

Bazy danych 1

Dokumentacja projektu semestralnego
Agnieszka Lasek

Uruchomienie projektu:

Aby uruchomić projekt lokalnie należy z poziomu konsoli (w przypadku Windows wystarczy wywołać polecenie w WSL) wywołać polecenia:

```
npm install --save express ejs pg body-parser dev nodemon multer  
dotenv  
npm run devStart
```

Powyższe polecenia powinny być uruchomione w lokalizacji głównego folderu (folder o nazwie projekt_sklep). Pierwsze polecenie służy do zainstalowania wszystkich paczek, wymaganych do uruchomienia projektu. Natomiast drugie polecenie przy poprawnym zainstalowaniu wszystkich pakietów powinno uruchomić server i wyświetlić na konsoli informację o tym, że aplikacja została uruchomiona na porcie 8500 (w razie potrzeby port można zmienić w pliku .env). Po poprawnym wykonaniu powyższych instrukcji aplikacja powinna działać pod adresem <http://localhost:8500/>.

Projekt koncepcji, założenia

Zdefiniowanie tematu projektu:

Projekt zakłada stworzenie bazy danych dla sklepu internetowego oraz graficznego interfejsu użytkownika, pozwalającego mu na komunikację z bazą w czasie rzeczywistym i przeglądanie jej zawartości.

Analiza wymagań użytkownika:

Projekt zakłada możliwość dodawania przez użytkownika nowych danych do każdej tabeli, edytowania oraz usuwania danych z tabeli produktów oraz przeglądania utworzonych widoków.

Zaprojektowanie funkcji:

- Wykorzystanie w bazie widoków zostało zrealizowane w celu uproszczenia zapytań samego programisty m.in. przy zwracaniu informacji takich jak: pełna informacja o najdroższych zamówieniach, zestawienie listy kategorii wraz z liczbą przyporządkowanych do nich produktów czy ranking kategorii według średniej ceny produktów należących do każdej kategorii
- Wykorzystanie kluczy obcych zostanie zaprezentowane na diagramie ERD przedstawionym w tej dokumentacji
- Wykorzystanie funkcji agregujących takich jak licznik COUNT() czy średnia AVG(), które znalazły zastosowanie np. w rankingu kategorii według średniej ceny należących do nich produktów czy w przedstawieniu łącznej liczby produktów należących do każdej kategorii
- Przegląd danych i wprowadzanie danych do tabel, a także wyprowadzanie informacji w postaci raportów zostało zrealizowane poprzez wyświetlanie tabel na poszczególnych podstronach aplikacji i utworzenie do każdej z nich formularza pozwalającego wprowadzać dane
- Aplikacja po stronie bazy danych i interfejsu użytkownika została napisana w języku polskim
- Widok zawierający klauzulę GROUP BY i HAVING został zaprezentowany w zakładce "Ranking kategorii wg cen"
- Walidacja danych wprowadzanych w formularzach przeprowadzana jest częściowo po stronie bazy danych za pomocą operacji TRIGGER i częściowo po stronie aplikacji przez język javascript

Projekt diagramów (konceptualny)

Zdefiniowanie encji(obiektów) oraz ich atrybutów:

Producent: id_producent - klucz główny, nazwa_producent, email_producent

Produkt: id_produkt - klucz główny, nazwa_produkt, cena_produkt, id_producent

Kategoria: id_kategoria - klucz główny, nazwa_kategoria, opis_kategoria

Zamowienie: id_zamowienie - klucz główny, id_klient, id_kurier, koszt_zamowienia, data_zlozenia_zamowienia, data_dostarczenia, adres

Opinie: id_opinie - klucz główny, id_produkt, id_klient, data, ocena, opinia_tresc

Kurier: id_kurier - nazwa_firmy, koszt_dostawy, czas_dostawy

Klient: id_klient - klucz główny, email_klient, haslo

Zaprojektowanie relacji pomiędzy encjami:

Logika bazy danych została zaprezentowana za pomocą diagramu ERD zamieszczonego poniżej.

