для невеликих локальних додатків або мобільних застосунків. СУБД можуть бути реляційними (де дані організовані у вигляді таблиць із чіткими зв'язками між ними) або нереляційними (NoSQL), які краще підходять для роботи з неструктурованими або частково структурованими даними, як-от документи, графи чи великі масиви ключ-значення.

Таким чином, база даних є сховищем інформації, а СУБД — інструментом для її ефективного використання. Разом вони формують фундамент сучасних інформаційних систем, без яких неможливо уявити роботу жодного великого веб сайту, мобільного додатка чи корпоративного сервісу.

Для туристичної агенції база даних та система управління базами даних (СУБД) можуть стати незамінним інструментом у щоденній роботі, значно спрощуючи облік, пошук, аналіз і обслуговування клієнтів. Уявімо, що в агенції є сотні або навіть тисячі клієнтів, різноманітні тури, готелі, авіарейси, візові послуги, бронювання, платежі тощо. Уся ця інформація потребує організованого зберігання та зручного доступу. Саме для цього і потрібна база даних.

За допомогою СУБД туристична агенція може створити централізовану систему, де будуть зберігатися дані про клієнтів (імена, контакти, історія бронювань), деталі турів (напрямки, дати, ціни, наявність місць), контракти з готелями, постачальниками послуг, рейси, трансфери тощо. Наприклад, менеджер легко може знайти інформацію про те, які тури купував конкретний клієнт, які дати він обрав минулого разу або які напрями були найпопулярнішими за останній рік.

Крім того, СУБД дозволяє налаштовувати доступ до даних: адміністрація має повний контроль, менеджери — доступ до потрібної їм інформації, а клієнти — тільки до своїх даних через онлайн-кабінет. Це забезпечує безпеку інформації та зменшує ризики втрати чи помилок. Також можлива автоматизація певних процесів: наприклад, нагадування про оплату, підтвердження бронювання, формування звітів для бухгалтерії або маркетингу.

Таким чином, впровадження бази даних і СУБД у туристичній агенції не тільки спрощує управління великою кількістю інформації, а й підвищує якість обслуговування клієнтів, зменшує час на рутинні задачі та сприяє розвитку бізнесу завдяки точнішій аналітиці й кращому плануванню.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ

Предметна область курсової роботи - Туристичне бюро.

Задача полягає у тому, щоб:

- Створити базу даних для обслуговування співробітників фірми.
- Реалізувати додавання, редагування та видалення даних про путівки.
- Додати валідацію даних, внесених користувачем.
- Створити шаблони для путівок із вказанням змісту програми.
- Реалізувати пошук та фільтрацію за ціною, країною, містом, тощо.
- Додати використання знижок.
- Зберігати дані клієнтів.

За змістом путівки можуть поділятися на

- пляжні;
- екскурсійні;
- оздоровчі (відпочинок).

Кожен тип має свої особливості організації, необхідний набір документів та обмеження. Пляжні та оздоровчі тури схожі за своїм спрямуванням, оскільки обидва орієнтовані на малорухливий відпочинок, але відрізняються програмою та умовами проживання. Екскурсійні тури створені для пізнання культури й історії, їх організація потребує гідної та ознайомчої програми, основаної на підтверджених фактах та знаннях місцевості. Пригодницькі та екстремальні тури, у свою чергу, передбачають активний відпочинок. Але екстремальні тури мають особливі вимоги до фізичної підготовки учасників.

Тур займає певний час: від години до декількох тижнів, не враховуючи часу на дорогу. Також в деяких випадках потрібно враховувати місце проживання та харчування. Пошук путівок за визначеними параметрами, допомога із збиранням потрібних документів, оформлення покупки, тощо здійснюється співробітниками бюро. Якщо до початку путівки залишається менше 7 днів, вона отримує статус «гарячої». Можливість редагувати дані може бути доступна тільки у

адміністраторів. До розрахунків треба додавати знижки, які спираються на статус путівки, вік клієнтів та кількість їх успішних поїздок.

Інформація, яка має бути в базі даних:

Країна:

- Назва – унікальне, текстове значення, обов'язкове.
- Континент - текстове значення, обов'язкове;

Місто:

- Код – унікальне, текстове значення, обов'язкове.
- Назва – текстове значення, обов'язкове;
- Код\_країни – числове, зовнішній ключ до таблиці "Країна", обов'язкове.

Тип туру:

- Код – унікальне, числове, обов'язкове;
- Назва – текстове значення, обов'язкове;
- Обмеження – текстове значення

Турист:

- Код – унікальне, числове, обов'язкове;
- ПІБ – текстове значення, обов'язкове;
- Номер телефону – числове (10), обов'язкове;
- Електронна пошта – текстове значення;
- Дата народження – дата у форматі ЧЧ.ММ.РРРР;

Співробітник:

- Код – унікальне, числове, обов'язкове;
- ПІБ – текстове значення, обов'язкове;
- Посада – текстове значення (менеджер, адміністратор, гід);
- Номер телефону – числове (10);
- Стать – текстове значення (чоловік, жінка);
- Дата народження – дата у форматі ЧЧ.ММ.РРРР.

Готель:

- Код – унікальне, числове, обов'язкове;
- Назва – текстове значення, обов'язкове;



- Код\_місто – числове, зовнішній ключ до таблиці "Місто", обов'язкове;
- Адреса – текстове значення, обов'язкове;
- Рейтинг – числове значення (від 1 до 5), обов'язкове;
- Контактна інформація – текстове значення, обов'язкове.

#### Тип транспорту:

- Код – унікальне, числове, обов'язкове;
- Тип транспорту – текстове значення (літак, автобус, поїзд), обов'язкове;

#### Транспорт:

- Код – унікальне, числове, обов'язкове;
- Код\_типу\_туру – зовнішній ключ до таблиці "Тип транспорту", обов'язкове
- Компанія – текстове значення, обов'язкове;
- Час відправлення – дата у форматі ЧЧ.ММ.РРРР і час у форматі ГГ:ХХ, обов'язкове;
- Час прибуття – дата у форматі ЧЧ.ММ.РРРР і час у форматі ГГ:ХХ, обов'язкове;

#### Харчування:

- Код – зовнішній ключ до таблиці "Харчування", обов'язкове;
- Тип – текстове значення (тільки сніданок; сніданок і вечеря; сніданок, обід і вечеря; багаторазове харчування), обов'язкове;
- Ціна за харчування – числове значення, необов'язкове.

#### Путівка:

- Код – унікальне, числове, обов'язкове;
- Назва – текстове значення, обов'язкове;
- Ціна – числове значення, обов'язкове;
- Код\_готель – зовнішній ключ до таблиці "Готель", обов'язкове;
- Тривалість – числове значення;
- Код\_тип\_туру – зовнішній ключ до таблиці "Тип туру", обов'язкове;
- Опис – текстове значення, обов'язкове.
- Код\_харчування – зовнішній ключ до таблиці "Харчування", обов'язкове
- Дата початку – дата у форматі ЧЧ.ММ.РРРР, обов'язкове

- Дата кінця – дата у форматі ЧЧ.ММ.РРРР, обов'язкове
- Кількість вільних – числове значення;

Знижка:

- Код – унікальне, числове, обов'язкове;
- Назва – текстове значення (за віком, за кількістю бронювань, за горячістю), обов'язкове;

- Відсоток - числове значення, обов'язкове;

Бронювання:

- Код – унікальне, числове, обов'язкове;
- Код\_клієнт – зовнішній ключ до таблиці "Клієнт", обов'язкове;
- Код\_путівка – зовнішній ключ до таблиці "Путівка", обов'язкове;
- Код\_співробітника – зовнішній ключ до таблиці "Співробітник", обов'язкове;

- Дата бронювання – дата у форматі ЧЧ.ММ.РРРР, обов'язкове;
- Кількість осіб – числове значення, обов'язкове;
- Знижка – числове значення (в %), обов'язкове;

## 2 ПРОЄКТУВАННЯ БАЗИ ДАНИХ

### 2.1 Побудова діаграм

Модель «сутність-зв'язок» – модель даних, яка дозволяє описувати концептуальні схеми за допомогою узагальнених конструкцій блоків. ER-модель – це засіб опису моделей даних. Існує ряд моделей для представлення знань, але одним з найзручніших інструментів уніфікованого представлення даних, незалежного від програмного забезпечення, що його реалізує, є модель «сутність-зв'язок» (рисунок 2.1).

