

	Dokumentacja projektu
Autorzy	Adrian Skowroński Michał Szukała
Temat projektu	<b>Biuro podróży – skocznie narciarskie świata</b>
Główne technologie	PHP, Laravel, MySQL, Bootstrap, JavaScript

## **Wstęp:**

Projekt to aplikacja internetowa, której celem jest ułatwienie miłośnikom sportów zimowych odkrywania i planowania wyjazdów na skocznie narciarskie na całym świecie. Użytkownik przeglądający stronę ma dostęp do szerokiej gamy funkcji, które umożliwiają przeglądanie i rezerwację aktualnych ofert biura podróży, interaktywną eksplorację wycieczek za pomocą mapy świata, a także możliwość złożenia zapytania o indywidualnie dopasowaną ofertę. Dodatkowo, platforma oferuje przestrzeń dla użytkowników do dzielenia się wrażeniami z podróży na blogu oraz zarządzanie kontem użytkownika za pomocą panelu klienta.

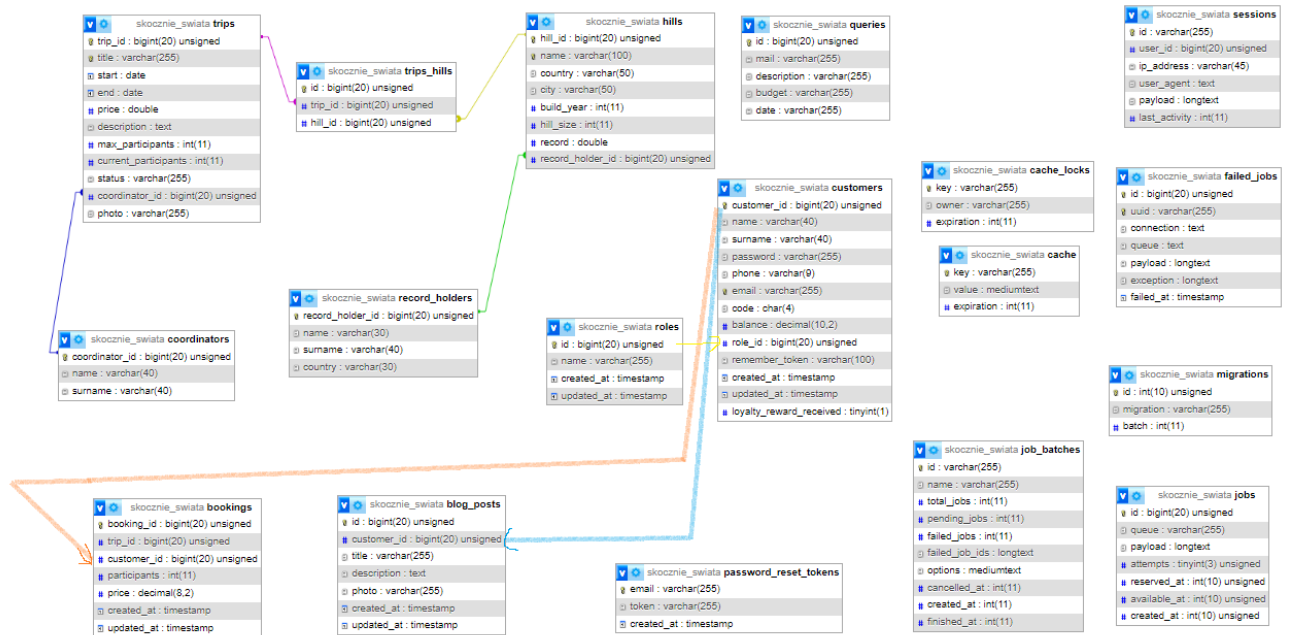
## **Narzędzia i technologie:**

W projekcie wykorzystywane są język programowania PHP w wersji 8.2.12 oraz JavaScript/Node.js w wersji v20.13.1. Projekt wykorzystuje bazę danych MySQL, którą zarządzamy za pomocą narzędzia phpMyAdmin. W projekcie wykorzystywane są biblioteki Bootstrap i Leaflet.js. Wykorzystywane frameworki to Laravel w wersji 11 oraz starter kit Laravla: Laravel Breeze. Wykorzystywany jest również framework CSS: Tailwind CSS. Podczas pracy wykorzystywany był ponadto rozproszony system kontroli wersji GIT (<https://github.com/adrian-skowronski/laravel-biuro-podrozy>) oraz oprogramowanie XAMPP.

## **Dokumentacja i Źródła:**

- PHP: <https://www.php.net/docs.php>
- Laravel: <https://laravel.com/docs>
- MySQL: <https://dev.mysql.com/doc/>
- phpMyAdmin: <https://docs.phpmyadmin.net/pl/latest/>
- JavaScript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- Node.js: <https://nodejs.org/en/docs/>
- Bootstrap: <https://getbootstrap.com/docs/>
- Leaflet.js: <https://leafletjs.com/reference.html>
- Tailwind CSS: <https://tailwindcss.com/docs>
- Laravel Breeze: <https://laravel.com/docs/starter-kits#breeze-and-blade>
- XAMPP: <https://www.apachefriends.org/index.html>

