

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ ЛЬВІВСЬКИЙ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА  
ФАКУЛЬТЕТ УПРАВЛІННЯ ФІНАНСАМИ ТА БІЗНЕСУ**

**Кафедра цифрової економіки та бізнес-аналітики**  
(назва кафедри)

**КУРСОВА РОБОТА**

з Проектування та адміністрування БД і СД  
(назва дисципліни/дисциплін)

на тему:

Інформаційна система для роботи з клієнтами туристичного  
оператора

**спеціальність:** 0305 Економіка та підприємництво  
(код та найменування спеціальності)

**спеціалізація:** 6.030502 Економічна кібернетика  
(найменування спеціалізації)

**освітній ступінь:** Бакалавр  
(бакалавр/магістр)

**Науковий керівник:**  
Ярема О.Р.к.є.н,доц.  
(науковий ступінь, посада, прізвище, ініціали)  
“ ” 2020 р.  
(підпис)

**Виконавець:**  
студент(ка) групи УФЕ-31с  
Іванів Т.М.  
(прізвище, ініціали)  
“ ” 2020 р.  
(підпис)

**Загальна кількість балів** \_\_\_\_\_

\_\_\_\_\_  
(підпис, ПП членів комісії)

**ЛЬВІВ 2020**

## ЗМІСТ

<b>ВСТУП.....</b>	<b>3</b>
<b>РОЗДІЛ 1. АНАЛІЗ ВИМОГ.....</b>	<b>5</b>
1. Постановка завдання.....	5
<b>РОЗДІЛ 2. РОЗРОБКА БАЗИ ДАНИХ .....</b>	<b>7</b>
2.1. Опис моделі даних .....	7
2.1.1. Перелік таблиць бази даних.....	8
2.1.2. Перелік полів таблиць бази даних.....	9
2.2. Нормалізація реляційних відношень.....	10
2.3. Визначення типів даних.....	12
2.4. Запити до таблиць бази даних.....	14
<b>РОЗДІЛ 3. РОЗРОБКА ВЕБ-САЙТУ.....</b>	<b>18</b>
3.1. Інструменти для створення веб-сайту.....	18
3.2. Структура веб-сайту.....	19
3.3. Програмування серверної частини.....	20
3.4. Програмування клієнтської частини.....	21
3.5. Розміщення веб-сайту на локальному сервері.....	21
<b>ВИСНОВКИ.....</b>	<b>22</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>24</b>
<b>ДОДАТКИ.....</b>	<b>25</b>

## ВСТУП

Сучасною об'єктивною реальністю є широке впровадження у всі сфери життєдіяльності особи, суспільства та держави сучасних інформаційних технологій, розгортання на їх основі різноманітної складності локальних і глобальних інформаційних систем, призначених для прискорення обміну інформацією та доступу до різноманітних інформаційних джерел.

Основною ідеєю сучасних інформаційних технологій є концепція про те, що всі дані повинні бути організовані в бази даних. Відповідно до даної концепції основою інформаційної технології є дані, організовані в БД, які адекватно відображають реалії дійсності в тій чи іншій предметній області і забезпечують користувача актуальною інформацією. Створення і функціонування таких баз даних управляються спеціальними програмними комплексами - системами управління базами даних (СУБД). Ядром будь-якої бази даних є модель даних. Модель даних являє собою безліч структур даних, обмежень цілісності і маніпулювання даними. За допомогою моделі даних можуть бути представлені об'єкти предметної області та взаємозв'язки між ними.

Метою даної роботи є проектування бази даних та реалізація Web-орієнтованої системи по роботі з клієнтами туристичного оператора. У даному проекті наведені короткі теоретичні відомості про засоби, використані для реалізації практичної задачі, конкретна постановка задачі, опис її функціональних можливостей. Початковим етапом практичного завдання є проектування бази даних на основі системи управління базами даних MySQL, складання запитів взаємозв'язків між різними таблицями.

Наступним етапом є побудова інтерфейсу web-сторінки і забезпечення взаємодії її зі створеною базою даних на основі PHP.

Об'єктом проведеного дослідження даної роботи є практичне використання наступних засобів розробки: PHP, MySQL, CSS та HTML.

Предметом організація даних з використанням реляційної моделі даних, методи та особливості взаємодії засобів проектування баз даних та Web-технологій.

## РОЗДІЛ 1. АНАЛІЗ ВИМОГ

### 1. Постановка завдання

Практичним завданням курсової роботи є розробка Web-орієнтованої системи по роботі з клієнтом туристичного оператора.

Метою роботи є створення програмного продукту, який дозволить клієнтам здійснювати бронювання туру, переглядати наявні тури, добавляти основну інформацію про себе, реєструватися і авторизуватися на сайті.

У процесі створення web-системи буде використано такі технології:

MySQL — це система керування та роботою з базами даних, яка характеризується високою швидкістю, надійністю та простотою у використанні. MySQL надає великий набір функціональних можливостей, які мають безпечне середовище для зберігання, редагування та отримання даних.