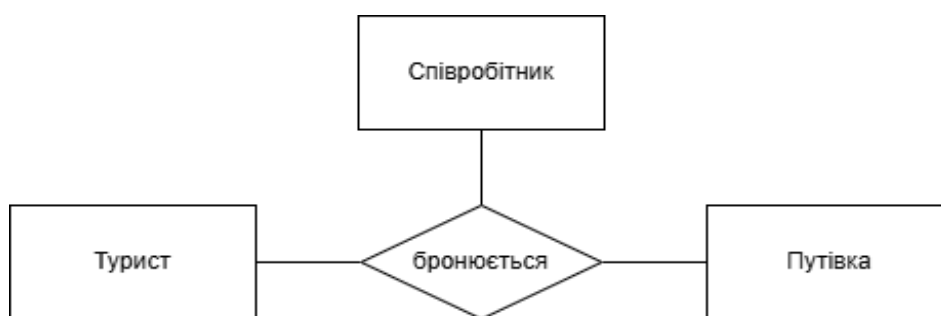


Рис.2.1 – Концептуальна модель

Після детальнішого аналізу, було сформовано більш детальну ер-модель. На ній зображені поля та зв'язки між сутностями.

Діаграми «сутність-зв'язок» (ERD) дозволяють візуалізувати взаємозв'язки між різними об'єктами системи та описувати концептуальні моделі предметної області. Враховуючи специфіку предметної області та розроблену концептуальну модель, створимо ER-діаграму для бази даних (рисунок 2.2).

Діаграми варіантів використання (use case diagrams) застосовуються для візуалізації сценаріїв взаємодії користувачів (actors) із системою та її функціями (use cases). У контексті турагенції акторами виступають різні типи користувачів, а їхні дії із системою відображаються еліпсами, що символізують варіанти використання. Актори позначаються символами людини, які з'єднуються з варіантами

використання напрямленою асоціацією – стрілкою, що спрямована від актора до відповідного сценарію. Крім того, між акторами можуть бути зв'язки узагальнення.

У процесі аналізу предметної області було виділено три типи користувачів системи турагенції – адміністратор, турист та менеджер. Діаграма варіантів використання представлена на рисунку 2.3.