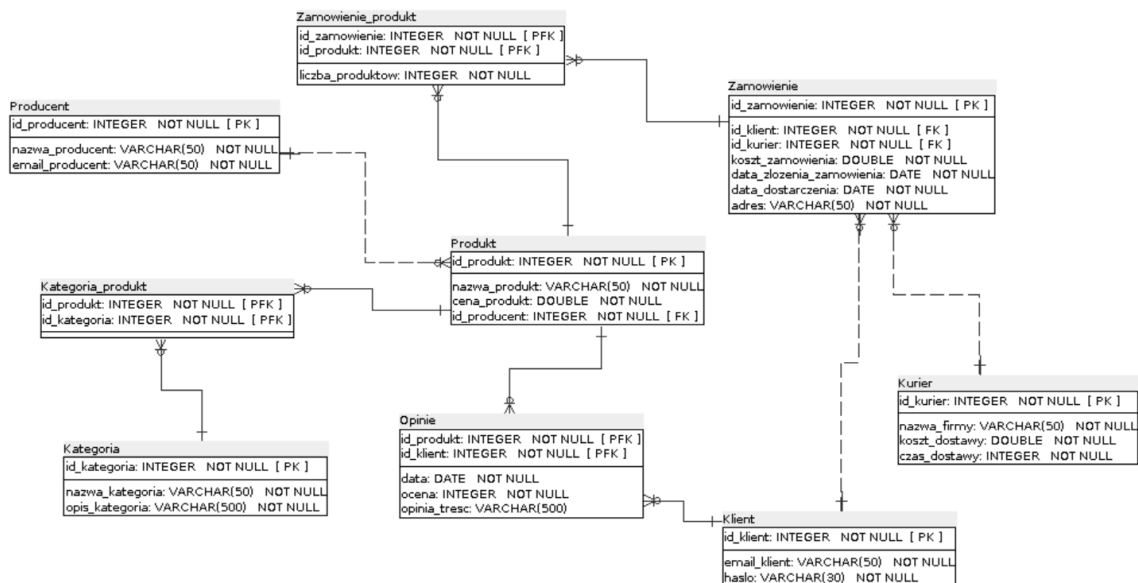


Diagram 1. Diagram ERD bazy projektu.

Projekt logiczny

Projektowanie tabel, kluczy, indeksów:

- Tabela **Producent** określa producentów produktów znajdujących się w bazie. Jeden producent może być producentem wielu towarów, ale jeden produkt jest produkowany tylko przez jednego producenta.
- Tabela **Produkt** określa produkt znajdujący się w bazie, który może być dodany do zamówienia. Do każdego produktu można wystawić opinię. Każdy produkt może być przyporządkowany do wielu kategorii, które określają jego zastosowanie lub cechy.
- Tabela **Kategoria** określa dostępne kategorie, do których mogą zostać przyporządkowane poszczególne produkty. Każda kategoria ma krótki opis, który ją definiuje. Pozostaje ona w relacji m:n z tabelą **Produkt**, ponieważ do jednej kategorii może należeć wiele produktów, natomiast jeden produkt może mieć wiele kategorii. Z tego powodu tabele te są ze sobą połączone tabelą asocjacyjną **Kategoria_produkty**.

- Tabela **Zamowienie** zawiera wszystkie potrzebne informacje na temat złożonych zamówień. Jest ona połączona z tabelą **Produkt** za pomocą tabeli asocjacyjnej **Zamowienie_produkt**, w której przechowywana jest lista wszystkich produktów zamówionych w każdym zamówieniu oraz informacja o ich liczbie.
- Tabela **Opinie** przechowuje opinie klientów na temat poszczególnych produktów, z uwzględnieniem oceny produktu oraz daty wystawienia opinii.
- Tabela **Klient** zawiera listę klientów zarejestrowanych w bazie sklepu. Zamówienia mogą być przypisywane jedynie do zapisanego w bazie klienta.
- Tabela **Kurier** zawiera listę dostępnych firm kurierskich, które możemy wybrać do zrealizowania zamówienia. Każda firma ma określoną opłatę wymaganą do zrealizowania zamówienia i przewidywany czas dostawy określony w dniach.