PHP — це мова програмування, яка дозволяє генерувати HTML-сторінки на стороні сервера.

HTML (англ. HyperText Markup Language – мова розмітки гіпертекстових документів) — стандартна мова розмітки веб-сторінок в Інтернеті. Більшість веб-сторінок створюються за допомогою мови HTML (або XHTML). Документ HTML оброблюється браузером та відтворюється на екрані у звичному для людини вигляді.

CSS (англ. Cascading Style Sheets ) — це спеціальна мова стилю сторінок, що використовується для опису їхнього зовнішнього вигляду. Самі ж

сторінки написані мовами розмітки даних. CSS є основною технологією всесвітньої павутини, поряд із HTML та JavaScript.

Отже, для реалізації веб-сайту було обрано скриптову серверну мову програмування PHP та СУБД MySQL. MySQL складається з двох частин серверної і клієнтської. Сервер MySQL постійно працює на комп'ютері. Клієнтська програма, якою в даному випадку є PHP, посилає серверу MySQL SQL-запити через механізм сокетів (тобто за допомогою мережових засобів), сервер їх обробляє і запам'ятовує результат. Тобто скрипт (клієнт) вказує, яку інформацію він хоче отримати від сервера баз даних. Потім сервер баз даних посилає відповідь (результат) клієнтові (скрипту). Для роботи з MySQL буду використовувати PHP-функції, що призначені для роботи з СУБД. Також для обробки даних будуть використовуватись такі можливості PHP, як створення змінних, класів, використання логічних операцій порівняння даних тощо. Для розділення інформації, яка міститься на сайті та програми, яка її видає, використовуються бази даних MySQL. Вся інформація сайту буде зберігатися в базі даних, а скрипт PHP буде здійснювати вибірку необхідної інформації для конкретної веб-сторінки.

Для управління базами даних використаємо програму phpMyAdmin. phpMyAdmin – це сукупність скриптів написаних на PHP, що забезпечують повноцінну, в тому числі віддалену роботу з базами даних MySQL, через веб-інтерфейс.

Розгортання програмного продукту буде відбуватися за допомогою локального сервера XAMPP.

## РОЗДІЛ 2. РОЗРОБКА БАЗИ ДАНИХ

### 2.1. Розробка і опис моделі даних

В процесі проектування і опису моделі даних будується інфологічна модель бази даних. Для цього необхідно виділити всі об'єкти БД, зазначити їхні характеристики (атрибути) і встановити зв'язки між об'єктами. Інфологічний рівень – це інформаційно-логічну модель (ІЛМ) предметної області (ПО), в якій виключена надмірність даних і відображені інформаційні особливості об'єкта управління, без урахування особливостей і специфіки конкретної СУБД.

Для того, щоб виконати проектування бази даних необхідно зрозуміти логіку майбутнього сайту. Функціонал web-сайту можна забезпечити через наступні процеси:

- процес пошуку наявного туру;
- процес здійснення бронювання(заказу) туру.
- додавання інформації користувачем про себе;

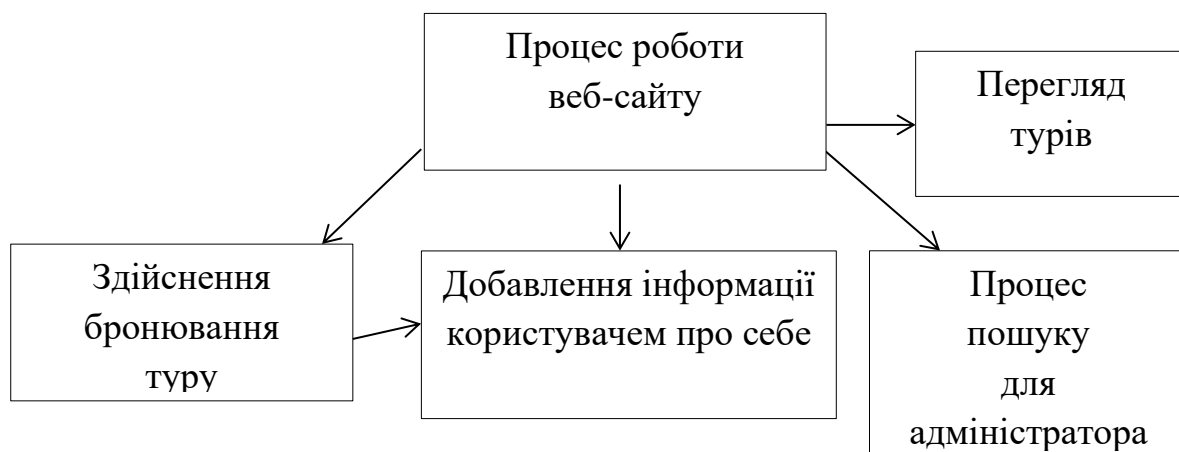


Рис. 2.1 – Діаграма функцій процесу роботи веб-сайту

Завершальним етапом проектування є створення концептуальної моделі бази даних, яка зображена на рисунку 2.2.