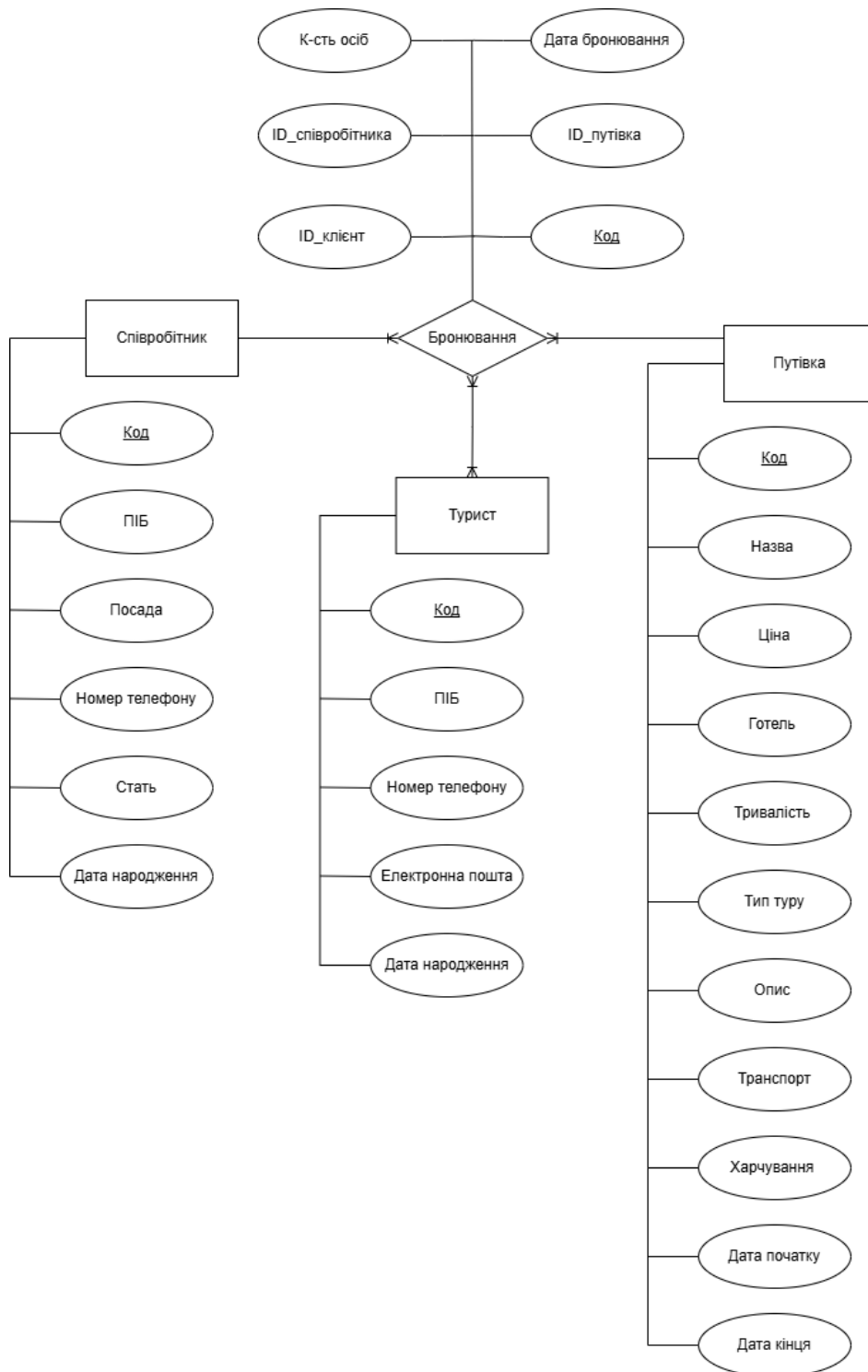


Рис.2.2 – ER-модель

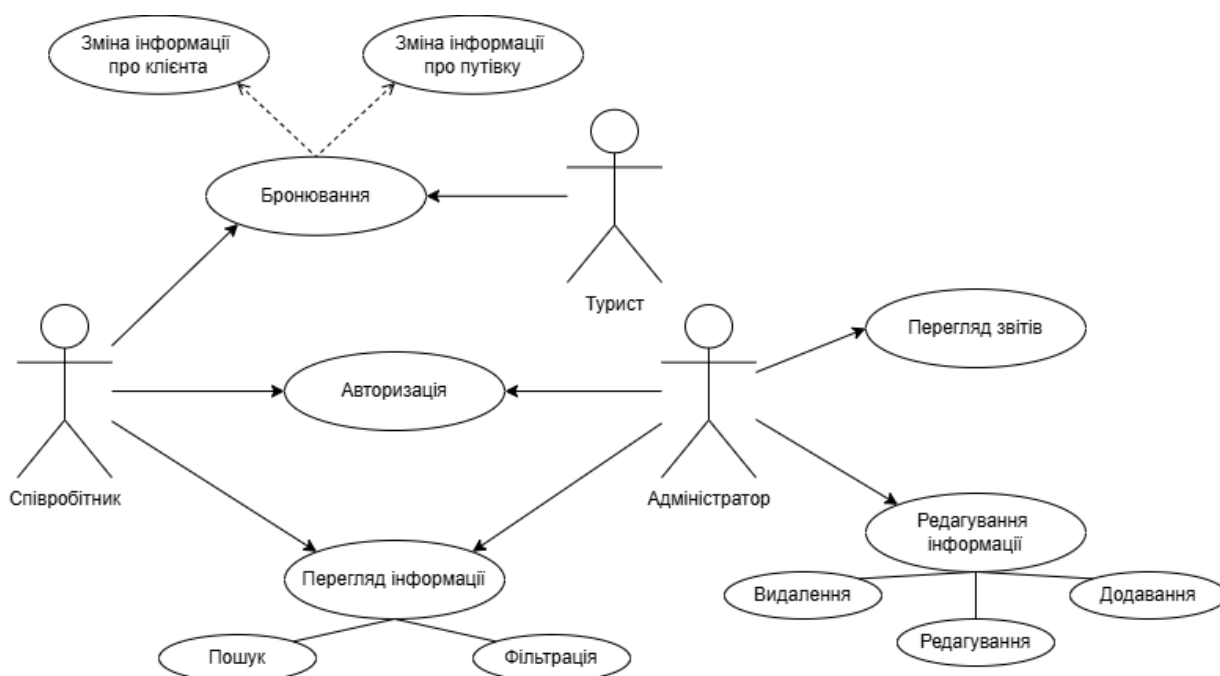


Рис.2.3 – Use-case

## 2.2 Побудова реляційної моделі даних

Основний принцип баз даних – уникнення надлишкових даних. Для цього використовується процес нормалізації.

Нормалізація – це метод усунення повторюваної інформації в базі даних.

– Перша нормальна форма (1NF) означає, що таблиця відповідає базовим вимогам:

- кожен стовпець містить лише неподільні (атомарні) значення;
- стовпці є однорідними за типом даних;
- кожен рядок унікальний;
- таблицю можна редагувати в будь-якому напрямку.

Для досягнення 1НФ дані організовують у прямокутні таблиці (один атрибут – один стовпець), складні атрибути розбивають на прості, а багатозначні виносять у окремі сутності, наприклад, «Категорія». Також усі таблиці містять зовнішні та внутрішні ключі.

Друга нормальна форма (2NF) передбачає, що всі атрибути залежать виключно від усього ключа, а не від його частини. Якщо первинний ключ складається з одного атрибута, така таблиця автоматично відповідає 2НФ.

Третя нормальна форма (3NF) усуває транзитивні залежності між неключовими атрибутами, а також відповідає всім вимогам 2НФ.

Нормалізовані відношення відображені в таблицях 2.1-2.14

Таблиця 2.1 – Відношення «Країна»

| Код | Назва | Континент |
|-----|-------|-----------|
|     |       |           |

Таблиця 2.2 – Відношення «Місто»

| Код | Назва | Код_країни |
|-----|-------|------------|
|     |       |            |

Таблиця 2.3 – Відношення «Тип туру»

| Код | Назва | Обмеження |
|-----|-------|-----------|
|     |       |           |

Таблиця 2.4 – Відношення «Турист»

| Код | ПІБ | Номер телефону | Ел. пошта | Дата народження |
|-----|-----|----------------|-----------|-----------------|
|     |     |                |           |                 |

Таблиця 2.5 – Відношення «Посада»

| Код | Назва |
|-----|-------|
|     |       |

Таблиця 2.6 – Відношення «Співробітник»

| Код | ПІБ | Код_посади | Номер телефону |
|-----|-----|------------|----------------|
|     |     |            |                |

Таблиця 2.7 – Відношення «Готель»

| Код | Назва | Код міста | Адреса | Рейтинг | Контактна інформація |
|-----|-------|-----------|--------|---------|----------------------|
|     |       |           |        |         |                      |

Таблиця 2.8 – Відношення «Тип транспорту»

| Код | Тип |
|-----|-----|
|     |     |

Таблиця 2.9 – Відношення «Транспорт»

| Код | Код типу туру | Час відправлення | Час прибуття |
|-----|---------------|------------------|--------------|
|     |               |                  |              |

Таблиця 2.10 – Відношення «Харчування»

| Код | Тип | Ціна за харчування |
|-----|-----|--------------------|
|     |     |                    |

Таблиця 2.11 – Відношення «Путівка\_1»

| Код | Назва | Код готель | Тривалість | Код тип туру | Опис | Код харчування | Кількість місць |
|-----|-------|------------|------------|--------------|------|----------------|-----------------|
|     |       |            |            |              |      |                |                 |

Таблиця 2.12 – Відношення «Путівка\_2»

| Код | Код_путівки_1 | Дата початку | Дата кінця | Ціна |
|-----|---------------|--------------|------------|------|
|     |               |              |            |      |

Таблиця 2.13 – Відношення «Знижка»

| Код | Назва | Відсоток |
|-----|-------|----------|
|     |       |          |

Таблиця 2.14 – Відношення «Бронювання»

| Код | Код туриста | Код путівки | Код співробітника | Дата бронювання | Кількість осіб | Знижка |
|-----|-------------|-------------|-------------------|-----------------|----------------|--------|
|     |             |             |                   |                 |                |        |

### 2.3 Вибір СУБД та опис фізичної моделі бази даних

Створення фізичної моделі бази даних — це один із завершальних етапів розробки, на якому логічна модель перетворюється у фізичну структуру, що безпосередньо залежить від обраної системи керування базами даних (СКБД). На цьому етапі всі відношення, визначені в логічній моделі реляційної бази даних, трансформуються у відповідні SQL-команди для створення фізичних об'єктів, таких як таблиці, індекси, зв'язки між таблицями та обмеження цілісності. Внаслідок цього формується так звана внутрішня схема бази даних, яка визначає, як дані будуть організовані та збережені.

Окрім внутрішньої схеми, може бути створена й зовнішня схема, яка відображає представлення даних з точки зору користувачів. Вона дозволяє налаштовувати різні рівні доступу до інформації, визначати представлення (віртуальні таблиці) та адаптувати базу даних до конкретних потреб кінцевих користувачів або прикладних програм.

Головна мета цього етапу – коректне перетворення логічної моделі реляційної бази даних у набір команд для створення її фізичних об'єктів у середовищі конкретної СКБД, зокрема, у SQLite. Це забезпечує ефективне збереження, пошук та маніпуляцію даними в базі.

Провівши аналіз статей на тему СУБД було обрано СУБД MySQL Workbench.

MySQL Workbench є потужним і зручним інструментом для роботи з MySQL, особливо корисним для розробників і адміністраторів баз даних. Він значно спрощує моделювання, написання SQL-запитів та управління серверами, проте має деякі недоліки, пов'язані з продуктивністю та стабільністю при роботі з великими обсягами даних.