Słowniki danych:

Producent:

1. id_producent - klucz główny, int, not null
2. nazwa_producent - varchar, max 50 znaków, not null
3. email_producent - varchar, max 50 znaków, not null

Produkt:

1. id_produkt - klucz główny, int, not null
2. nazwa_produkt - varchar, max 50 znaków, not null
3. cena_produkt - double, not null
4. id_producent - klucz obcy tabeli **Producent**

Kategoria:

1. id_kategoria - klucz główny, int, not null
2. nazwa_kategoria - varchar, max 50 znaków, not null
3. opis_kategoria - varchar, max 500 znaków, not null

Kategoria_produkt:

1. id_produkt - klucz obcy tabeli **Produkt**

2. id_kategoria - klucz obcy tabeli **Kategoria**

Klient:

1. id_klient - klucz główny, int, not null
2. email_klient - varchar, max 50 znaków, not null
3. haslo - varchar, max 30 znaków, not null

Opinie:

1. id_produkt - klucz obcy tabeli **Produkt**
2. id_klient - klucz obcy tabeli **Klient**
3. data - date, not null
4. ocena - int, not null
5. opinia_tresc - varchar, max 500 znaków

Kurier:

1. id_kurier - klucz główny, int, not null
2. nazwa_firmy - varchar, max 50 znaków, not null
3. koszt_dostawy - double, not null
4. czas_dostawy - int, not null

Zamowienie:

1. id_zamowienie - klucz główny, int, not null
2. id_klient - klucz obcy tabeli **Klient**
3. id_kurier - klucz obcy tabeli **Kurier**
4. koszt_zamowienia - double, not null
5. data_zlozenia_zamowienia - date, not null
6. data_dostarczenia - date, not null
7. adres - varchar, max 50 znaków, not null

Zamowienie_produkt:

1. id_zamowienie - klucz obcy tabeli **Zamowienie**
2. id_produkt - klucz obcy tabeli **Produkt**
3. liczba_produkow - int, not null

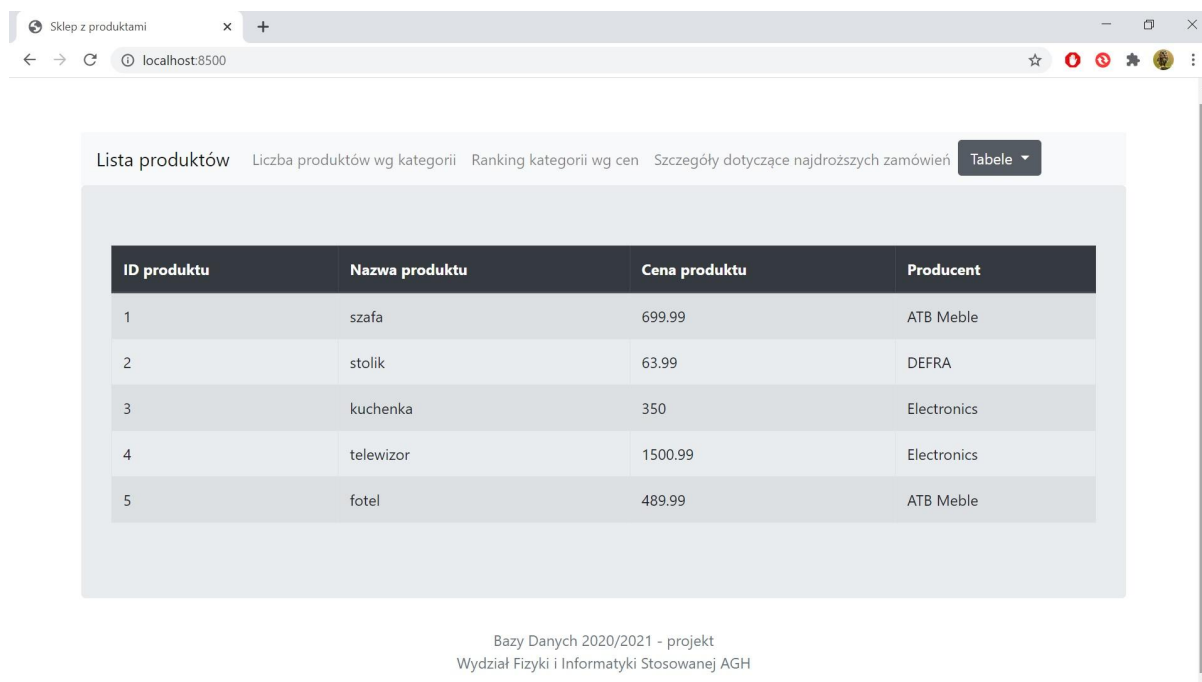
Sprawdzenie, czy tabele spełniają 1., 2., 3., postać normalną:

- I postać normalna jest spełniona - dane przechowywane w tabelach są atomowe (każde pole przechowuje jedną informację). Każda encja posiada klucz główny identyfikujący jednoznacznie dany wiersz, który jednoznacznie opisuje pojedynczy obiekt.
- Każda tabela przechowuje dane dotyczące tylko konkretnej klasy obiektów i nie powoduje redundancji danych. Prowadzi to do stwierdzenia, że II postać normalna również jest spełniona.
- Fakt, że żadna kolumna informacyjna nie należąca do klucza nie zależy też od innej kolumny informacyjnej, nie należącej do klucza świadczy o spełnieniu trzeciej postaci normalnej.