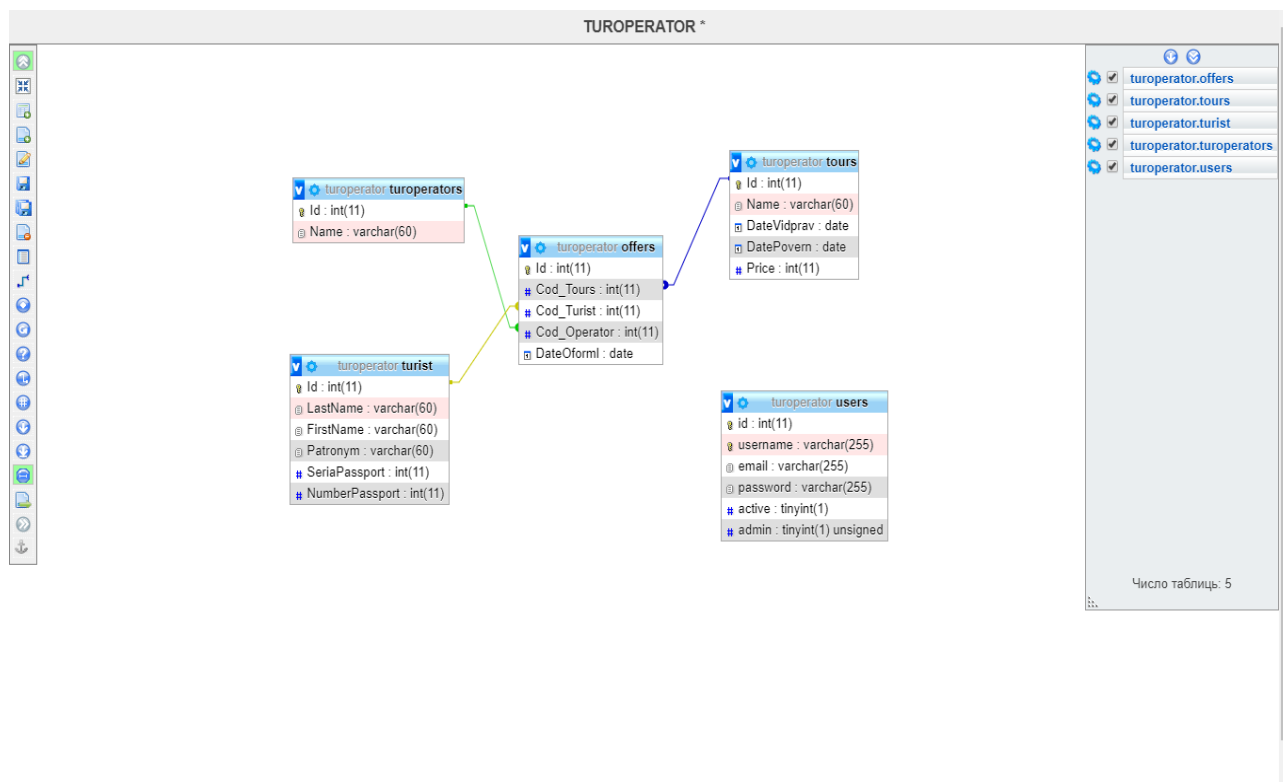


Рис. 2.2 – Концептуальна модель

### 2.1.1. Перелік таблиць бази даних

Таблицями бази даних є:

- turist – таблиця, у якій міститься основна інформація про туриста;
- turoperator – таблиця, у якій міститься ім'я туроператора;
- tours – таблиця, у якій знаходяться дані про тур;
- offers – таблиця, у якій міститься інформація про заказ;
- country – таблиця, у якій знаходиться інформація про країну;



е) users – таблиця, яка знаходиться в базі даних registrations і містить в собі дані про авторизованих користувачів та адміністратора;

### **2.1.2. Перелік полів таблиць бази даних**

Поля, які ідентифікують властивості таблиці «tulist»:

- а) id – ідентифікаційний код туриста;
- б) LastName – прізвище туриста;
- в) FirstName – ім'я туриста;
- г) Patronym – по-батькові;
- д) SeriaPassport – серія паспорта туриста;
- е) NumberPassport – номер паспорта;

Поля, які ідентифікують властивості таблиці «turoperator»:

- а) id – ідентифікаційний код туроператора;
- б) Name – ім'я туроператора;

Поля, які ідентифікують властивості таблиці «tours»:

- а) id – ідентифікаційний код номеру;
- б) Name – назва категорії номеру;
- в) DateVidprav – дата відправлення туриста;
- г) DatePovern – дата повернення туриста;
- д) price – ціна туру.

Поля, які ідентифікують властивості таблиці «offers»:

- а) id – ідентифікатор заказу;
- б) Cod\_Tours – код тура;
- в) Cod\_Turist – код туриста;
- г) Cod\_Operatora – код туроператора;
- д) DateOforml – дата оформлення заказу.

