

Architektura I - "GdzieJestLek"

Jakub Skrajny, Maciej Wojtala

18 maja 2020

1 Elementy systemu oraz ich komunikacja

System składa się z trzech elementów.

1. przeglądarka(klient)
2. serwer
3. baza danych

Przeglądarka oraz baza danych komunikują się jedynie z serwerem. Komunikacja zaczyna się po stronie klienta. Wysyłane jest żądanie do serwera. Serwer na podstawie żądania, serwer wysyła do bazy danych zapytanie SQL. Następnie serwer otrzymuje odpowiedź i przekazuje ją do klienta.

2 Przeglądarka

W polu body znajdują się następujące elementy:

- up
- search
- sample
- buts

Element up pełni formę nagłówka.

W elemencie search znajdują się:

- IN
- SBT

Definicje pól IN oraz SBT są opisane w dokumencie wymagań funkcjonalnych.

Element sample jest równoważny elementowi OUT z wymagań funkcjonalnych.

W elemencie buts znajdują się:

- L
- R

Definicja w dokumencie wymagań funkcjonalnych.

2.1 sample - OUT

Po odebraniu danych z serwera w postaci tablicy dwuwymiarowej, po stronie przeglądarki rozpoczyna się proces wyświetlenia wyniku.

Wykonywane są następujące funkcje:

- setParams
- rowSpanTabPrepare
- HTMLTabElemsPrepare
- disButs
- display

setParams - ustala zmienne globalne na odpowiednie wartości wyjściowe

rowSpanTabPrepare - tworzy tabele rowSpanTab, na podstawie której do komórek tabeli będzie przyporządkowana wartość rowspan(sposób łączenia komórek został opisany w wymaganiach funkcjonalnych)

HTMLTabElemsPrepare - tworzy tablicę tabel HTML(HTMLTabElems), każdy element tablicy jest elementem OBJ(definicja w wymaganiach funkcjonalnych), komórki tych tabel mają ustaloną wartość rowspan na podstawie rowSpanTabPrepare

disButs - zażąda atrybutem disabled przycisków L oraz R, przycisk L / R ma ustawiony atrybut disabled wtedy i tylko wtedy, gdy nie ma już wyników mniejszych/większych leksykograficznie od aktualnie wyświetlonych wyników.

display - tworzy nową tabelę HTML z elementów HTMLTabElems oraz ją wyświetla

3 Klient-serwer

Za pomocą przycisków SBT, L, R (opisanych w dokumencie pt. Analiza wymagań I - 'GdzieJestLek') klient wysyła żądanie do serwera.

Działanie po wciśnięciu SBT: Serwer inicjuje wyszukiwanie w bazie danych odpowiednich rekordów, które są zależne od wyrazu wpisanego w pole IN. Zwraca maksymalnie pierwsze 100 rekordów.

Działanie po wciśnięciu L / R:

Serwer na podstawie danych żądania zwraca maksymalnie 100 rekordów mniejszych/większych leksykograficznie z danych przetrzymanych przez serwer jako zmienna sesji.

3.1 Wysłanie żądanie

Wciśnięcie SBT powoduje wywołanie func(1). Wciśnięcie L / R powoduje wywołanie func(0). Parametr wskazuje na to, czy należy zaaktualizować wyszukiwany wyraz. 1 - wyszukanie po aktualnym wyrazie w polu IN, 0 - zmiana

wyświetlanej części wyniku.

Funkcja `func` sprawdza, czy dany wyraz jest poprawny (definicja z dokumentu jak wyżej). W sprawdzaniu pomaga funkcja `check()`, która sprawdza pojedynczy znak. Jeśli wyraz nie jest poprawny, to wyświetlany zostaje stosowny komunikat za pomocą funkcji `tell()`. Jeśli wyraz jest poprawny, to wysyłane jest żądanie do serwera za pomocą funkcji `jQuery.ajax()`. Żądanie jest typu `post`, za wykonanie tego żądania odpowiada skrypt `go.php`, serwer otrzymuje jako daną zawartość `IN`.

3.2 Odpowiedź na żądanie

Jeśli żądanie zakończy się sukcesem, serwer zwraca odpowiedź jako napis. Jest to JSON opisujący dwuwymiarową tablicę. Ta tablica zawiera rekordy zwrócone przez bazę danych w wyniku wyszukiwania. Jeśli żądanie nie zakończy się sukcesem, zostanie wyświetlony stosowny komunikat za pomocą funkcji `tell()`.

4 Struktura bazy danych

Baza danych jest zrealizowana w systemie Oracle (Oracle Database). Baza zawiera jedną tabelę o nazwie „Leki”. Tabela ta zawiera następujące kolumny: „Nazwa”, „Substancja_czynna”, „Postac”, „Dawka”, „Zawartosc_opakowania”, „Kod_EAN”, „Poziom_odplatnosci”, „Zakres_objetych_refundacja”, „Wysokosc_dopłaty”.

4.1 Skrypt tworzący bazę

```
$ create table Leki
$ (
$   Nazwa varchar2(2000) not null,
$   Substancja_czynna varchar2(2000) not null,
$   Postac varchar2(2000) not null,
$   Dawka varchar2(2000) not null,
$   Zawartosc_opakowania varchar2(2000) not null,
$   Kod_EAN varchar2(2000) not null,
$   Poziom_odplatnosci varchar2(2000) not null,
$   Zakres_objetych_refundacja varchar2(2000) not null,
$   Wysokosc_dopłaty numeric(19,4) not null
$ );
```

5 Serwer-baza

Po odebraniu danych od klienta, serwer łączy się z bazą danych za pomocą funkcji `oci_connect()`. W przypadku błędu zwracany jest komunikat o błędzie. Następnie serwer konstruuje zapytanie (opisane w podrozdziale), parsuje go za pomocą `oci_parse()` oraz wykonuje za pomocą `oci_execute()`. W przypadku błędu zwracany jest stosowny komunikat. Następnie do tablicy `tab` zapisywane są kolejne rekordy wyniku wyszukiwania (`oci_fetch_array()`). Na końcu przy pomocy funkcji `json_encode()` zwracana jest tablica dwuwymiarowa `tab`.

5.1 Zapytanie

```
$ select distinct Lek.Nazwa, Lek.Substancja_czynna, ...
$ from Lek inner join (
$   select distinct Nazwa, Substancja_czynna
$   from Lek
$   where Nazwa like '%".$str.%"'
$   or Substancja_czynna like '%".$str.%"'
$   or ...) h
$ on Lek.Nazwa = h.Nazwa $ and Lek.Substancja_czynna = h.Substancja_czynna
$ order by Lek.Nazwa, Lek.Substancja_czynna, ...
```