

Dokumentacja
Wojciech Typer 279730

Założenie:

Aplikacja desktopowa została stworzona z myślą o osobach podróżujących, które cenią sobie szybki i wygodny dostęp do kluczowych informacji o transporcie oraz zakwaterowaniu. Program oferuje interfejs, który pozwala użytkownikom (niebędącym administratorami) na łatwe wyszukiwanie połączeń transportowych między wybranymi miastami oraz przeglądanie dostępnych hoteli w danej lokalizacji.

Zalogowany użytkownik może:

- a) Znaleźć połączenie pomiędzy dwoma wskazanymi miastami - rodzaj środka transportu (samolot, pociąg, autobus), datę odjazdu i przyjazdu oraz cenę biletu.
- b) Wyszukać hotel we wskazanym mieście: jego nazwę oraz liczbę gwiazdek (od 1 do 5).

Baza Danych:

Na drugim nadesłanym pdf'ie znajduje się diagram bazy danych przedstawiający tabele oraz połączenia między nimi. Baza zawiera 6 tabel:

country:

- a) int country_id: primary_key, auto increment
- b) varchar name: not null, unique
- c) varchar code: not null, unique

city:

- a) int city_id: primary_key, auto increment
- b) varchar name: not null, unique
- c) int country_id: not null

hotel:

- a) int hotel_id: primary_key, auto increment
- b) varchar name: not null
- c) int city_id: not null
- d) int rating: not null (1-5)

transport:

- a) int transport_id: primary key, auto_increment
- b) varchar name: not null, unique

city_transport:

- a) int city_transport_id: primary key, auto increment
- b) int city_departure_id: not null
- c) int city_arrival_id: not null
- d) double price: not null
- e) datetime departure_time: not null
- f) datetime arrival_time: not null
- g) int transport_id: not null

user:

- a) int user_id: primary key, auto_increment
- b) varchar login: not null, unique
- c) varchar password: not null
- d) varchar status(admin, normal): not null

Połączenia pomiędzy tabelami:

- 1) country → city (1 do wielu) w tabeli city znajduje się country_id. Wiele miast może być przypisanych do tego samego państwa (Warszawa, Kraków, Wrocław, Jelenia Góra, Wilkanowo w Polsce)
- 2) city → hotel (1 do wielu) w tabeli hotel znajduje się city_id. Wiele hoteli może być przypisanych do jednego miasta
- 3) city → city_transport (wiele do wielu) w tabeli city_transport znajdują się zarówno city_departure_id i city_arrival_id, reprezentujące id miasta z którego środek transportu wyjeżdża i id miasta do którego dojeżdża
- 4) transport → city_transport (1 do wielu) w tabeli city_transport znajduje się transport_id. Jedno transport_id może być przypisane do wielu połączeń.

Zabezpieczenia przed sql injection:

W programie użyto prepared statement (params!, za pomocą biblioteki mysql).

Zamiast surowego konkatenowania stringów, prepared statements separują kod SQL od wartości przekazywanych przez użytkownika. Silnik bazy danych interpretuje je jako surowe dane, a nie część zapytania sql.

Klucze podstawowe, kandydackie i obce:

Tabela	klucz podstawowy	klucze kandydackie	klucze obce
country	country_id	name, code	brak
city	city_id	name	country_id
hotel	hotel_id	brak	city_id
city_transport	city_transport_id	brak	city_departure_id, city_arrival_id, transport_id
transport	transport_id	name	brak
user	user_id	login	brak

Normalizacja:

1NF:

- W bazie danych wszystkie atrybuty są atomowe
- Każda tabela ma klucz główny (PRIMARY KEY)
- Brak powtarzających się grup w żadnej tabeli

2NF:

- Wszystkie atrybuty niekluczowe są w pełni funkcyjnie zależne od klucza: Dla tabel z kluczem jednoatrybutowym nie istnieje ryzyko zależności częściowych, ponieważ nie ma kluczy częściowych.

3NF:

- Brak zależności tranzytywnych (kolumna nie będąca kluczem nie może zależeć od innej kolumny nie będącej kluczem):
 - w tabeli hotel name, rating, city_id zależy od hotel_id
 - w tabeli city_transport wszystkie kolumny są zależne od city_transport_id
 - w tabeli user status zależy od user_id, a nie od login ani password
 - w tabeli city kolumny są zależne od city_id
 - tabele transport i country przechowują unikalne dane

Prawa dostępu różnych użytkowników:

- a) **viewer:** Użytkownik viewer ma prawo dostępu do selectowania informacji ze wszystkich tabel. Dodatkowo może dodawać do tabeli user, co jest potrzebne podczas zakładania nowego konta.
- b) **admin:** Użytkownik admin ma prawa do selectowania, dodawania, usuwania i update'owania wszystkich tabel w bazie danych. Te uprawnienia są mu potrzebne do jego funkcjonalności w programie.

Rola adminów:

Program pozwala użytkownikowi zalogowanemu jako admin na:

- a) usuwanie/dodawanie nowych państw
- b) usuwanie/dodawanie nowych miast
- c) usuwanie/dodawanie nowych hoteli
- d) dodawanie nowych połączeń pomiędzy miastami
- e) zmienianie statusu użytkownika zalogowanego jako normal na admin

Procedury:

Obecnie w bazie danych znajduje się jedna procedura, która dodaje od 12.02.2025 do 19.02.2025 codziennie pociąg z Jeleniej Góry do Wrocławia o ustalonej godzinie przyjazdu, odjazdu i cenie. Docelowo, wszystkie przyszłe procedury w systemie powinny umożliwiać **automatyczne dodawanie cyklicznych połączeń** między dowolnymi miastami, zachowując te same parametry przez cały okres obowiązywania rozkładu. Przykładem inspiracji są tutaj **Koleje Dolnośląskie**, których połączenia charakteryzują się powtarzalnością:

- a) ta sama cena na tym samym odcinku drogi
- b) ten sam czas odjazdu i przyjazdu dla wszystkich dni (teoretycznie...)

Triggery:

Obecnie w bazie danych znajduje się kilka triggerów:

- a) trigger sprawdzający czy cena transportu nie jest ujemna
- b) trigger usuwający powiązane transporty w przypadku usunięcia miasta
- c) trigger sprawdzający czy czas przyjazdu jest późniejszy niż czas odjazdu

W przyszłości wraz z rozwojem programu i bazy powinno się dodać więcej triggerów które bardziej zabezpieczają bazę danych. Dodatkowo można by było dodać trigger, który aktualizuje ceny transportu, jeśli średnia cena wzrośnie itd