Поля, які ідентифікують властивості таблиці «users» бази даних registrations:

- а) id – ідентифікатор авторизованого користувача;
- б) username – логін користувача;
- в) email – електронна адреса користувача;
- г) password – пароль користувача;
- д) admin – ідентифікатор адмінства;

## **2.2. Нормалізація реляційних відношень**

Нормалізація представляє процес розподілу даних по окремим пов'язаних таблиць. Нормалізація усуває надмірність даних (data redundancy) і тим самим уникнути порушення цілісності даних при їх зміні, тобто уникнути аномалій зміни (update anomaly).

Як правило, нормалізація переважно застосовується при висхідному підході проектуванні бази даних, тобто коли ми всі атрибути, які треба зберегти в базі даних, групуємо за сутностями, для яких потім створюються таблиці. Однак при низхідному підході, коли спочатку виявляються сутності, а потім їх атрибути та зв'язки між ними, нормалізація також може застосовуватися, наприклад, для перевірки коректності розроблених таблиць.

У ненормалізованій формі таблиця може зберігати інформацію про двох і більше сутності. Також вона може містити повторювані стовпці. Також стовпці можуть зберігати повторювані значення. В нормалізованій же формі кожна таблиця зберігає інформацію лише про одну сутності.

Нормалізація передбачає застосування нормальних форм до структури даних. Існують 7 нормальних форм. Кожна нормальна форма (за винятком першого) передбачає, що до даних вже була застосована

попередня нормальна форма. Наприклад, перш ніж застосувати третю нормальну форму даних повинна бути застосована друга нормальна форма. Та строго кажучи, база даних вважається нормалізованою, якщо до неї застосовується третя нормальна форма і вище.

Перша нормальна форма (1NF) передбачає, що зберігаються дані на перетині рядків і стовпців повинні представляти скалярне значення, а таблиці повинні містити повторюваних рядків.

Друга нормальна форма (2NF) передбачає, що кожен стовпець, який не є ключем, повинен залежати від первинного ключа.

Третя нормальна форма (3NF) передбачає, що кожен стовпець, який не є ключем, повинен залежати тільки від первинного ключа.

В таблицях даної курсової роботи кожна таблиця має ідентифікатор, який має значення первинного ключа (Primary Key), причому три поля таблиці offers мають значення зовнішнього ключа (Foreign Key) і зв'язані з полем id таблиці Tours.

Зв'язок між цими полями показаний на ER-діаграмі:

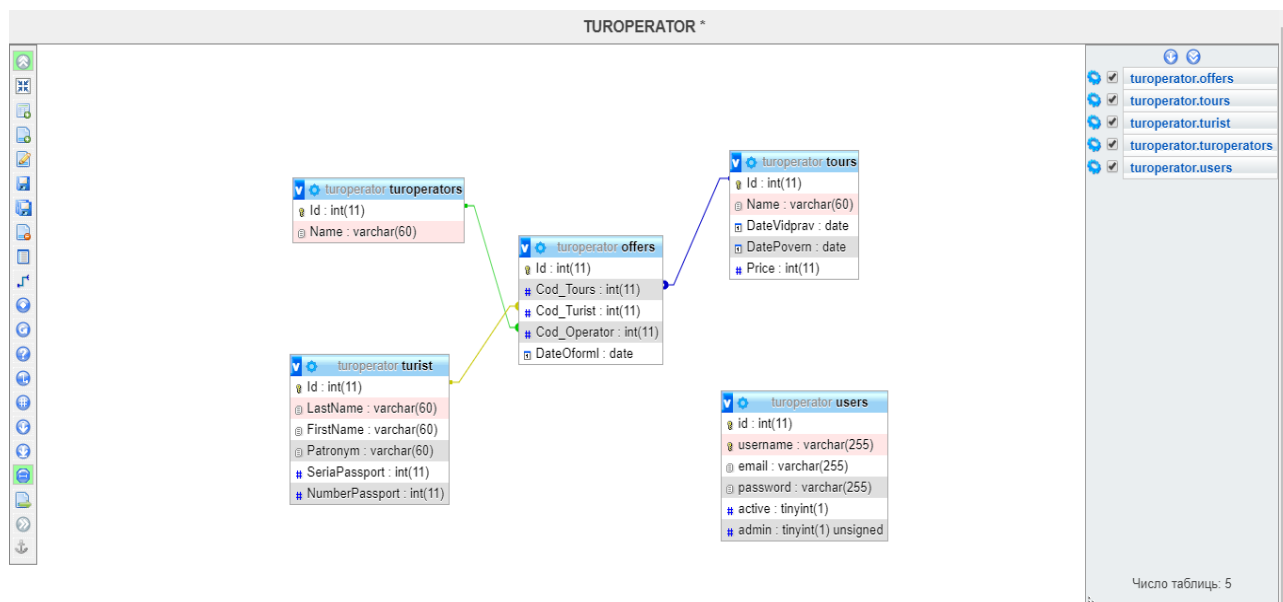


Рис.2.3 ER-діаграма зв'язку між таблицями

### 2.3. Визначення типів даних

Структура MySQL трирівнева: бази даних – таблиці – записи. Бази даних і таблиці MySQL фізично представляються файлами з розширеннями `frm`, `MYD`, `MYI`. Ім'я бази даних MySQL унікальна в межах системи, а таблиці - в межах бази даних, поля - в межах таблиці.