Переваги:

- Зручний графічний інтерфейс.
- Вбудовані інструменти для моделювання.
- Інструменти для управління сервером.
- Надійність.

Недоліки:

- Високі вимоги до ресурсів.
- Не підтримує всі можливості інших СКБД.
- Обмежений функціонал у безкоштовній версії.

Фізична модель даних відображена в таблиці 2.15.

Таблиця 2.15 – Фізична модель бази даних

| Назва стовпця    | Тип даних   | Обмеження                | Підпис                              |
|------------------|-------------|--------------------------|-------------------------------------|
| Таблиця Країна   |             |                          |                                     |
| Id               | TINY INT    | PRIMARY KEY              | Код країни                          |
| Name             | VARCHAR(50) | NOT NULL,<br>UNIQUE      | Повна назва країни                  |
| Continent        | VARCHAR(50) | NOT NULL                 | Континент, до якого належить країна |
| Таблиця Місто    |             |                          |                                     |
| Id               | TINY INT    | PRIMARY KEY              | Код міста                           |
| Name             | VARCHAR(50) | NOT NULL                 | Назва міста                         |
| Id_Country       | CHAR(3)     | NOT NULL,<br>FOREIGN KEY | Код країни, до якої належить місто  |
| Таблиця Тип туру |             |                          |                                     |
| Id               | TINY INT    | PRIMARY KEY              | Код типу туру                       |
| Type             | VARCHAR(50) | NOT NULL,<br>UNIQUE      | Назва типу                          |
| Restriction      | TEXT        | NOT NULL                 | Обмеження для туру                  |
| Таблиця Турист   |             |                          |                                     |

Продовження таблиці 2.15

| Назва стовпця          | Тип даних    | Обмеження                             | Підпис                            |
|------------------------|--------------|---------------------------------------|-----------------------------------|
| Id                     | TINY INT     | PRIMARY KEY                           | Код туриста                       |
| Full_Name              | VARCHAR(100) | NOT NULL                              | Повне ім'я туриста (ПІБ)          |
| Phone_Number           | VARCHAR(15)  | NOT NULL                              | Номер телефону                    |
| Email                  | VARCHAR(50)  | NOT NULL,<br>UNIQUE                   | Електронна пошта                  |
| Date_Of_Birth          | DATE         | NOT NULL                              | Дата народження                   |
| Таблиця Посада         |              |                                       |                                   |
| Id                     | TINY INT     | PRIMARY KEY                           | Код посади                        |
| Name                   | VARCHAR(50)  | NOT NULL                              | Назва посади                      |
| Таблиця Співробітник   |              |                                       |                                   |
| Id                     | TINY INT     | PRIMARY KEY                           | Код співробітника                 |
| Full_Name              | VARCHAR(200) | NOT NULL                              | ПІБ                               |
| Position_id            | INT          | NOT NULL,<br>FOREIGN KEY              | Код посади                        |
| Phone_Number           | VARCHAR(15)  | NOT NULL                              | Номер телефону                    |
| Таблиця Готель         |              |                                       |                                   |
| Id                     | TINY INT     | PRIMARY KEY                           | Код готелю                        |
| Name                   | VARCHAR(100) | NOT NULL                              | Назва готелю                      |
| City_id                | INT          | NOT NULL,<br>FOREIGN KEY              | Код міста, де розташований готель |
| Address                | TEXT         | NOT NULL                              | Адреса готелю                     |
| Rating                 | DECIMAL(3,2) | CHECK(Rating >= 0<br>AND Rating <= 5) | Рейтинг готелю (від 0 до 5)       |
| Contact_Info           | VARCHAR(255) | NOT NULL                              | Контактна інформація готелю       |
| Таблиця Тип транспорту |              |                                       |                                   |

Продовження таблиці 2.15

| Назва стовпця      | Тип даних     | Обмеження                | Підпис                                  |
|--------------------|---------------|--------------------------|---|
| Id                 | TINY INT      | PRIMARY KEY              | Код типу транспорту                     |
| Type               | VARCHAR(20)   | NOT NULL                 | Тип транспорту                          |
| Таблиця Транспорт  |               |                          |   |
| Id                 | TINY INT      | PRIMARY KEY              | Код транспорту                          |
| Tour_type_id       | INT           | NOT NULL,<br>FOREIGN KEY | Код типу туру                           |
| Departure_Time     | DATETIME      | NOT NULL                 | Час відправлення транспорту             |
| Arrival_Time       | DATETIME      | NOT NULL                 | Час прибуття транспорту                 |
| Таблиця Харчування |               |                          |   |
| Id                 | TINY INT      | PRIMARY KEY              | Код харчування                          |
| Type               | VARCHAR(100)  | NOT NULL                 | Тип харчування                          |
| Price_Per_Meal     | DECIMAL(10,2) | NOT NULL                 | Ціна за харчування                      |
| Таблиця Путівка_1  |               |                          |   |
| Id                 | TINY INT      | PRIMARY KEY              | Унікальний код путівки                  |
| Name               | VARCHAR(100)  | NOT NULL                 | Назва путівки                           |
| Hotel_id           | INT           | NOT NULL,<br>FOREIGN KEY | ID готелю, до якого відноситься путівка |
| Duration           | INT           | NOT NULL                 | Тривалість путівки в днях               |
| Tour_type_iD       | INT           | NOT NULL,<br>FOREIGN KEY | ID типу туру                            |
| Description        | TEXT          | NOT NULL                 | Опис путівки                            |
| Food_iD            | INT           | NOT NULL,<br>FOREIGN KEY | ID харчування, яке включено в путівку   |
| Seats              | INT           | NOT NULL                 | Кількість місць                         |
| Таблиця Путівка_2  |               |                          |   |

Продовження таблиці 2.15

| Назва стовпця      | Тип даних     | Обмеження                               | Підпис                                       |
|--------------------|---------------|---|--|
| Id                 | TINY INT      | PRIMARY KEY                             | Код путівки                                  |
| Trip_main_id       | INT           | NOT NULL,<br>FOREIGN KEY                | Код 1 частини путівки                        |
| Start_date         | DATE          | NOT NULL                                | Дата початку путівки                         |
| End_date           | DATE          | NOT NULL                                | Дата кінця путівки                           |
| Price              | DECIMAL(10,2) | NOT NULL                                | Ціна путівки                                 |
| Таблиця Знижка     |               |   |  |
| Code               | TINY INT      | PRIMARY KEY                             | Код знижки                                   |
| Name               | VARCHAR(100)  | NOT NULL                                | Назва знижки                                 |
| Percentage         | DECIMAL(5,2)  | CHECK(Rating >= 0<br>AND Rating <= 100) | Відсоток знижки                              |
| Таблиця Бронювання |               |   |  |
| Code               | TINY INT      | PRIMARY KEY                             | Код знижки                                   |
| Tourist_Code       | INT           | NOT NULL,<br>FOREIGN KEY                | Код туриста, який здійснив бронювання        |
| Trip_Code          | INT           | NOT NULL,<br>FOREIGN KEY                | Код путівки                                  |
| Employee_Code      | INT           | NOT NULL,<br>FOREIGN KEY                | Код співробітника, який обробляв             |
| Booking_Date       | DATETIME      | NOT NULL                                | Дата бронювання                              |
| People_count       | INT           | NOT NULL                                | Кількість осіб, що можуть поїхати по путівці |
| Discount_Code      | INT           | FOREIGN KEY                             | Код знижки, якщо застосовується              |