## Baza danych (ERD):

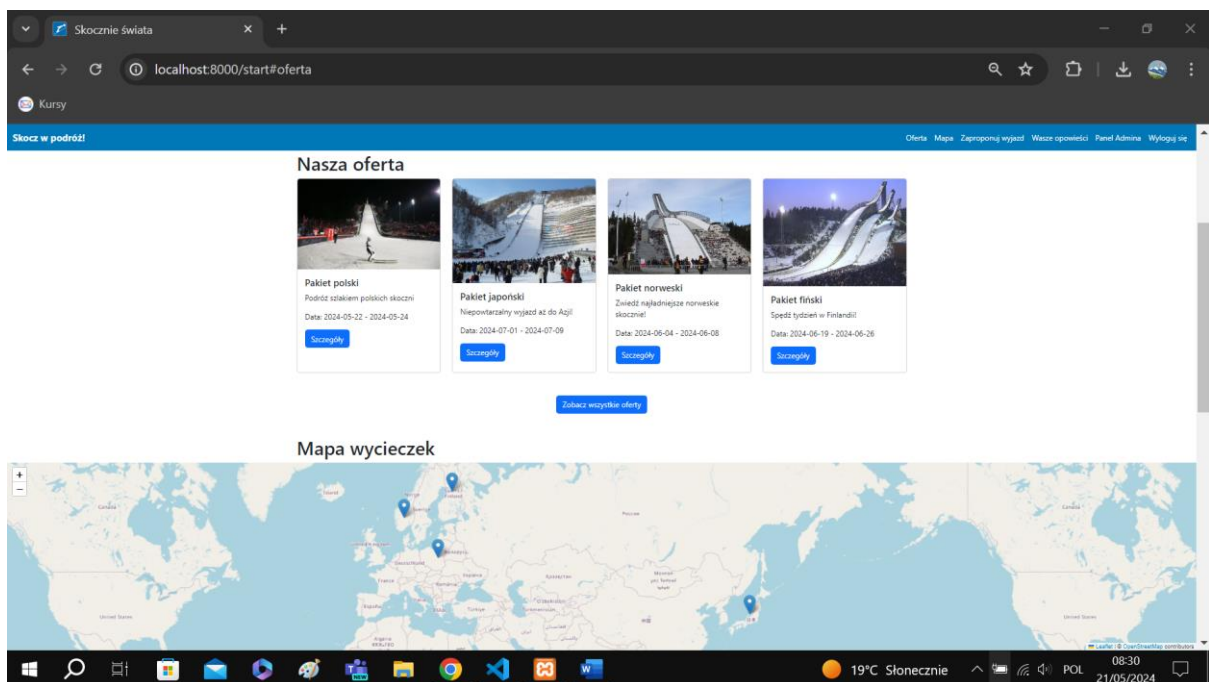
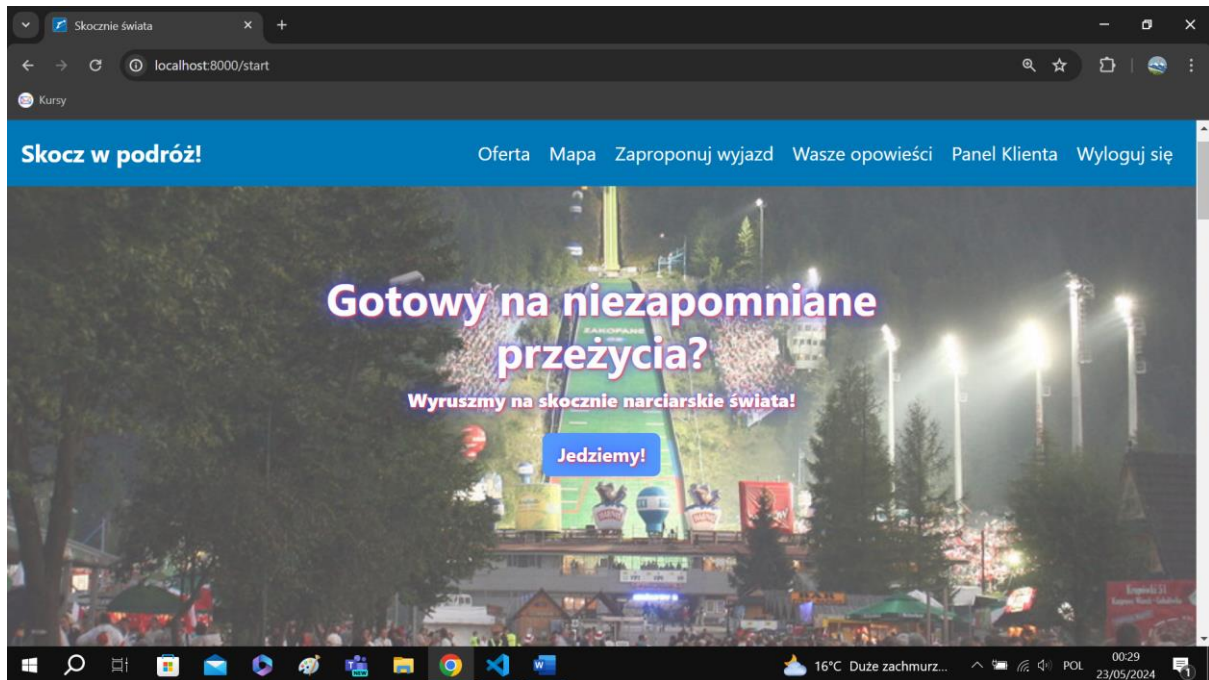