База даних з точки зору MySQL - це звичайний каталог, що містить файли певного формату - таблиці. Таблиці складаються із записів, а записи, у свою чергу, складаються з полів. Поле має два атрибути - ім'я і тип даних.

Основні типи даних, які використовуються в базі даних MySQL:

`VARCHAR` – може зберігати не більше 255 символів. На відміну від `CHAR`, для зберігання значення даного типу виділяється необхідна кількість пам'яті;

`TEXT` – може зберігати не більше 65 535 символів;

`CHAR` – використовується для зберігання рядків. Параметр в дужках дозволяє фіксувати довжину збереженої рядка. Максимальний розмір у байтах, який можна задати для рядка, – 255;

`INT` – діапазон від -2 147 483 648 до 2 147 483 647;

`DATE` – дата в форматі `PPPP-MM-ДД`;

`DATETIME` – дата і час в форматі `PPPP-MM-ДД ГГ: ХХ: СС`.

Для початку, в кореновому каталозі `phpMyAdmin` створимо нову базу даних, яка міститиме таблиці даних туристичного оператора. У рядку «Створити нову БД» введемо назву бази даних «`turoperator`» і обираємо кодування `utf8_general_ci`. Таке кодування дозволяє зберігати в таблицях даних символи кирилиці (при використанні іншого типу кодування, текстова інформація веб-сторінки може відображатись некоректно). Отже, після того, як створена база даних «`turoperator`»,

наповнимо її таблицями, які будуть містити інформацію зі сторінок веб-сайту.

Таблиця 2.1.

Склад таблиці «turist»

№п/п	Найменування атрибутів	Тип полів	Розмір полів
1.	id	INT	11
2.	LastName	VARCHAR	60
3.	FirstName	VARCHAR	60
4.	Patronym	VARCHAR	60
5.	SeriaPassport	INT	11
6.	NumberPassport	INT	11

Таблиця 2.2.

Склад таблиці «offers»

№п/п	Найменування атрибутів	Тип полів	Розмір полів
1.	id	INT	11
2.	Cod_Tours	INT	11
3.	Cod_Turist	INT	11
4.	Cod_Operatora	INT	11
5.	DateOfforml	DATE	

Таблиця 2.3.

Склад таблиці «turoperator»

№п/п	Найменування атрибутів	Тип полів	Розмір полів
1.	id	INT	11

2.	Name	VARCHAR	60
----	------	---------	----

Таблиця 2.4.

Склад таблиці «users»

№п/п	Найменування атрибутів	Тип полів	Розмір полів
1.	id	INT	11
2.	username	VARCHAR	255
3.	email	VARCHAR	255
4.	password	VARCHAR	255
5.	admin	TINYINT	1

Таблиця 2.5.

Склад таблиці «tours»

№п/п	Найменування атрибутів	Тип полів	Розмір полів
1.	id	INT	11
2.	Name	VARCHAR	60
4.	DateVidprav	DATE	
5.	DatePovern	DATE	
6.	Price	INT	11

## 2.4. Запити до таблиць бази даних

Web-сайт реалізує такі запити до бази даних:

Запит типу «INSERT» :

- (INSERT INTO users (username, email, password) VALUES ('\$username', '\$email' , '\$password' );) використовується для створення і додавання нового користувача до бази даних;

Прикладом використання запиту INSERT у коді є:

```
$query = "INSERT INTO users (username, email, password) VALUES ('$username','$email', '$password')";
```

```
$result = mysqli_query($connection, $query);
```

```
if($result){
```

```
    $smmsg = "Регістрація пройшла успішно";
```

```
    } else {
```

```
        $fsmmsg = "Помилка";
```

```
    }
```

```
}
```

- (INSERT INTO tours (Name, DateVidprav, DatePovern, Price) VALUES ( '\$name', '\$vidprav', '\$povern', '\$price');) використовується для здійснення бронювання туру.

Прикладом використання запиту INSERT у коді є:

```
$sql = "INSERT INTO tours (Name, DateVidprav, DatePovern, Price)
```

```
VALUES ( '$name', '$vidprav', '$povern', '$price')";
```

```
if ($conn->query($sql) === TRUE) {
```

```
    echo "Дані відправлені";
```

```
} else {
```

```
    echo "Помилка: " . $sql . "<br>" . $conn->error;
```

```
}
```

```
$conn->close();
```

- (INSERT INTO turist (LastName, FirstName, Patronym, SeriaPassport, NumberPassport) VALUES ( '\$lastname', '\$firstname', '\$patronym', '\$seriapassport', '\$numberpassport') відбувається додавання інформації про користувача(туриста) про себе в базу даних;

Прикладом використання запиту INSERT у коді є:

```
$sql = "INSERT INTO turist (LastName, FirstName, Patronym, SeriaPassport,
NumberPassport)
VALUES ( '$lastname', '$firstname', '$patronym', '$seriapassport',
'$numberpassport')";
if ($conn->query($sql) === TRUE) {
    echo "Дані відправлені";
} else {
    echo "Помилка: " . $sql . "<br>" . $conn->error;
}
$conn->close();
```

Запит типу «SELECT» :

- (SELECT Name, DateVidprav, DatePovern, Price FROM tours;)

здійснюється вибірка зазначених даних і вивід інформації на сторінку.