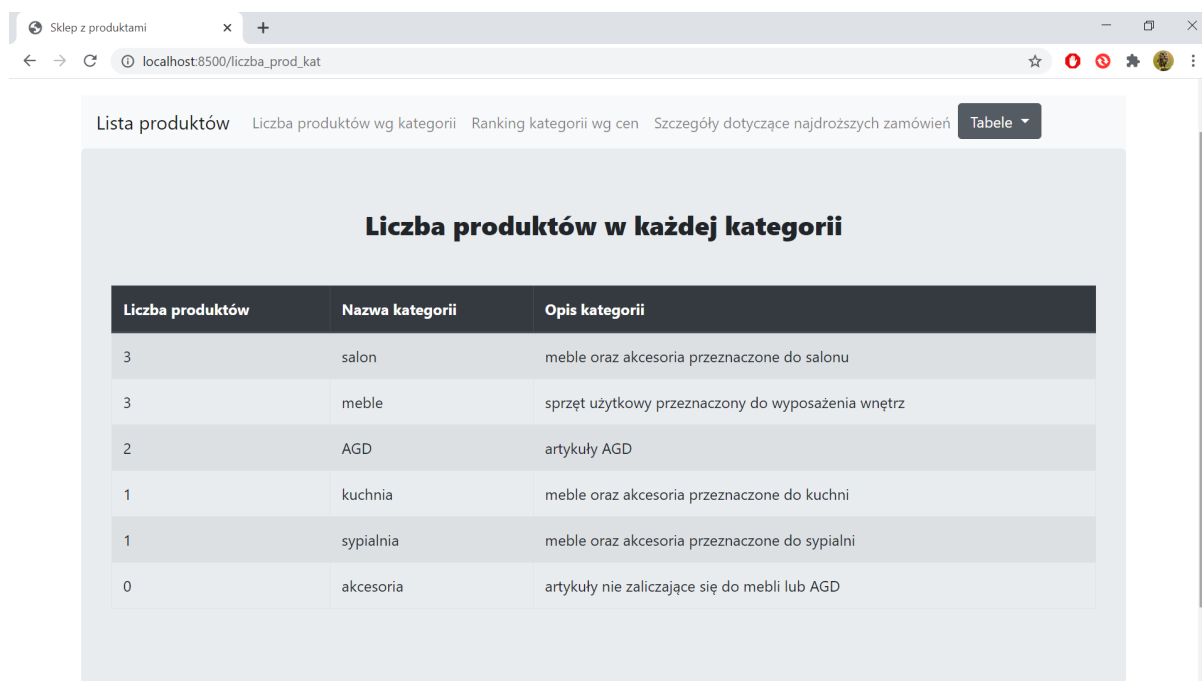
Projekt funkcjonalny

W tym podpunkcie została zaprezentowana szata graficzna i dostępne funkcjonalności projektu.

Aby przemieszczać się po aplikacji wystarczy w menu głównym, wyświetlającym się na pasku u góry ekranu wybierać tabele lub widoki, które chcemy obejrzeć lub edytować. W celu przetestowania funkcji usuwania i edycji produktów należy najpierw utworzyć nowy produkt, aby na nim wywoływać wyżej wymienione funkcje (aby zachować odpowiednie dane w innych tabelach i mieć możliwość przejrzania wszystkich uwzględnionych w bazie danych).



Obraz 1. Na obrazie zaprezentowana została strona startowa projektu - przedstawiająca listę produktów oraz ich producentów.



Obraz 2. Zawartość widoku prezentującego liczbę produktów w każdej kategorii.