## 2.4 Ескіз інтерфейсу користувача

Для більш зручної взаємодії з базою даних, необхідно розробити додаток, який буде надавати зручні інструменти для додавання, змінювання та перегляду інформації. Перед розробкою додатка, потрібно спроектувати ескізи інтерфейсу, для більш чіткого уявлення фінального додатку. Ескіз форми для авторизації користувача відображений на рисунку 2.4

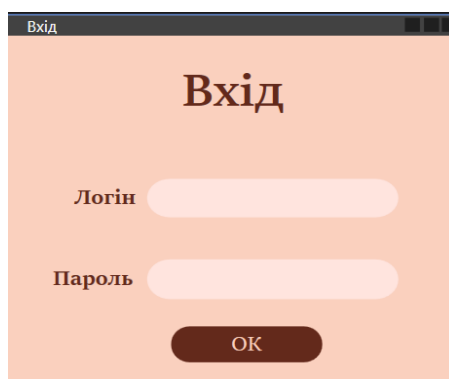


Рисунок 2.4 – Ескіз вікна авторизації

На рисунку 2.5 зображено ескіз головного вікна, де відкривається сторінка всіх доступних путівок. Зліва є фільтри для них а вгорі пошук за іменем. Фільтрація буде мати в собі більше варіантів після додавання бази даних. Кнопка “Перегляд” показує повну інформацію про путівки, таку як дата початку та кінця, ціна, тощо (Рисунок 2.6)

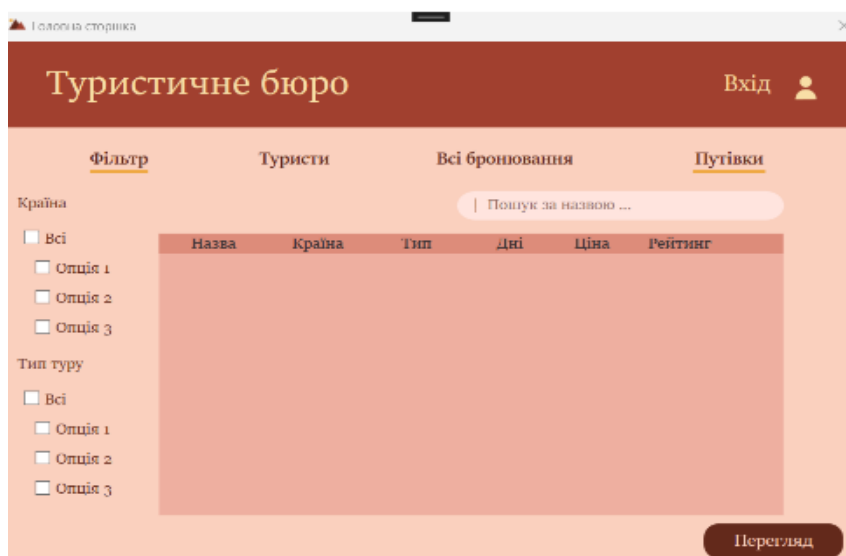


Рисунок 2.5 – Ескіз головного вікна

Вікно працюватиме лише при обиранні певного запису. Кнопка “Бронювати” відкриває відкриває вікно, зображене на рисунку 2.7. Деякі з тих полів вже будуть заповнені, беручи інформацію з попереднього вікна.

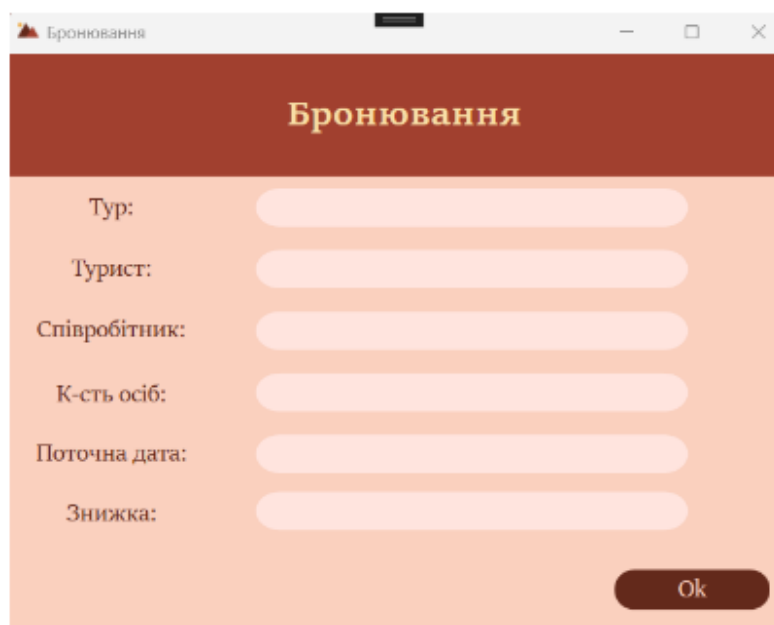


Вікно "Інформація про путівку" має заголовок "Інформація про путівку" на червоному фоні. Нижче розташовані два стовпці полів для введення даних:

|             |               |
|-------------|---------------|
| Назва:      | Транспорт:    |
| Місто:      | Харчування:   |
| Країна:     | Готель:       |
| Тип:        | Місця:        |
| Тривалість: | Дата початку: |
| Рейтинг:    | Дата кінця:   |
| Опис:       | Ціна:         |

У правому нижньому куті розташована кнопка "Бронювати".

Рисунок 2.6 – Вікно “Інформація про путівку”



Вікно "Бронювання" має заголовок "Бронювання" на червоному фоні. Нижче розташовані форми для введення даних:

|               |                      |
|---------------|----------------------|
| Тур:          | <input type="text"/> |
| Турист:       | <input type="text"/> |
| Співробітник: | <input type="text"/> |
| К-сть осіб:   | <input type="text"/> |
| Поточна дата: | <input type="text"/> |
| Знижка:       | <input type="text"/> |

У правому нижньому куті розташована кнопка "Ok".

Рисунок 2.7 – Ескіз бронювання путівки

На головному вікні також є кнопка для переходу на сторінку “Всі бронювання” (Рисунок 2.8). Кнопка “Редагувати” веде до того ж вікна, що показано на рисунку

2.7. Але поля вже заповнені, які можна відредагувати. Кнопка “Видалити” має видаляти обране бронювання.

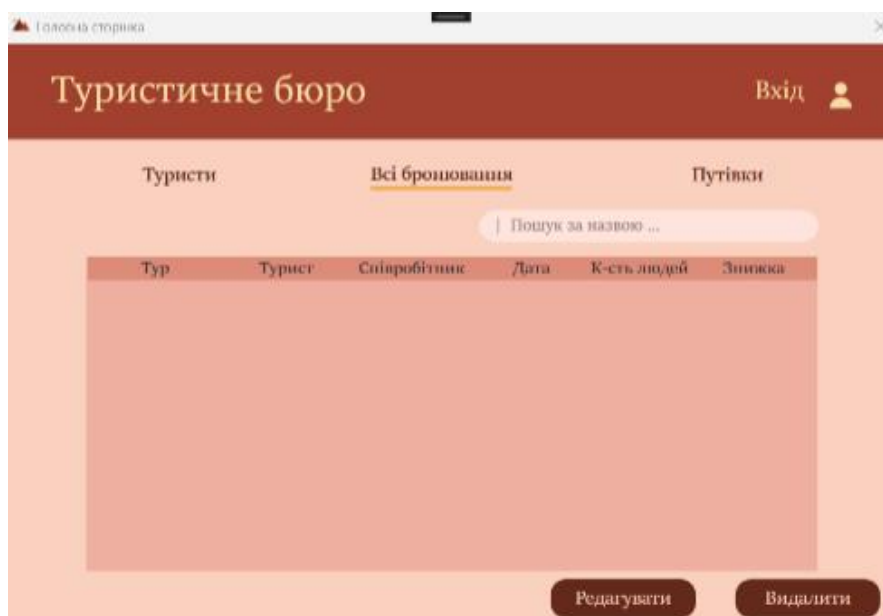


Рисунок 2.8 – Ескіз сторінки “Всіх бронювань”

На головному екрані, при натисканні на кнопку “Туристи” відкривається нова сторінка, де буде інформація про туристів, що вже є в системі (Рисунок 2.9).