## Jak uruchomić aplikację:

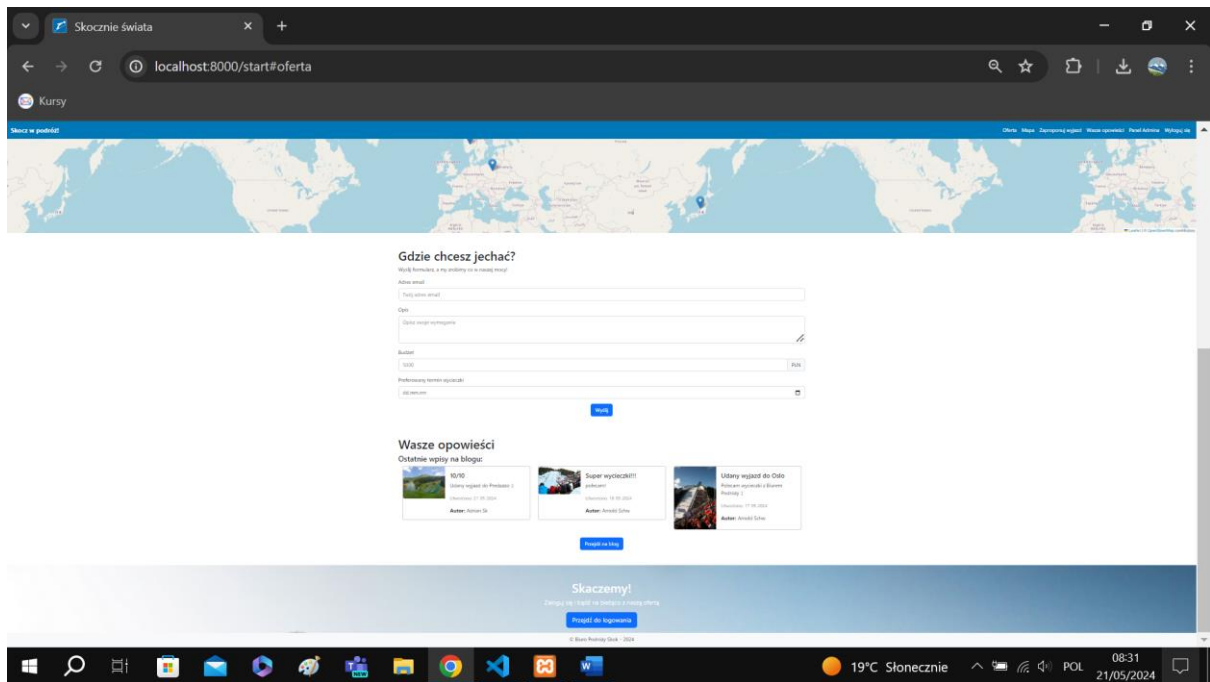
W celu uruchomienia aplikacji, potrzebujemy w XAMPPie włączyć MySQL i Apache. W folderze projektu, konsolowo uruchamiamy localhost, np. komendą `php artisan serve`. Aby zagwarantować działanie modułu uwierzytelniania użytkowników (logowanie, rejestracja, autoryzacja dostępu), należy również, komendą `npm run dev`, włączyć Node.js. W przeglądarce wpisujemy URL: `localhost:8000`, co uruchamia aplikację.

## Dokumentacja zdjęciowa działania aplikacji:

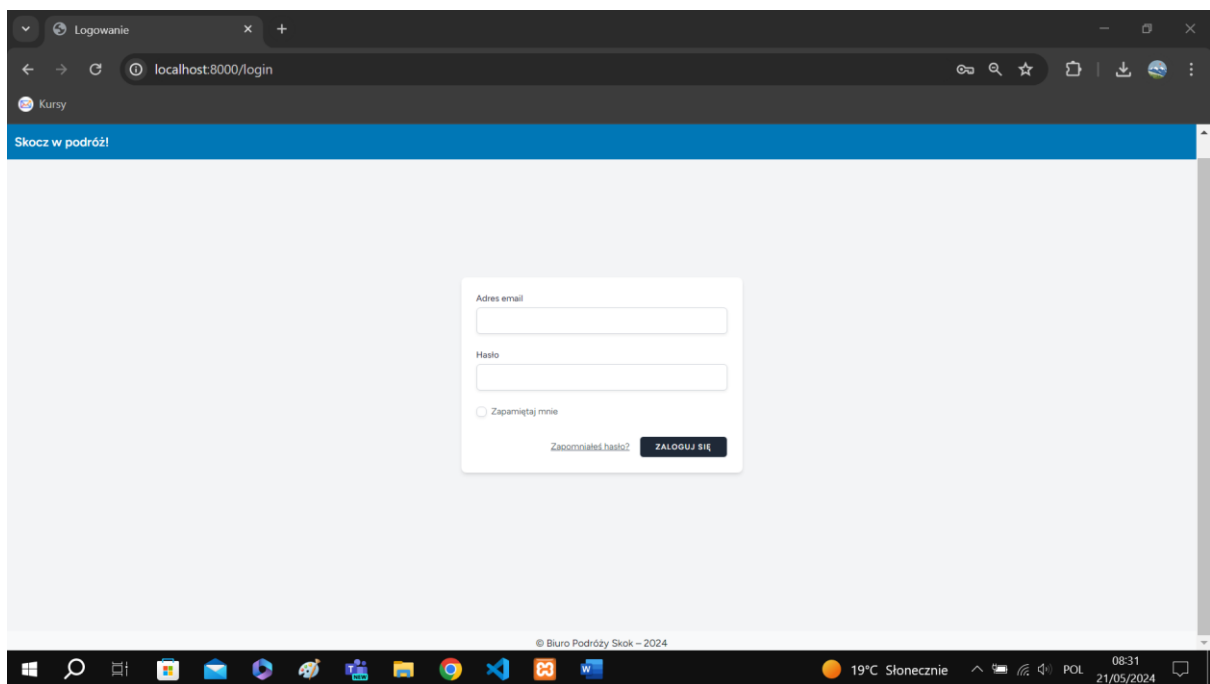
Widok startowy:



*Interaktywna mapa świata (biblioteka Leaflet.js) z zaznaczonymi wycieczkami – każda pinezka prowadzi do szczegółów danej wycieczki.*



Widok logowania:



Panel Klienta:

Pozostało 4 dni do Twojej następnej wycieczki.