Sklep z produktami

localhost:8500/ranking_cen

Lista produktów Liczba produktów wg kategorii Ranking kategorii wg cen Szczegóły dotyczące najdroższych zamówień Tabele

Ranking kategorii według średniej ceny produktów, które do niej należą

Dotyczy kategorii, do których należą co najmniej dwa produkty
Ranking jest ułożony od kategorii z najwyższą średnią ceną produktów do kategorii z najniższą średnią ceną produktów

Miejsce w rankingu	Nazwa kategorii	Opis kategorii	Średnia cena produktów w kategorii
1	AGD	artykuły AGD	925.50
2	salon	meble oraz akcesoria przeznaczone do salonu	684.99
3	meble	sprzęt użytkowy przeznaczony do wyposażenia wnętrz	417.99

Bazy Danych 2020/2021 - projekt
Wydział Fizyki i Informatyki Stosowanej AGH

Obraz 3. Widok prezentujący ranking kategorii według średniej ceny produktów.

Sklep z produktami

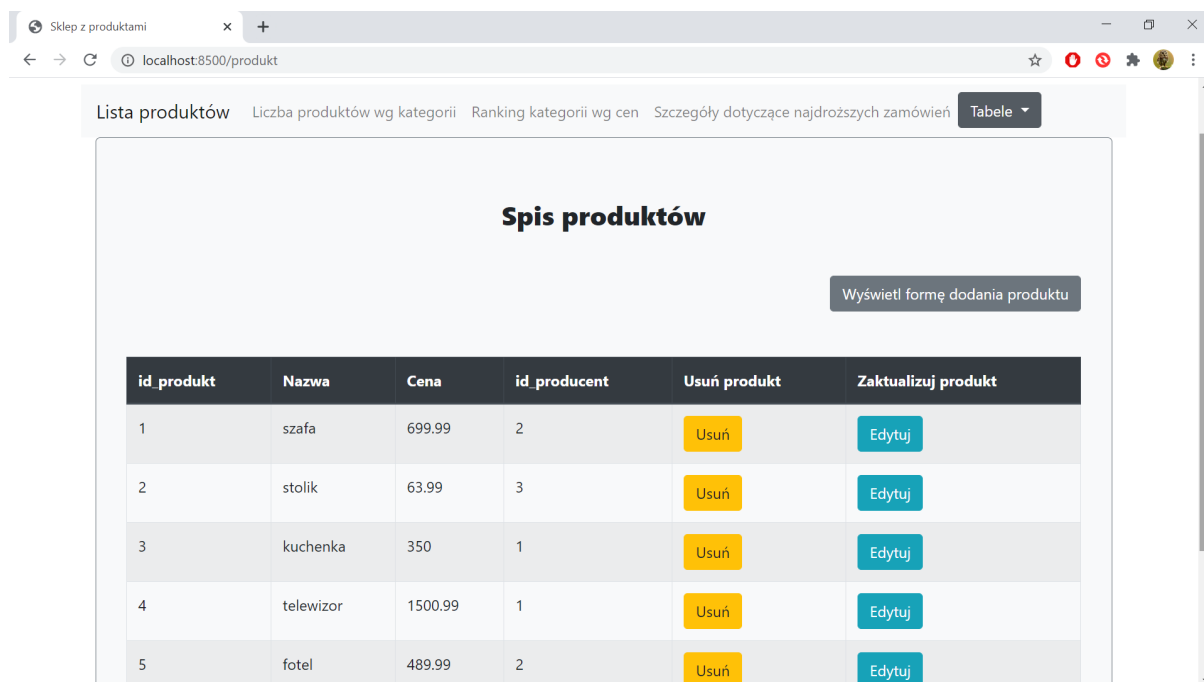
localhost:8500/zamowienia_info

Lista produktów Liczba produktów wg kategorii Ranking kategorii wg cen Szczegóły dotyczące najdroższych zamówień Tabele