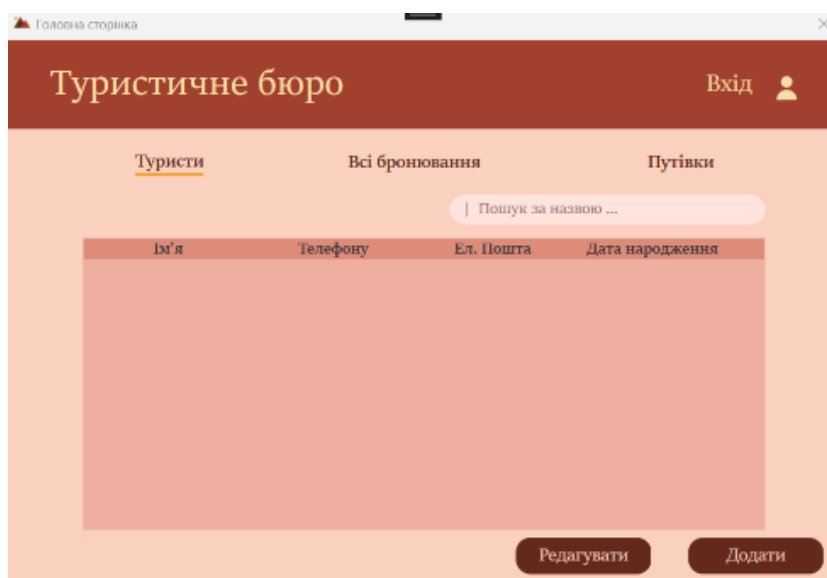
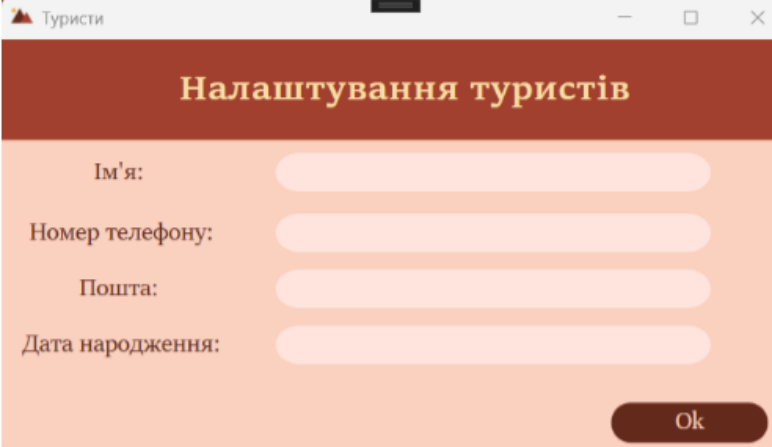


Рисунок 2.9 – Головне вікно, сторінка “Туристи”

Кнопки “Редагувати” та “Додати” ведуть до одного вікна - “Налаштування туристів”. Але при редагуванні поля заповнені вже існуючими даними, і це зображено на Рисунок 2.10.



Туристи

### Налаштування туристів

Ім'я:

Номер телефону:

Пошта:

Дата народження:

Ok

Рисунок 2.10 – Вікно “Налаштування туристів”



### 3 ОПИС ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

#### 3.1 Опис структури бази даних

База даних «wch» створена в СКБД MySQL. БД використовується для зберігання даних про продукти та товари, покупців.

Створення бази даних: CREATE DATABASE IF NOT EXISTS `travel\_agency`  
DEFAULT CHARACTER SET utf8 COLLATE utf8;

Найважливіша таблиця - бронювання: в ній зібрана основна інформація. Таблиця зв'язана з 5 іншими таблицями за вторинними ключами. Скрипт написаний нижче:

Таблиці країни та міста, які знаходяться в цих країнах, зберігають значення місць, де будуть проходити тури. В країні також зберігають континенти, на якому знаходяться ці країни:

```
CREATE TABLE country (  
    id TINYINT PRIMARY KEY,  
    country_name VARCHAR(50) NOT NULL UNIQUE,  
    continent VARCHAR(50) NOT NULL );  
  
CREATE TABLE city(  
    id TINYINT AUTO_INCREMENT PRIMARY KEY,  
    city_name VARCHAR(50) NOT NULL,  
    id_country TINYINT NOT NULL, FOREIGN KEY (id_country) REFERENCES  
country(id) );
```

В цій таблиці зберігаються типи турів

```
CREATE TABLE tour_type (  
    id TINYINT AUTO_INCREMENT PRIMARY KEY,  
    `type` VARCHAR(50) NOT NULL,  
    restriction text NOT NULL );
```

В таблиці туристів зберігається вся важлива інформація про клієнтів, що звертаються до турбюро

```
CREATE TABLE tourist (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    full_name VARCHAR(100) NOT NULL,
    phone_number VARCHAR(15) NOT NULL,
    email VARCHAR(50) NOT NULL UNIQUE,
    date_of_birth DATE NOT NULL );
```

В таблиці посад зберігаються всі можливі посади, на які можуть взяти людей в це бюро

```
CREATE TABLE job_position (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    name_position VARCHAR(50) NOT NULL );
```

Таблиця співробітників прив'язана до попередньої. В цій зберігається особиста інформація вже існуючих працівників агентства

```
CREATE TABLE employee (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    full_name VARCHAR(200) NOT NULL,
    id_position TINYINT NOT NULL, FOREIGN KEY (id_position) REFERENCES
job_position(id),
    phone_number VARCHAR(15) NOT NULL );
```

Таблиця готелів містить коротку інформацію про готелі, з яким співпрацює бюро. Для більшої інформації треба зв'язуватися за вказаними контактами

```
CREATE TABLE hotel (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    name_hotel VARCHAR(100) NOT NULL,
    id_city TINYINT NOT NULL, FOREIGN KEY (id_city) REFERENCES city(id),
    address TEXT NOT NULL,
    rating DECIMAL(3,2) CHECK(Rating >= 0 AND Rating <= 5),
    contact_info VARCHAR(255) NOT NULL );
```

Таблиця тип транспорту зберігає тільки види транспорту, а в наступній міститься більш розгорнута інформація про доступні транспорти

```
CREATE TABLE type_transport (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    type_transport VARCHAR(20) NOT NULL );
```

```
CREATE TABLE transport (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    id_type_transport TINYINT NOT NULL, FOREIGN KEY
(id_type_transport) REFERENCES type_transport(id),
    departure_time DATETIME NOT NULL,
    arrival_time DATETIME NOT NULL );
```

Таблиця Їжа призначена для списку варіантів харчування під час подорожі

```
CREATE TABLE food (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    type_food VARCHAR(100) NOT NULL,
    price_per_meal DECIMAL(10,2) NOT NULL );
```

