

Dokumentacja projektu wykonywanego w ramach zajęć Bazy Danych I

Prasnal

15.11.2016

1 Projekt koncepcji, założenia

1.1 Zdefiniowanie tematu projektu

Temat projektu - "Aplikacja webowa do zarządzania wydatkami rodziny"
Celem aplikacji jest zgromadzenie i uporządkowanie wszystkich wydatków rodziny w jednym miejscu. Usprawni ona planowanie budżetu a jej głównym zadaniem będzie segregacja danych w zależności od tego, co interesuje użytkownika.

1.2 Analiza wymagań użytkownika

Użytkownik po dodaniu się w aplikacji do swojej rodziny, będzie mógł kontrolować jej wydatki. Będzie miał możliwość dodawania swoich wydatków oraz przeglądania wydatków dodanych przez innych członków rodziny. Strona umożliwi mu analizę danych, co w szybki i prosty sposób da mu możliwość sprawdzenia, kto i na co wydaje pieniądze (konkretne rzeczy lub też kategoria do jakiej dana rzecz należy: dom, ogród, jedzenie etc.). Po wybraniu odpowiedniego rodzaju wyszukiwania, dowie się

w jakich sklepach najczęściej rodzina robi zakupy lub kiedy został kupiony dany przedmiot. Aplikacja umożliwiać będzie sprawdzenie w szybki sposób czy gwarancja kupionego sprzętu jest jeszcze ważna, czy już wygasła.

1.3 Zaprojektowanie funkcji

W aplikacji będą funkcje odpowiedzialne za dodawanie nowych rekordów, ich edycję (dopisywanie, usuwanie, poprawianie), wyszukiwania danych oraz sortowania (wydatki w danym okresie czasu, od najwyższej/najniższej ceny, który z członków rodziny wydaje najwięcej, kiedy została kupiona dana rzecz, najczęściej odwiedzane sklepy etc.)

2 Projekt diagramów

2.0.1 Rodzina

Tabela rodzina przechowuje dane o rodzinach do których mogą należeć konkretni użytkownicy określani numerem ID, przechowuje również informacje o nazwisku oraz dane do logowania (Hasło, Login).

2.0.2 Użytkownik

Tabela użytkownik przechowuje dane na temat użytkowników: ich nr ID, ID rodziny do której należą, nazwisko, imię, mail, nick, hasło oraz informację czy dany użytkownik jest administratorem w rodzinie.

2.0.3 Zakupy

W zakupach przechowujemy informacje nt. który użytkownik zrobił dane zakupy, kiedy oraz ile za nie zapłacił. Każde zakupy posiadają inny numer ID.

2.0.4 Przedmiot

W tabeli prprzedmiot znajdziemy takie informacje jak nazwa, ilość i cena zakupionego przedmiotu oraz na jakich zakupach został on kupiony.

2.0.5 Gwarancje

Do tabeli gwarancje możemy dodać dowolny przedmiot i przypisać mu datę kupna oraz datę końca gwarancji.

2.0.6 Dochod

W tabeli dochód możemy zapisać zyski konkretnej rodziny i przez jakiego użytkownika zostały zdobyte.

2.0.7 Sklep

W tabeli sklep można umieścić które zakupy zostały w nim zrobione oraz dane sklepu: jego nazwę, miejscowość w której się znajduje, kod pocztowy, numer oraz ulicę.

2.0.8 Wydatki jednorazowe

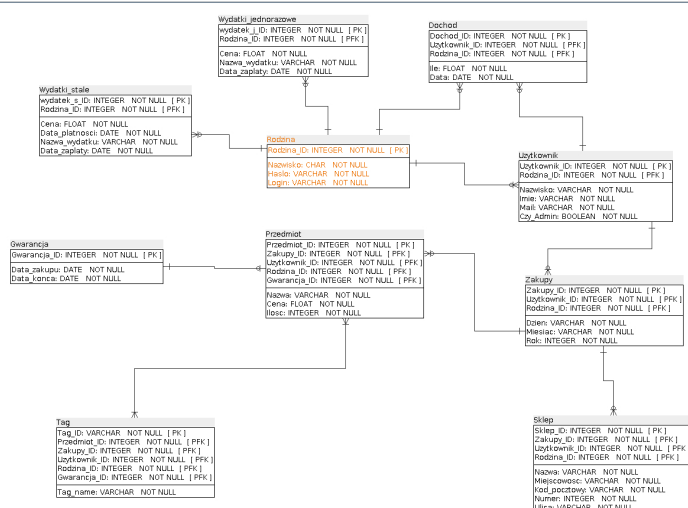
W wydatkach jednorazowych zamieszcza się wydatki dla konkretnej rodziny określanej przez Rodzina_{ID}, ich cenę oraz datę zapłaty