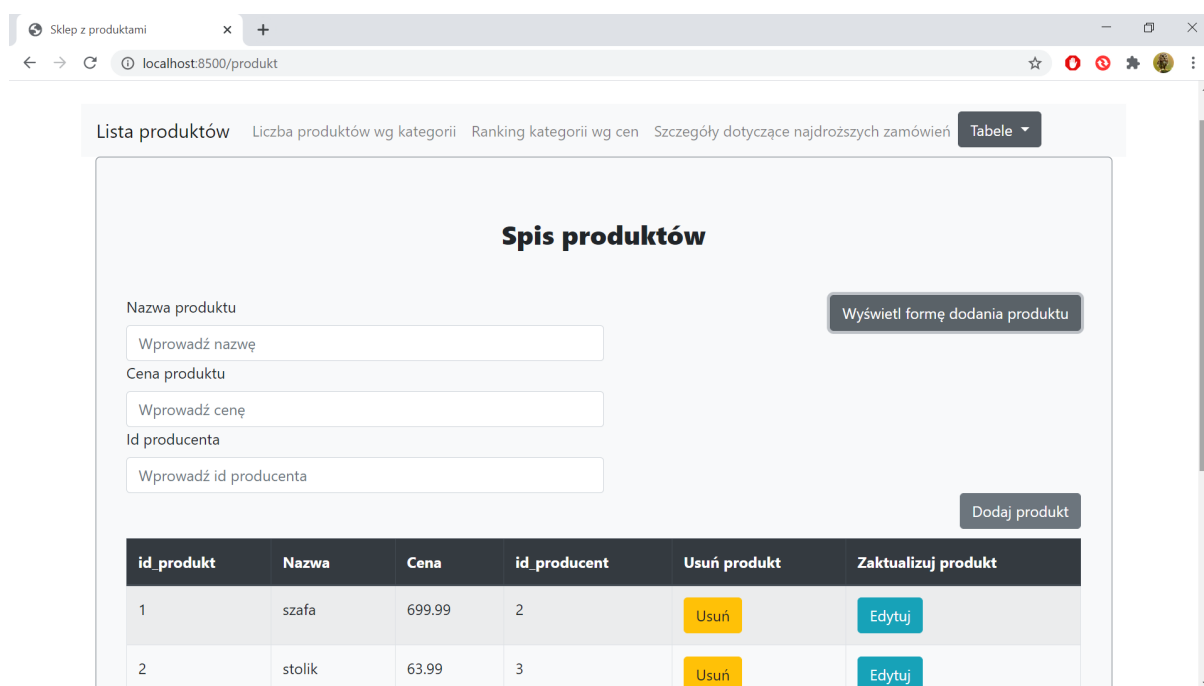
Szczegóły dotyczące zamówień, których łączny koszt wyniósł więcej niż 1000.00 zł

ID zamówienia	Liczba zamówionych produktów	Nazwa zamówionego produktu	Email klienta	Łączny koszt zamówienia	Data złożenia zamówienia	Data dostarczenia zamówienia	Nazwa firmy kurierskiej dowożącej zamówienie
1	3	fotel	basia98@gmail.com	1477.97	Tue Feb 02 2021 00:00:00 GMT+0100 (CET)	Sun Feb 07 2021 00:00:00 GMT+0100 (CET)	UPS
2	1	telewizor	nowak77@gmail.com	1516.49	Sun Jan 03 2021 00:00:00 GMT+0100 (CET)	Tue Jan 05 2021 00:00:00 GMT+0100 (CET)	DHL Express

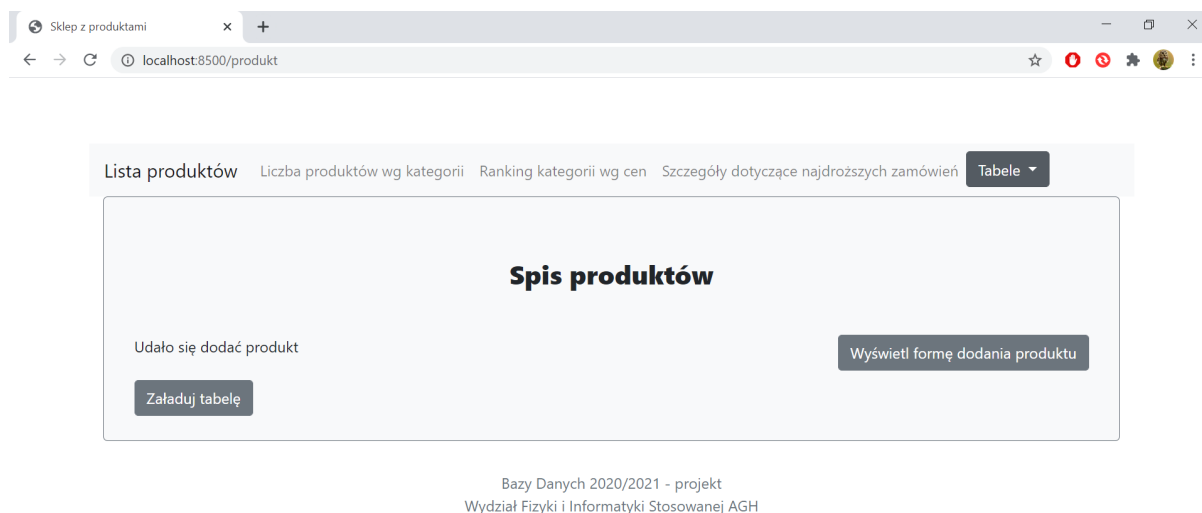
Obraz 4. Widok, w którym zaprezentowane są szczegóły najdroższych zamówień.