Перша частина путівки зберігає основну інформацію, на яку звертають увагу при пошуку потрібної. Також до неї прив'язано багато інших таблиць

```
CREATE TABLE voucher_1 (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    name_voucher VARCHAR(100) NOT NULL,
    id_hotel TINYINT NOT NULL, FOREIGN KEY (id_hotel) REFERENCES
hotel(id),
    duration INT NOT NULL,
    id_tour_type TINYINT NOT NULL, FOREIGN KEY (id_tour_type)
REFERENCES tour_type(id),
    id_transport TINYINT NOT NULL, FOREIGN KEY (id_transport) REFERENCES
transport(id),
    `description` TEXT NOT NULL,
    id_food TINYINT NOT NULL, FOREIGN KEY (id_food) REFERENCES
food(id),
    seats INT NOT NULL );
```

Друга частина призначена для “рухливої” частини, яка може змінюватися при різних обставинах. Наприклад, взимку деякі мандрівки стають дешевші

```
CREATE TABLE voucher_2 (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    id_trip TINYINT NOT NULL, FOREIGN KEY (id_trip) REFERENCES
voucher_1(id),
    start_date DATE NOT NULL,
    end_date DATE NOT NULL,
    Price DECIMAL(10,2) NOT NULL );
```

Окрема таблиця знижок має зарезервовані випадки, при яких дають знижки

```
CREATE TABLE discount (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    discount_name VARCHAR(100) NOT NULL,
    percentage DECIMAL(5,2) NOT NULL CHECK (percentage >= 0 AND
percentage <= 100) );
```

Також є таблиця бронювань, яка пов’язана з багатьма іншими.

```
CREATE TABLE booking (
    id TINYINT AUTO_INCREMENT PRIMARY KEY,
    id_tourist TINYINT NOT NULL, FOREIGN KEY (id_tourist) REFERENCES
tourist(id),
    id_voucher TINYINT NOT NULL, FOREIGN KEY (id_voucher) REFERENCES
voucher_2(id),
    id_employee TINYINT NOT NULL, FOREIGN KEY (id_employee)
REFERENCES employee(id),
    booking_date DATETIME NOT NULL,
    people_count INT NOT NULL,
    id_discount TINYINT, FOREIGN KEY (id_discount) REFERENCES discount(id)
);
```

Створені таблиці та їх зв’язки відображено на рисунку 3.1.

### 3.2 Опис задач автоматизації та інтерфейсу користувача

### 3.2 Опис задач автоматизації та інтерфейсу користувача

При запуску застосунку, першим з’являється Головне вікно на сторінці “Путівки” (Рисунок 3.2). В списку доступних путівок є майже однакові за виглядом, але у них різні ціна і дата проведення. Також доступні різні фільтри: за країною, за типом путівки, за проміжком, в який має проходити путівка та межі, в які потрапляє ціна путівки (Рисунок 3.3). Також над таблицею доступних путівок є поле пошуку за назвою путівки. Поле можна очистити за допомогою “хрестика”.



Рисунок 3.2 - Сторінка “Путівки” та початок фільтрів

**Фільтри**

☐ Оздоровчі

Ціна (грн)

Від:

До:

Дата

З:

По:

OK

Рисунок 3.3 – Кінець фільтрів

Також в правому верхньому куті є кнопка “Вхід”, яка відкриває вікно “Авторизація” (Рисунок 3.4) Без авторизації не можна бронювати путівки, додавати

та редагувати туристів, а також редагувати вже існуючі бронювання. У кожного працівника є свій логін та пароль. При входженні в систему буде відображатися повне ім'я працівника.



Вхід

Логін: murshef

Пароль: [masked]

ОК

Рисунок 3.4 – Авторизація

Після успішної авторизації знову відкривається головне вікно, де на сторінці “Путівки” є кнопка “Перегляд”. Вона виводить повну інформацію про обрану путівку (Рисунок 3.5).



Путівка

### Інформація про путівку

|              |                 |            |                        |
|--------------|-----------------|------------|------------------------|
| Назва        | Римська класика | Тип туру   | Оздоровчі              |
| Місто:       | Мілан           | Готель:    | Готель "Морський бриз" |
| Країна:      | Італія          | Транспорт: | Літак                  |
| Тривалість:  | 14              | Місце:     | 30                     |
| Дата початку | 11/5/2025       | Ціна:      | 9000.00                |
| Дата кінця:  | 11/12/2025      | Рейтинг:   | 1                      |

**Опис:** Відчуйте дух античного Риму, прогулявшись по Колізею, Пантеону та Ватикану. Протягом цього туру ви відвідаєте найвідоміші пам'ятки і насолодитесь середземноморською кухнею. Подорож літаком зробить вашу поїздку швидкою та комфортною.

Бронювати

Рисунок 3.5 – Інформація про путівку

Кнопка “Бронювати” веде до вікна “Бронювання” (Рисунок 3.6). Вибір співробітника недоступний і за замовчуванням обраний той, який авторизований в системі. З туристів можна вибрати тільки існуючого. Автоматично встановлено поточну дату. Знижка нараховується в залежності від вибраного:

- Дитяча, якщо вибраний пункт “Є діти” (недоступно, якщо їде 1 людина);
- Пенсійна, якщо вибраний пункт “Є люди похилого віку”;
- Оптова, якщо від 5 людей в путівці;
- Рання, якщо між датою бронювання та датою початку путівки мінімум 2 місяці.

Доступна тільки одна знижка, і якщо співпадає декілька пунктів, нараховується найбільша з них. Після розрахунку знижки можна змінити налаштування.

Рисунок 3.6 – Бронювання путівки

Після натискання на кнопку “ОК” вже буде розрахована фінальна сума путівки і її буде виведено у спливаючому повідомленні (Рисунок 3.7).



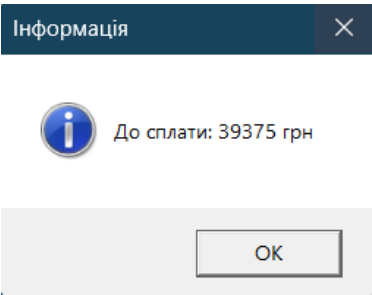


Рисунок 3.7 – Оплата

На Головному вікні також є сторінка “Всі бронювання” (Рисунок 3.8), де виводиться список всіх бронювань. Справа доступна фільтрація по полях: проміжок терміну, у який може проходити путівка, посада працівника, що оформлював та наявність або відсутність знижки. Так само, як і на “Путівках” можна зробити пошук за назвою бронювання. Під таблицею є загальна сума з усіх бронювань.

При фільтрації вона змінюється. Кнопка “Редагувати” веде на те саме вікно, що зображено на Рисунку 3.6. Але цього разу інформація вже заповнена даними путівки, що редагується. Так само можна спочатку розрахувати знижку, а потім вже забронювати.

