Projekt Bazy Danych – Wyszukiwarka lotów WiadroAirlines

Adrian Krzemiński, Miłosz Sabadach

Wymagania projektowe:  
  
Baza danych realizująca funkcjonalność przechowywania rekordów wraz z operacjami INSERT, UPDATE i DELETE, zapewniająca unikalność wprowadzanych danych (klucze, ograniczenia pól) oraz umożliwiająca wykonanie transakcji, po których zostanie zachowana spójność bazy. Przykład: system sprzedaży biletów do kina, biblioteka, księgarnia, carrental, bikesharing, scootersharing, itp.  
  
Wymagania systemowe  
a. Bazę danych o nazwie DB\_PROJ\_NAZWISKO1\_NAZWISKO2 tworzymy w psql (PostgreSQL).  
b. Obsługa bazy danych następuje przez interfejs www, napisany w dowolnym języku programowania.  
c. Stworzony "system" nie działa w oparciu o konieczność wcześniejszej znajomości schema w celu wykonia dowolnej operacji.  
  
Wymagania merytoryczne:  
a. Grupa wykonuje projekt samodzielnie.  
b. Członek grupy potrafi opisać koncepcję oraz sposób działania swojego "podsystemu" (baza danych, interfejs www)  
c. Projekt podlega ocenie podczas prezentacji prowadzącemu zajęcia.  
d. Grupa oddaje również projekt w postaci elektronicznej (PDF) przesłanej na adres prowadzącego. Plik powinien zawierać: opis organizacji bazy danych (schema) oraz kody źródłowe (dump + kod interfejsu)

Baza danych wzoruje się na aplikacji Fantasy Premier League – lidze w której użytkownicy mogą

tworzyć drużyny złożone z zawodników grających w realnych meczach i zdobywających punkty,

które są dodawane na konto użytkownika. Jeżeli administrator bazy wykona update w tabeli

total\_points spowoduje to automatyczny update punktow w drużynach użytkownika.

Poniższy schemat przedstawia tabele playersdb z zaimportowanymi zawodnikami grającymi w

Premier League, oraz dwie drużyny użytkowników, które użytkownik sam populuje wybierając

zawodników z playersdb z ograniczonym budżetem.

Ograniczenie budżetu zapewniamy przy pomocy funkcji sprawdzającej sume w tabeli cost

wykonywanej przy dodawaniu nowego zawodnika do tabeli użytkownika.

Interfejs został zbudowany przy użyciu technologii:

node.js

ejs

bootstrap

Przykładowe komendy psql użyte przy budowaniu bazy:

copy playersdb (id, name, team, position, cost, total\_points) from

'C:/Users/bbryk/Desktop/BDPROJ/players.csv' delimiter ',' CSV HEADER;

create table playersdb (id integer UNIQUE, name varchar(20), team varchar(3),position

varchar(3), cost decimal, total\_points integer, primary key (id, total\_points));

create table user2 (id integer UNIQUE, name varchar(20) primary key, team varchar(3),position

varchar(3), cost decimal, total\_points integer, FOREIGN KEY (id) references playersdb(id) on

update cascade);

create function check\_budget\_user2() returns trigger as '

Begin

if NEW.cost +(select sum(cost) from user2)>100 then raise notice ''Budget exceeded'';

return null; else return new;

end if;

end; 'Language 'plpgsql';

create trigger budget\_trigger\_user2 before insert on user2 for each row execute procedure

check\_budget\_user2();

insert into user1 select \* from playersdb where name='David Luiz';

update playersdb set total\_points=20 where id=21;