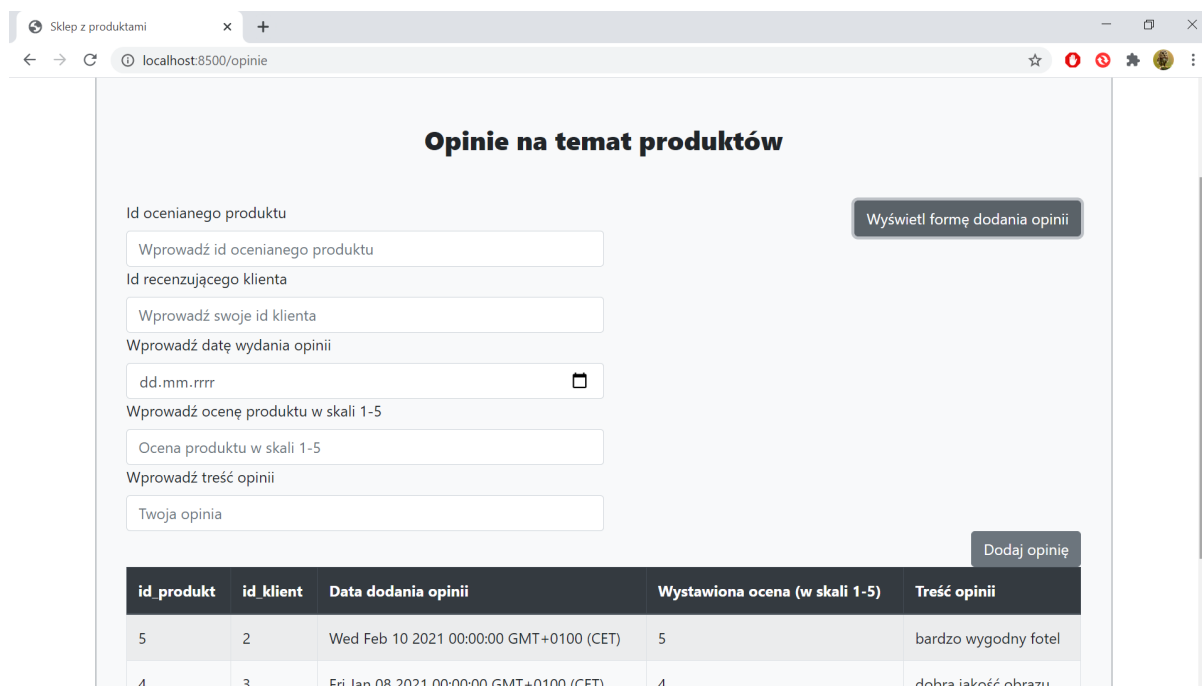
Obraz 5. Tabela produktów, do której możemy dodać produkt za pomocą formularza do tego przeznaczonego oraz edytować lub usunąć wybrany produkt korzystając z odpowiednich przycisków.



Obraz 6. Formularz służący do dodawania produktów.



Obraz 7. Wiadomość wyświetlająca się po poprawnym dodaniu produktu. Aby załadować tabelę z nowym produktem należy nacisnąć przycisk “Załaduj tabelę”.



Obraz 8. Formularz służący do dodawania opinii w tabeli opinia.

Dokumentacja

Wprowadzanie danych:

Dostępne jest w każdej tabeli, po wyświetleniu przypisanego jej formularza do wprowadzania nowych danych. Po uzupełnieniu i wysłaniu formularza dane zostaną wprowadzone do tabeli i pojawi się znak “Załaduj tabelę” w celu zaktualizowania wyświetlanych danych.

Krótką instrukcja obsługi aplikacji:

Jako instrukcję obsługi aplikacji należy traktować poprzedni punkt, prezentujący projekt funkcjonalny aplikacji.

Wykaz literatury:

<https://stackoverflow.com/>

<https://www.postgresql.org/docs/>

<https://www.w3schools.com/sql/>