Прикладом використання запиту SELECT у коді є:

```
$sql = "SELECT Name, DateVidprav, DatePovern, Price FROM tours";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
```



```

echo "Назва тура: " . $row["Name"]. " <br> Дата відправлення: " .
$row["DateVidprav"]. " <br> Дата повернення: " . $row["DatePovern"]. "
<br>Ціна: " . $row["Price"]. " <br>";
}
} else {
echo "0 results";
}
$conn->close();

```

- (SELECT DateVidprav, Name, DatePovern FROM tours  
WHERE ".\$DateVidprav." LIKE '%"'.\$FirstName.'"%";) відбувається вибірка  
певних елементів з таблиці за критерієм;

Прикладом використання запиту SELECT у коді є:

```

$sql = "SELECT DateVidprav, Name, DatePovern FROM tours
WHERE ".$DateVidprav." LIKE '%"'.$FirstName.'"%";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo "Дата відправлення: " . $row["DateVidprav"]. " Назва: " .
        $row["Name"]. " - Дата повернення: " . $row["DatePovern"]. "<br>";
    }
} else {
echo "0 results";
}
$conn->close();

```

## **РОЗДІЛ 3. РОЗРОБКА ВЕБ-САЙТУ**

### **3.1. Інструменти для створення веб-сайту**

Для створення веб-сайту було використано локальний веб-сервер ХАМРР, який дає можливість розміщувати власні сайти на локальному сервері.

Цей веб-сервер входить в список уже скомплектованих пакетів програми, до якого входять Apache, PHP, MySQL, phpMyAdmin.

- ХАМРР;
- Denwer;
- OpenServer

В базовий пакет ХАМРР включає в себе такі складові:

- Web-server Apache;
- СУБД MySQL;

Програму для адміністрування та управління базою даних phpMyAdmin.

### **3.2. Структура веб-сайту**

Веб-сайт складається з index.php(головна сторінка з формою реєстрації), login.php (сторінка авторизації при натисканні кнопки Вхід), index.php в папці admin(сторінка для входу адміністратора після натискання кнопки Адмін Вхід), search.php (при натисканні однієї з головних вкладок Пошук здійснюється перехід на сторінку Пошуку інформації за визначеними критеріями для адміністратора), після введення необхідних полів та натискання кнопки Пошук здійснюється перехід на сторінку файлу search1.php (відображається результат пошуку), при переході на вкладку меню Заказати відкривається сторінка файлу offers.php, де знаходиться форма Заказ туру, де вводиться необхідна інформація і заноситься в базу даних.

Після вводу інформації відбувається перехід на сторінку файлу offers1.php(на якій є форма додавання інформації про користувача), після появляється сторінка offers2.php(де виводиться інформація про успішне додавання даних), при переході на вкладку Пакет турів відкривається сторінка файлу tourlist.php, де виводиться інформація про наявні тури, в logout.php знаходить код для видалення інформації про користувача який вийшов з акаунту, файл content.php містить php-код для підключення до бази даних.

### 3.3. Програмування серверної частини

Для програмування серверної частини сайту був використаний можливості мови PHP.

Приклад програмування обробки форми реєстрації на стороні сервера:

```
<?php
require('connect.php');

if (isset($_POST['username']) && isset($_POST['password'])) {
    $username = $_POST['username'];
    $email = $_POST['email'];
    $password = $_POST['password'];

    $query = "INSERT INTO users (username, email, password) VALUES
('$username','$email', '$password')";
    $result = mysqli_query($connection, $query);

    if($result){
        $smmsg = "Реєстрація пройшла успішно";
    } else {
        $fsmmsg = "Помилка";
    }
}
```

```

}

}

?>

```

### 3.4. Програмування клієнтської частини

Для відображення та взаємодії з веб-сайтом було підключено стилі CSS, мову розмітки HTML та технологію Boostrapp при створенні форм для внесення даних. Весь код для відображення тексту та стилів знаходяться в файлах розширення .php

Приклад програмування відображення на веб-сторінці форми реєстрації:

```

<div class="container">
    <form class="form-singin" method ="POST">
        <h2>Реєстрація</h2>
        <?php          if(isset($smsg)){ ?><div      class="alert      alert-success"
role="alert"><?php echo $smsg;?></div><?php }?>
        <?php          if(isset($fsmg)){ ?><div      class="alert      alert-danger"
role="alert"><?php echo $fsmg;?></div><?php }?>
        <input      type="text"      name="username"      class="form-control"
placeholder="Username" required>
        <input      type="email"      name="email"      class="form-control"
placeholder="Email" required>
        <input      type="password"      name="password"      class="form-control"
placeholder="Password" required>
        <button      class="btn      btn-lg      btn-primary      btn-block"
type="submit">Реєстрація</button>
        <a href="login.php" class="btn btn-lg btn-primary btn-block">Вхід</a>
        <a href="admin/index.php" class="btn btn-lg btn-primary btn-
block">Адмін Вхід</a>

```

</form>

</div>

### **3.5. Розміщення веб-сайту на локальному сервері.**

Для роботи даного веб-сайту на локальному сервері XAMPP потрібно всі необхідні файли та ресурси помістити в папку htdocs кореневої папки хампр. Після за посиланням <http://localhost:82/PHP1/index.php> зайти на головну сторінку веб-сайту.