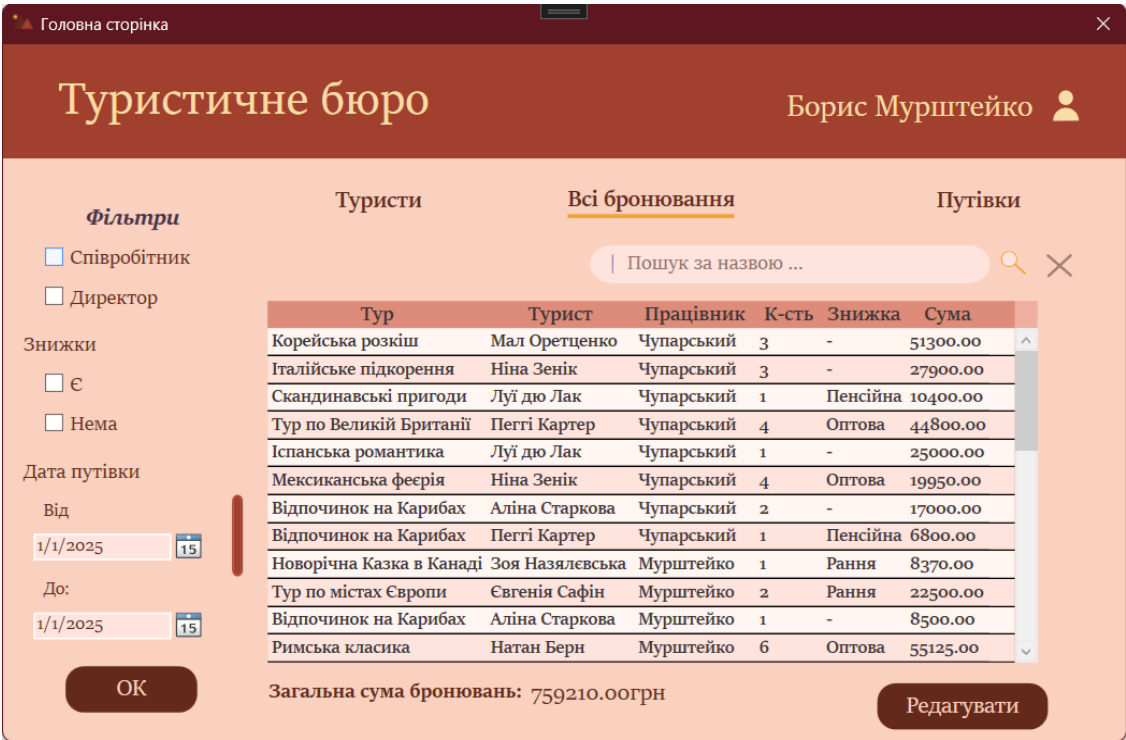


Рисунок 3.8 – Сторінка всіх бронювань

На сторінці “Туристи” (Рисунок 3.9) список вже існуючих туристів. Є фільтри за постійністю клієнтів (більше 1 бронювання – вже постійний клієнт) та за номером телефону. І є пошук за іменем туриста.

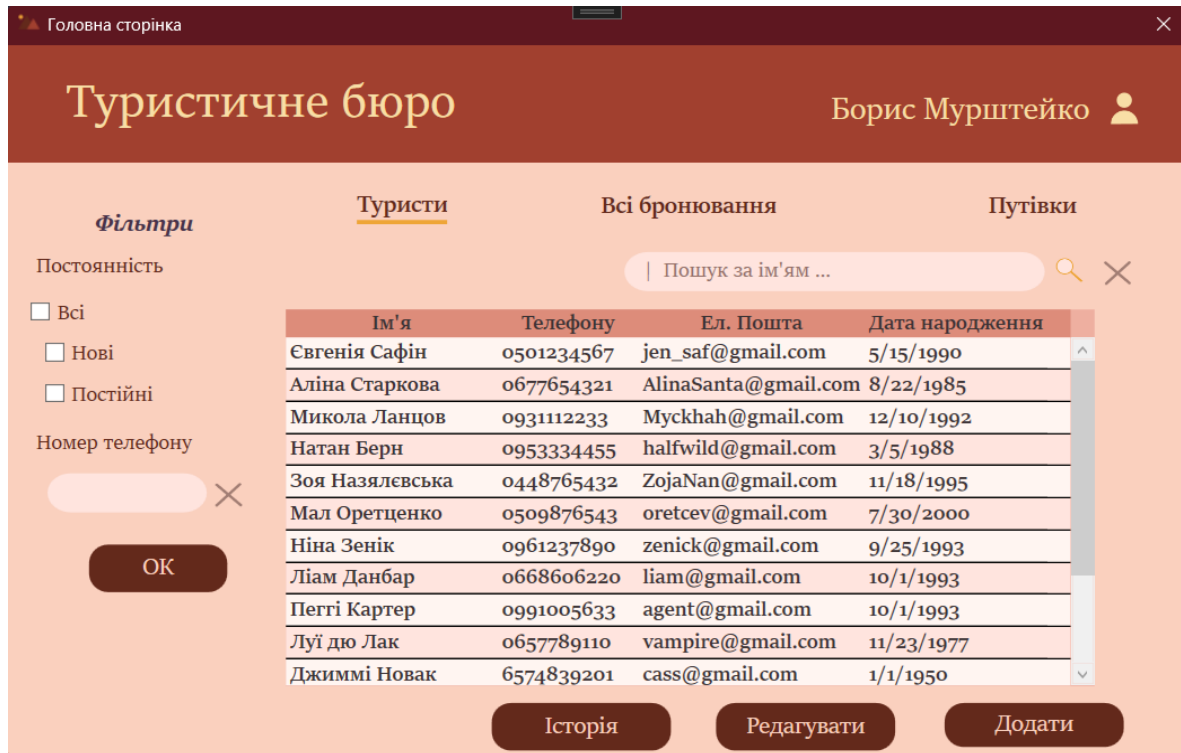


Рисунок 3.9 – Сторінка туристів

Можна додавати нових, або редагувати існуючих – відкривається одне й те саме вікно, але воно відрізняється тим, що при додаванні (Рисунок 3.10) поля пусті (окрім дати), а при редагуванні (Рисунок 3.11) вже заповнені інформацією.

Рисунок 3.10 – Додавання

Рисунок 3.11 – Редагування

Окрім того, доступна кнопка перегляду історії бронювань обраного користувача (Рисунок 3.12)



Рисунок 3.11– Історія бронювань туриста

## ВИСНОВКИ

У підсумку, використання бази даних та системи управління базами даних у туристичній агенції є не просто корисним, а фактично необхідним інструментом для ефективного функціонування бізнесу. Така система дозволяє надійно зберігати, впорядковувати та оперативно обробляти великі обсяги інформації, які щоденно проходять через агенцію. Завдяки цьому значно спрощуються процеси бронювання, обліку клієнтів, управління турами, фінансовими операціями та аналітикою.

Особливу роль у практичному використанні СУБД відіграє інтерфейс взаємодії з користувачем. Інтерфейс – це те, як співробітники агенції або самі клієнти взаємодіють із базою даних: через програми, веб-сайти, мобільні додатки чи CRM-системи. Добре продуманий, інтуїтивно зрозумілий інтерфейс значно підвищує ефективність роботи персоналу, зменшує кількість помилок, пришвидшує обслуговування клієнтів та робить взаємодію з сервісом приємною. Наприклад, менеджер може в кілька кліків знайти тур, сформулювати пропозицію та відправити її клієнтові, не звертаючись до безлічі файлів чи записів.

У цифрову епоху, коли швидкість обробки інформації та зручність сервісу є критичними факторами конкурентоспроможності, саме комбінація надійної бази даних, потужної СУБД та якісного інтерфейсу дає туристичним агенціям змогу розвиватися, автоматизувати процеси, забезпечувати високий рівень обслуговування та адаптуватися до потреб сучасного ринку.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Інформація про бази даних вцілому та СУБД зокрема [Електронний ресурс].  
- Режим доступу: <https://uk.wikipedia.org/wiki>
2. Документація про підключення бази даних до WPF [Електронний ресурс]. -  
Режим доступу: <https://metanit.com/sharp/wpf/19.2.php>
3. Вже існуючий сайт туристичного агентства для аналізу предметної області  
[Електронний ресурс]. - Режим доступу: <https://anex-tour.com.ua/uk/>