2.0.9 Wydatki stałe

W wydatkach stałych zamieszcza się wydatki dla konkretnej rodziny określanej przez Rodzina_{ID} ich cenę, tanę w której należało zapłacić dany rachunek oraz datę kiedy go zapłacono.

2.1 Relacje między encjami

Większość tabel jest powiązane między sobą relacją jeden do wielu, relacją jeden do jeden jest połączona wyłącznie tabela Przedmiot z tabelą Gwarancje.



Grafika przedstawiająca relacje między tabelami w pełnej rozdzielczości dołączona znajduje się w folderze z projektem.

3 Projekt logiczny

Większość poleceń SQL można znaleźć w pliku tabele.sql część jest jednak wywoływana bezpośrednio w pythonie i można znaleźć je w pliku func.py

Normalizacja tabel polega na zaprojektowaniu tabel w taki sposób, by nie występowała nadmiarowość tych samych danych. W bazie każda

relacja opisuje jeden obiekt a każda kolumna jest wartością skalarną (1NF), kolumny nie są zależne między sobą (2NF, 3NF)

4 Projekt funkcjonalny

Prezentacja i obsługa danych możliwa jest za pomocą strony internetowej na której dostępne są tabele. Dane od Rodziny o indeksie równym jeden prezentowane są na stronie "Analiza danych" w trzech tabelach: "Tabela Użytkowników", "Tabela Wydatków" oraz "Tabela Gwarancji". Dane w tych tabelach mogą być segregowane alfabetycznie (Od A/Z) lub po numerach (malejąco/rosnąco) poprzez kliknięcie w opis kolumny pojedynczo lub dwukrotnie. Wykres na stronie przedstawia również stosunek wydatków rodziny stałych do wydatków tymczasowych. Po lewej stronie strony widoczny jest panel umożliwiający do przejścia do konkretnych tabel oraz uzupełnienia ich nowymi danymi. Po przejściu na interesującą nas podstronę można wpisać dane i kliknąć przycisk "Dodaj" co sprawi, że dane zostaną dodane do bazy a aktualna tabela pojawi się na górze strony.

5 Dokumentacja

5.1 Wprowadzanie danych

Dane mogą zostać wprowadzone do bazy jedynie za pomocą formularzy dostępnych na podstronach.

5.2 Dokumentacja użytkownika

Aby włączyć aplikację należy mieć zainstalowany framework Flask oraz najnowszą wersję NodeJS (szczegóły instalacji dostępne w README.md)

w folderze frontend). Dla ułatwienia instalacji do projektu (w folderze głównym) dołączony jest plik requirements.txt. Z pliku tego za pomocą komendy: `pip install -r requirements.txt` można zainstalować brakującego flaska i ewentualne inne biblioteki. Zalecana jest jednak instalacja ręczna. Następnie możemy uruchomić projekt. W tym celu przechodzimy do folderu głównego z projektem i uruchamiamy `hello.py` poleceniem `"python hello.py"`, w innym terminalu należy wejść do folderu frontend i wpisać polecenie `"npm start"`. Po wykonaniu tych czynności w przeglądarce internetowej na stronie: <http://127.0.0.1:8080/> pojawi się aplikacja.

5.3 Dokumentacja techniczna

W pliku `test.py` znajduje się połączenie z bazą danych oraz funkcje z komentarzami opisującymi ich zastosowanie. Używane są one w pliku `test.py` by przerobić wynik wywołania na json i za pomocą flaska umieścić go na odpowiedniej podstronie. Dane te są następnie pobierane w poszczególnych plikach `*.jsx` i umieszczane za pomocą ReactJS w odpowiednim miejscu. We wszystkich plikach znajduje się opis działania poszczególnych elementów kodu.