Dziękujemy za wybór naszego Biura Podróży!  
Twoje saldo zostało jednorazowo zasilone 100 zł, ponieważ osiągnąłeś poziom 10.000 zł za dotychczasowe rezerwacje.

Dane Klienta

Pole	Wartość
ID	6
Imię	Piotr
Nazwisko	Fijas
Telefon	444555444
Email	piotr@mail.com
Kod	4444
Saldo	1005.00
Rola	Klient
Utworzony	2024-05-22 19:48:46
Ostatnia aktualizacja	2024-05-22 21:42:23

Historia Rezerwacji

Sortuj według ceny: Rosnąco Sortuj

ID	Wycieczka	Liczba uczestników	Cena	Data rezerwacji
22	Pakiet norweski	3	11697.00	2024-05-22 21:51:01

Doładuj saldo

Powrót do panelu

© Biuro Podróży Skok – 2024

Doładuj saldo

localhost:8000/klient/add\_money

Kursy

Skocz w podróż!

Panel Klienta Wyloguj się

Doładuj saldo

Kwota doładowania

Kod

Doładuj

Powrót do panelu

© Biuro Podróży Skok – 2024

W celu doładowania salda, zaimplementowaliśmy „atrapę płatności”, tzn. użytkownik przy rejestracji podaje „kod”, który musi wprowadzić przy doładowaniu salda, aby zostało ono zautoryzowane i zrealizowane.

Widok ofert:

Oferta


localhost:8000/oferty

Kursy

Skocz w podróż!

Panel Klienta Wyloguj się

## Nasza oferta



**Archiwalna**


**Pakiet polski**

Podróż szlakiem polskich skoczni

Data: 2024-05-20 - 2024-05-24

Cena: 1090 PLN/os.

Szczegóły



**Aktualna**


**Pakiet japoński**

Niepowtarzalny wyjazd aż do Azji!

Data: 2024-07-01 - 2024-07-09

Cena: 8799 PLN/os.

Szczegóły



**Ostatnie miejsce**


**Pakiet norweski**

Zwiedź najładniejsze norweskie skocznie!

Data: 2024-05-27 - 2024-06-08

Cena: 3899 PLN/os.

Szczegóły



**Koniec zapisów**

**Pakiet fiński**

Spędź tydzień w Finlandii!

Data: 2024-05-24 - 2024-06-26

Cena: 6099 PLN/os.

Szczegóły

© Biuro Podróży Skok – 2024

16°C Duże zachmurz...

00:33 23/05/2024

Wycieczka

localhost:8000/trips/4

Kursy

Skocz w podróż!

Panel Klienta Wyloguj się

## Szczegóły wycieczki

**Pakiet norweski**

Zwiedź najładniejsze norweskie skocznie!

Data rozpoczęcia: 2024-05-27

Data zakończenia: 2024-06-08

Cena: 3899 zł


Maksymalna liczba uczestników: 15

Aktualna liczba uczestników: 9

Koordinator: Egośław Linda

Zwiedzane skocznie:

Granasen  
Holmenkollbakken



Zarezerwuj wycieczkę

Wróć do listy wycieczek

© Biuro Podróży Skok – 2024

16°C Duże zachmurz...

00:33 23/05/2024

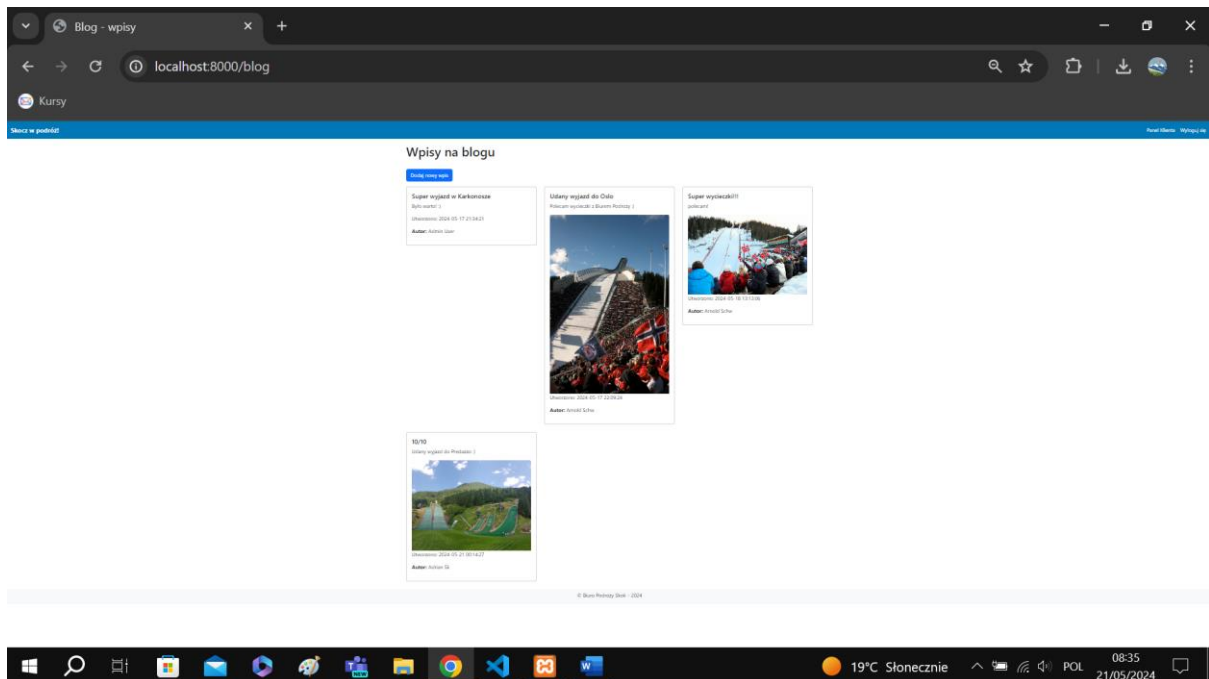
Dla wycieczek o statusach „Koniec zapisów” lub „Archiwalna”, możliwość rezerwacji została wyłączona.