## ВИСНОВКИ

Метою даної курсової роботи було створення бази даних «Turoperator». Створена база даних MySQL та СУДБ можуть бути застосовані в туристичному агенстві для зберігання та ведення обліку всіх поїздок, обліку всіх співробітників, обліку всіх путівок та їх цін. База даних створена і застосована, що говорить про виконану мету курсової роботи.

У процесі виконання даної курсової роботи було досліджено і протестовано можливості технологій MySQL, PHP, HTML та CSS. В результаті дослідження закріплено отримані знання і створено web-орієнтовану інформаційну систему для роботи з клієнтами туристичного оператора, досліджені можливості використання створеного веб-сайту, крім того створено і спроектовано базу даних, реалізовані запити для доступу до цих даних в базі, під'єднано і налаштовано зв'язок між базою даних і веб-сайтом.

У Розділі 1, в якій знаходиться теоретична частина, описано технічне завдання для розробки програмного продукту.

У Розділі 2 було описано процес розробки бази даних: встановлено предметну область, проаналізовано нормалізацію реляційних відношень, описано типи даних, сформовано та складено запити для повноцінної роботи з базою даних та пов'язаним з нею сайтом. Процес проектування відбувається в програмному середовищі для адміністрування та полегшення роботи з базою даних phpMyAdmin. База даних складається з п'яти таблиць, яка має функціональні залежності між атрибутами таблиць та встановлені зв'язки між елементами створеної бази даних, перевірена на відповідність за трьома нормальними формами (1НФ, 2НФ, 3НФ). Крім цього створено діаграми зв'язків для демонстрації моделі даних.

У Розділі 3 описуються етапи розробки веб-сайту. По-перше в розділі спочатку було представлено структуру сайту та інструменти для

створення сайту . Далі було представлено деякі частини коду: серверні скрипти, які є характерною складовою програмування серверної частини веб-сайту, в основному це стосується форм на сайті для зв'язку та взаємодії користувача і адміністратора з базою даних.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- Гайдоржи В.І. :Основи проектування та використання баз даних [Текст]: Навч. Посібник / В.І. Гайдоржи, О.А. Дацюк; Нац. Техн. Ун-т України «Київський політехн. Ін-т». 2-ге вид, випр. І доп. – К.: Політехніка, 2004. – 254с.
- Дейт К. Дж. Введення в системи баз даних / [Дейт К. Дж] – 8-е видання .: Пер.
- З англ. – М .: Видавничий дім «Вільямс», 2005. -1328 с.
- Конноллі Т.: Бази даних. Проектування, реалізація і супровід. Теорія і практика / Конноллі Т., Бегг К. – [3-е видання. : Пер. З англ.] – М.: Видавничий дім «Вільямс», 2003. – 1440 с.
- Пітер Лабберс, Брайан Олберс, Френк Салім: HTML5 для професіоналів. Потужні інструменти для розробки сучасних веб-додатків – СПб .: Видавничий дім «Пітер», 2011. – 272 с.
- Харрінгтон Дж. Л.: Проектування реляційних баз даних / [Харрінгтон Дж. Л.] – М .: Видавництво «Лорі», 2006 – 232 с.
- РНР [Електронний ресурс]. <https://en.wikipedia.org/wiki/RНР>
- Нормалізація баз даних [Електронний ресурс]. <https://uk.wikipedia.org/wiki>
- Реляційна база даних [Електронний ресурс]. <https://uk.wikipedia.org/wiki>











## Додаток Б

## Список доступних турів

"Мандри"



Головна

Про нас

Пакет турів

Події

Заказати

Пошук

Туроператор "Мандри".

Список доступних турів.