Rezerwacja wycieczki:

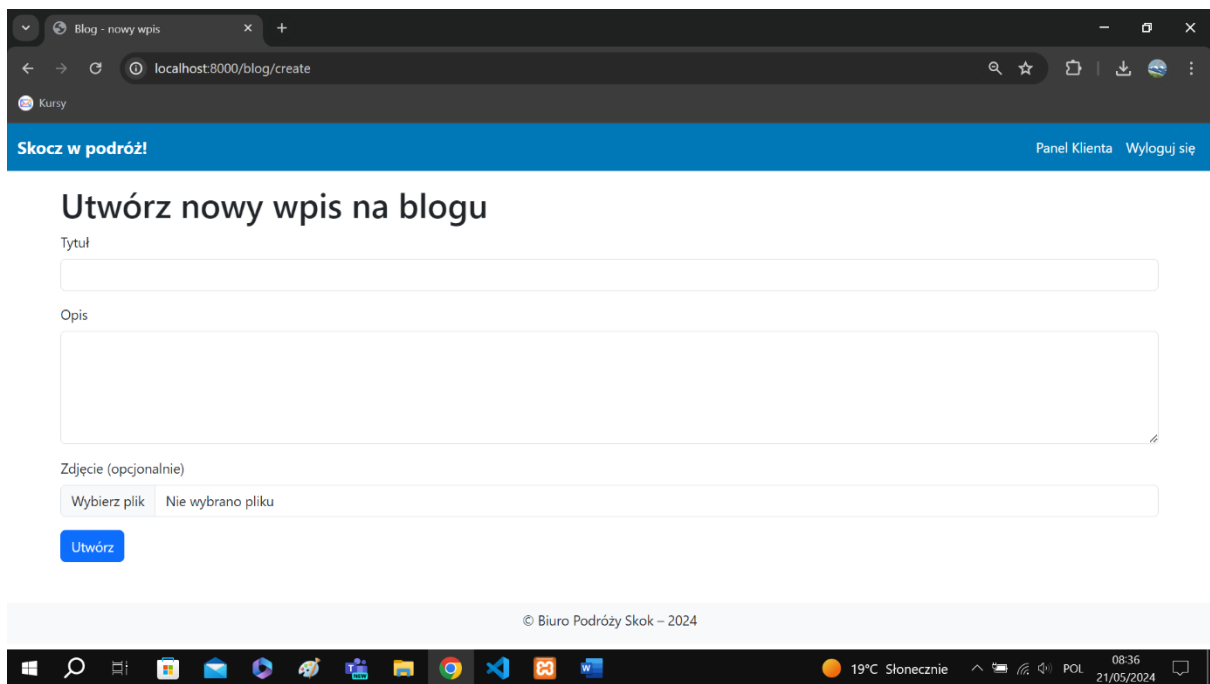
The screenshot shows a web browser window with the address bar displaying 'localhost:8000/trips/4/book'. The page has a blue header with the text 'Skocz w podróż!' and a 'Panel Klienta Wyloguj się' link. The main content area is titled 'Zarezerwuj wycieczkę' and 'Pakiet norweski'. It displays the start date 'Data rozpoczęcia: 2024-05-27' and the end date 'Data zakończenia: 2024-06-08'. Below this is a form for 'Liczba osób:' with a value of '3'. The total price is shown as 'Cena całkowita: 11697.00 zł /w tym podatek/'. There are two buttons: 'Zarezerwuj i zapłać' and 'Wróć do listy wycieczek'. The footer of the page says '© Biuro Podróży Skok – 2024'.

Zastosowany skrypt JS pozwala na automatyczne przeliczenie i wyświetlenie ceny całkowitej za wycieczkę podczas rezerwacji.

Blog:

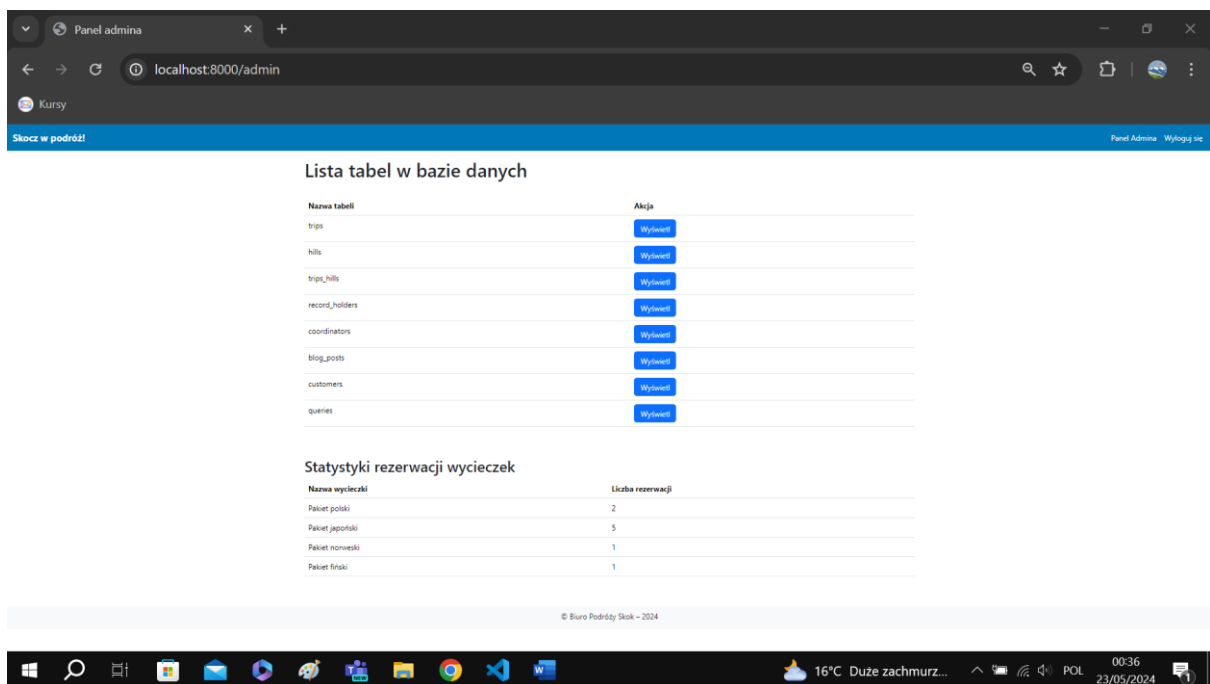






Wpisy na blogu mogą dodawać wyłącznie zalogowani użytkownicy.

Panel admina:



## Ranking wydanych pieniędzy przez klientów

Ranking	Id klienta	Imię	Nazwisko	Suma wydanych pieniędzy
1	7	kasia	Szukała	577296.00
2	4	Arnold	Schw	563136.00
3	3	Adrian	Sk	218644.00
4	6	Piotr	Fijas	35091.00
5	8	Michał	Szukała	3899.00

Ranking w panelu admina.

Wycieczki - lista - admin

localhost:8000/trips

Kursy

Skocz w podróż!

Panel Admina Wyloguj się

Lista wycieczek

[Dodaj nową wycieczkę](#)

#	Tytuł	Od	Do	Cena	Opis	Max_uczest.	Obecnie_uczest.	Status	Id koordynatora	Zdjęcie		
1	Pakiet polski	2024-05-20	2024-05-24	1090	Podróż szlakiem polskich skoczni	20	37	archiwalna	1	pakiet_polski.jpg	<a href="#">Edycja</a>	<a href="#">Usuń</a>
3	Pakiet japoński	2024-07-01	2024-07-09	8799	Niepowtarzalny wyjazd aż do Azji!	25	25	aktualna	3	pakiet_japonski.jpg	<a href="#">Edycja</a>	<a href="#">Usuń</a>
4	Pakiet norweski	2024-05-27	2024-06-08	3899	Zwiedź najładniejsze norweskie skocznie!	15	9	Ostatnia szansa	2	pakiet_norweski.jpg	<a href="#">Edycja</a>	<a href="#">Usuń</a>
5	Pakiet fiński	2024-05-24	2024-06-26	6099	Spędź tydzień w Finlandii!	30	3	koniec zapisów	1	pakiet_finski.jpg	<a href="#">Edycja</a>	<a href="#">Usuń</a>

Powrót do panelu admina

© Biuro Podróży Skok – 2024

16°C Duże zachmurz...


POL

00:37  
23/05/2024


## Dodaj nową wycieczkę

Tytuł

Od

Do

Cena

Opis

Maksymalna liczba uczestników

Aktualna liczba uczestników

Status

Id koordynatora

Nazwa zdjęcia

**Dodaj**

© Biuro Podróży Skok – 2024

*W panelu admina zaimplementowano operacje CRUD dla głównych tabel bazy danych*

### **Elementy języka proceduralnego wykorzystywane w projekcie:**

Do realizacji założeń aplikacji, wykorzystujemy elementy języka proceduralnego, które zaimplementowaliśmy w bazie danych (z poziomu phpMyAdmina). Są to:

- procedura obsługująca mechanizm rezerwacji wycieczki, z wykorzystaniem transakcji:

Details

Nazwa procedury

make\_booking

Typ

PROCEDURE

	Direction	Nazwa	Typ
Parametry	IN	p_customer_id	INT
	IN	p_trip_id	INT
	IN	p_participants	INT

Add parameter

Definition

```

1 BEGIN
2   DECLARE trip_price DECIMAL(10,2);
3
4   -- pobierz cenę wycieczki
5   SELECT price INTO trip_price FROM trips WHERE trip_id = p_trip_id LIMIT 1;
6
7   -- rozpocznij nową rezerwację
8   START TRANSACTION;
9
10  -- zaktualizuj saldo użytkownika
11  UPDATE customers
12  SET balance = balance - (p_participants * trip_price)
13  WHERE customer_id = p_customer_id;
14
15  -- zaktualizuj liczbę aktualnych uczestników wycieczki
16  UPDATE trips
17  SET current_participants = current_participants + p_participants
18  WHERE trip_id = p_trip_id;
19
20  -- dodaj nową rezerwację
21  INSERT INTO bookings (customer_id, trip_id, participants, price, created_at, updated_at)
22  VALUES (p_customer_id, p_trip_id, p_participants, p_participants * trip_price, NOW(), NOW());
23
24  -- zakończ rezerwację (transakcję)
25  COMMIT;
26 END

```

Ta procedura SQL służy do dokonywania rezerwacji wycieczek. Pobiera ona cenę wycieczki na podstawie jej identyfikatora, następnie rozpoczyna transakcję, aktualizuje saldo użytkownika oraz liczbę aktualnych uczestników wycieczki, dodaje rekord nowej rezerwacji do bazy danych i kończy transakcję, zapewniając integralność danych;

- procedura zliczająca liczbę rezerwacji danej wycieczki, z wykorzystaniem kursora:

**Details**

Nazwa procedury

count\_bookings\_with\_cursi

Typ

PROCEDURE ▼

Parametry

	Direction	Nazwa	Typ
⚡	IN ▼	p_trip_id	INT
⚡	OUT ▼	booking_count	INT

Add parameter

Definition

```
1 BEGIN
2   DECLARE done INT DEFAULT 0;
3   DECLARE booking_id INT;
4
5   DECLARE cur CURSOR FOR
6   SELECT booking_id FROM bookings WHERE trip_id = p_trip_id;
7
8   DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
9
10  SET booking_count = 0;
11
12  OPEN cur;
13
14  read_loop: LOOP
15    FETCH cur INTO booking_id;
16    IF done THEN
17      LEAVE read_loop;
18    END IF;
19    SET booking_count = booking_count + 1;
20  END LOOP;
21
22  CLOSE cur;
23 END
```

Kursor cur usprawnia w tej procedurze wyszukiwanie wycieczek po ID w tabeli bookings, co pozwala na zliczenie liczby rezerwacji danej wycieczki;

- procedura odliczająca dni do najbliższej zarezerwowanej wycieczki klienta, z wykorzystaniem kursora

Details

Nazwa procedury

calculate\_days\_to\_next\_trip

Typ

PROCEDURE

Parametry

	Direction	Nazwa	Typ	Długość/
#	IN	p_customer_id	INT	
#	OUT	p_days_left	INT	
#	OUT	p_message	VARCHAR	255

Add parameter

```
1 BEGIN
2   DECLARE done INT DEFAULT 0;
3   DECLARE trip_start_date DATE;
4   DECLARE days_left INT;
5   DECLARE min_days_left INT DEFAULT 9999;
6
7   DECLARE cur CURSOR FOR
8     SELECT t.start
9     FROM trips t
10    JOIN bookings b ON t.trip_id = b.trip_id
11    WHERE b.customer_id = p_customer_id AND t.start > CURDATE()
12    ORDER BY t.start ASC;
13  DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = 1;
14
15  OPEN cur;
16
17  read_loop: LOOP
18    FETCH cur INTO trip_start_date;
19    IF done THEN
20      LEAVE read_loop;
21    END IF;
22
23    SET days_left = DATEDIFF(trip_start_date, CURDATE());
24
25    IF days_left < min_days_left THEN
26      SET min_days_left = days_left;
27    END IF;
28  END LOOP;
29
30  CLOSE cur;
31
32  IF min_days_left < 9999 THEN
33    SET p_days_left = min_days_left;
34    SET p_message = CONCAT('Najbliższa wycieczka za ', min_days_left, ' dni.');

```
35  ELSE
36    SET p_days_left = NULL;
37    SET p_message = 'Nie masz wycieczek przed sobą';
38  END IF;
39 END
```


```

Kursor cur pozwala na sprawne wyszukanie wycieczek zarezerwowanych przez danego użytkownika, aby następnie procedura mogła ustalić liczbę dni pozostałych do najbliższej wycieczki. Informacja ta jest przekazywana do panelu klienta;

- procedura tworząca ranking klientów według całkowitej kwoty wydanych przez nich pieniędzy

Details

Nazwa procedury

GetCustomerRanking

Typ

PROCEDURE ▾

Parametry

Direction

Nazwa

Typ

Długość/Wartości

Add parameter

Definition

```
1 BEGIN
2   SELECT
3     customers.customer_id,
4     customers.name AS imie,
5     customers.surname AS nazwisko,
6     SUM(bookings.participants * bookings.price)
7   AS suma_wydanych_pieniedzy
8   FROM
9     customers
10    JOIN
11      bookings ON customers.customer_id =
12                bookings.customer_id
13   GROUP BY
14     customers.customer_id, imie, nazwisko
15   ORDER BY
16     suma_wydanych_pieniedzy DESC;
17 END
```

W tej procedurze z tabeli bookings sumuje się kwoty wszystkich rezerwacji dokonanych przez danego klienta, celem utworzenia rankingu w panelu admina.

- wyzwalacz aktualizujący status wycieczki

**Details**

Nazwa wyzwalacza	update_trip_status
Table	trips
Czas	BEFORE
Event	INSERT
Definition	<pre>1 BEGIN 2   IF NEW.start &gt; CURDATE() + INTERVAL 7 DAY THEN 3     SET NEW.status = 'aktualna'; 4   ELSEIF NEW.start &gt; CURDATE() + INTERVAL 2 DAY 5   THEN 6     SET NEW.status = 'Ostatnia szansa'; 7   ELSEIF NEW.start &gt; CURDATE() THEN 8     SET NEW.status = 'koniec zapisów'; 9   ELSE 10    SET NEW.status = 'archiwalna'; 11  END IF; 12 END</pre>
Definer	root@localhost

Wyzwalacz pomaga w utrzymaniu logiki biznesowej, która zakłada, że nie można rezerwować wycieczek, które rozpoczną się za 3 lub mniej dni. Tym wycieczkom, do których pozostał tydzień lub mniej, przypisuje się status „Ostatnia szansa”, aby zachęcić klientów do rezerwacji;