Кожний тур розрахований на 10 днів

Назва тура: Тур по Україні

Дата відправлення: 2020-01-01

Дата повернення: 2020-01-10

Ціна: 200

Назва тура: Тур по Європі

Дата відправлення: 2020-01-10

Дата повернення: 2020-01-20

Ціна: 600

Назва тура: Тур по Китаї

Дата відправлення: 2020-01-10

Дата повернення: 2020-01-20

Ціна: 800

Назва тура: Тур по Канаді

Дата відправлення: 2020-01-10

Дата повернення: 2020-01-20

Ціна: 700

Назва тура: Тур по Німеччині

Дата відправлення: 2020-01-20

Дата повернення: 2020-01-30

Ціна: 250

Назва тура: Тур по Азії

Дата відправлення: 2020-02-01

Дата повернення: 2020-02-10

Ціна: 350

Назва тура: Тур по Африці

Дата відправлення: 2020-02-01

Дата повернення: 2020-02-10

Ціна: 850

Назва тура: Тур по Південній Азії

Дата відправлення: 2020-02-10

Дата повернення: 2020-02-20

Ціна: 900

Назва тура: Тур по Близькому Сходу

Дата відправлення: 2020-02-10

Дата повернення: 2020-01-20

Ціна: 1000

Назва тура: Тур по Островам Тихого океану

Дата відправлення: 2020-03-01

Дата повернення: 2020-03-10

Ціна: 1500

Назва тура: Тур по Бразилії

Дата відправлення: 2020-03-01

Дата повернення: 2020-03-10

Ціна: 600

Назва тура: Тур по Штатам

Дата відправлення: 2020-03-01

Дата повернення: 2020-03-10

Ціна: 850

Назва тура: Тур по Франції

Дата відправлення: 2020-03-10

Дата повернення: 2020-03-20

Ціна: 450

Назва тура: Тур по Австралії

Дата відправлення: 2020-03-10

Дата повернення: 2020-03-20


Ціна: 1050




Copyright Information goes here.

Всі права захищені

Наші авторизовані Агенти



Famous Cook



Навколо Світу

## Додаток В

## Форма для бронювання

The screenshot displays the 'Мандри' (Mandry) website interface. On the left, there is a vertical sidebar with the title '"Мандри"' and a collage of travel-related images including a Las Vegas sign, a camel, people in costumes, a boat, a couple with a surfboard, a windmill, and a group of people. The main content area has a dark red header with navigation links: Головна, Про нас, Пакет турів, Події, Заказати, and Пошук. Below the header, the title 'Туроператор "МАНДРИ" Заказ туру' is centered. The booking form consists of several input fields: 'Назва' (Name), 'mm/dd/yyyy' (Start date), 'mm/dd/yyyy' (End date) with a dropdown arrow, and 'Ціна' (Price) with a tooltip that says 'Please fill out this field.'. A blue 'Заказати' (Book) button is positioned below the form. At the bottom left of the main area, there is a small image of a hot air balloon.

## Додаток Г

## Форма для логування користувача

The screenshot shows the 'Мандри' website with the login and registration section. The header is dark red with the same navigation links as in the previous screenshot. Below the header, the title 'Вхід' (Login) is centered. The login form includes 'Username' and 'Password' input fields, followed by a blue 'Вхід' (Login) button and a blue 'Реєстрація' (Registration) button. Below these buttons, there is a link 'Немає акаунту? Зареєструйтесь' and a blue 'Вийти' (Logout) button. On the left, the same sidebar with the title '"МАНДРИ"' and travel images is visible. The main content area below the login form contains a welcome message: 'Ласкаво просимо на наш сайт Туроператор МАНДРИ'. It follows with a paragraph about the company's services, a paragraph about the benefits of traveling, and a paragraph about the variety of flight options. The text is in Ukrainian and describes the company's offerings and the value of travel.

## Додаток Д

## Форма для додавання даних про користувача

The screenshot shows the website 'МАНДРИ' with a dark red header and sidebar. The main content area is white and contains the following elements:

- Header:** 'МАНДРИ' logo and navigation links: Головна, Про нас, Пакет турів, Події, Заказати, Пошук.
- Left Sidebar:** A vertical strip of images including a 'LAS VEGAS' sign, a camel, people in costumes, a couple on a boat, a windmill, and a group of people.
- Main Content Area:**
  - Text: 'Дані відправлені' (Data sent).
  - Section: 'Туроператор "МАНДРИ" Додавання даних про користувача' (Travel operator 'MANDRY' Adding user data).
  - Form fields:
    - Прізвище: (Last name)
    - Ім'я: (First name)
    - По-батькові: (Patronymic)
    - Серія паспорта: (Passport series)
    - Номер паспорта: (Passport number)
  - Button: 'Добавити' (Add).

## Додаток Д

## Форма для пошуку даних про туриста

The screenshot shows the website 'МАНДРИ' with a dark red header and sidebar. The main content area is white and contains the following elements:

- Header:** 'МАНДРИ' logo and navigation links: Головна, Про нас, Пакет турів, Події, Заказати, Пошук.
- Left Sidebar:** A vertical strip of images including a 'LAS VEGAS' sign, a camel, people in costumes, a couple on a boat, a windmill, and a group of people.
- Main Content Area:**
  - Section: 'Туроператор "МАНДРИ".' (Travel operator 'MANDRY').
  - Text: 'Пошук даних про туриста. Виберіть критерій пошуку: Дата відправлення ▼' (Search for tourist data. Select search criteria: Departure date ▼).
  - Form field: 'Введіть рядок:' (Enter text:).
  - Button: 'Пошук' (Search).

At the bottom of the page, there is a footer with the text 'Наші авторизовані Агенти' (Our authorized Agents) and logos for 'Famous Cook' and 'Навколо Світу' (Around the World).