- wyzwalacz obsługujący przyznanie klientowi nagrody lojalnościowej

Details	
Nazwa wyzwalacza	loyalty_reward
Table	bookings
Czas	AFTER
Event	INSERT
Definition	<pre>1 BEGIN 2   DECLARE total_reservations DECIMAL(10, 2); 3   DECLARE reward_received BOOLEAN; 4 5   -- Obliczenie sumy wszystkich rezerwacji danego klienta 6   SELECT SUM(price) INTO total_reservations 7   FROM bookings 8   WHERE customer_id = NEW.customer_id; 9 10  -- Sprawdzenie, czy klient już otrzymał nagrodę lojalnościową 11  SELECT loyalty_reward_received INTO reward_received 12  FROM customers 13  WHERE customer_id = NEW.customer_id; 14 15  -- Sprawdzenie, czy suma przekracza 10 000 zł i nagroda nie została jeszcze przyznana 16  IF total_reservations &gt; 10000 AND reward_received = FALSE THEN 17    -- Dodanie 100 zł do salda klienta 18    UPDATE customers 19    SET balance = balance + 100, loyalty_reward_received = TRUE 20    WHERE customer_id = NEW.customer_id; 21  END IF; 22 END</pre>

Wyzwalacz ustawia pole „reward\_received” w tabeli customers wartość true, pod warunkiem, że łączna kwota wszystkich rezerwacji danego użytkownika przekroczy 10.000 zł. Usprawnia to przyznanie jednorazowej nagrody lojalnościowej w postaci zasilenia salda użytkownika 100 zł;

- widoki obsługujące sortowanie wycieczek danego klienta po cenie

Sortowanie po cenie rosnąco:

✓ Pokazano wiersze 0 - 7 (8 total, Wykonanie zapytania trwało 0,0005 sekund(y).)

```
SELECT * FROM `sortedbookingsasc`
```

☐ Profilowanie [ Edytuj w linii ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

☐ Show all | Liczba wierszy: 25 ▼ Filter rows:

Extra options

		booking_id	trip_id	customer_id	participants	price	created_at	updated_at
<input type="checkbox"/>	Edit  Copy  Delete	1	1	3	2	2180.00	2024-05-20 18:21:42	2024-05-20 18:21:42
<input type="checkbox"/>	Edit  Copy  Delete	2	1	3	3	3270.00	2024-05-20 19:32:44	2024-05-20 19:32:44
<input type="checkbox"/>	Edit  Copy  Delete	22	4	6	3	11697.00	2024-05-22 21:51:01	2024-05-22 21:51:01
<input type="checkbox"/>	Edit  Copy  Delete	20	3	3	2	17598.00	2024-05-21 18:16:49	2024-05-21 18:16:49
<input type="checkbox"/>	Edit  Copy  Delete	21	3	3	2	17598.00	2024-05-22 21:44:26	2024-05-22 21:44:26
<input type="checkbox"/>	Edit  Copy  Delete	4	5	3	3	18297.00	2024-05-21 08:00:25	2024-05-21 08:00:25
<input type="checkbox"/>	Edit  Copy  Delete	19	3	3	3	26397.00	2024-05-21 15:07:31	2024-05-21 15:07:31
<input type="checkbox"/>	Edit  Copy  Delete	3	3	4	8	70392.00	2024-05-21 00:06:21	2024-05-21 00:06:21

Sortowanie po cenie malejąco:

Pokazano wiersze 0 - 7 (8 total, Wykonanie zapytania trwało 0,0004 sekund(y).)

```
SELECT * FROM `sortedbookingsdesc`
```

☐ Profilowanie [ Edytuj w linii ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

☐ Show all | Liczba wierszy:  Filter rows:

Extra options

			booking_id	trip_id	customer_id	participants	price	created_at	updated_at
<input type="checkbox"/>	Edit	Copy	Delete	3	3	4	8 70392.00	2024-05-21 00:06:21	2024-05-21 00:06:21
<input type="checkbox"/>	Edit	Copy	Delete	19	3	3	3 26397.00	2024-05-21 15:07:31	2024-05-21 15:07:31
<input type="checkbox"/>	Edit	Copy	Delete	4	5	3	3 18297.00	2024-05-21 08:00:25	2024-05-21 08:00:25
<input type="checkbox"/>	Edit	Copy	Delete	20	3	3	2 17598.00	2024-05-21 18:16:49	2024-05-21 18:16:49
<input type="checkbox"/>	Edit	Copy	Delete	21	3	3	2 17598.00	2024-05-22 21:44:26	2024-05-22 21:44:26
<input type="checkbox"/>	Edit	Copy	Delete	22	4	6	3 11697.00	2024-05-22 21:51:01	2024-05-22 21:51:01
<input type="checkbox"/>	Edit	Copy	Delete	2	1	3	3 3270.00	2024-05-20 19:32:44	2024-05-20 19:32:44
<input type="checkbox"/>	Edit	Copy	Delete	1	1	3	2 2180.00	2024-05-20 18:21:42	2024-05-20 18:21:42

Widoki są wykorzystywane w panelu klienta do realizacji funkcjonalności sortowania wycieczek po cenie.

## **Podsumowanie**

Projekt pozwolił nam doskonalić w praktyce umiejętności z zakresu:

- technologii webowych: HTML, CSS, Bootstrap, JavaScript, PHP, Blade, Laravel;
- zarządzania bazami danych: MySQL, phpMyAdmin, CRUD, elementy języka proceduralnego;
- pracy z systemem kontroli wersji GIT;
- zarządzania projektem.