

SPDB

Projekt z SPDB, semestr 13Z, EiTI. Temat 2 - Wizualizacja danych udostępnionych w konkursie Predict which 311 issues are most important to citizens. Prowadzący dr inż. Grzegorz Protaziuk

Autorzy

- Michał Świątochowski
- Dariusz Dudziński

Wymagania serwera aplikacji

- Apache 2
- mod_rewrite (rewrite engine dla Apache2)
- PHP 5.4.*
- PHP5-OCI8 (moduł do połączeń z bazą danych Oracle)
- Composer (<https://getcomposer.org>) - menedżer zależności bibliotek dla PHP)

Wymagania serwera baz danych

- Oracle Database 11g (lub wyższe),
- Oracle Spatial (np. w wersji Enterprise)

Instalacja aplikacji

- Należy pobrać i rozpakować źródła do katalogu np. /var/www/spdb (`$HOME`)
- W katalogu `$HOME` należy uruchomić komendę `composer install` która zainstaluje potrzebne biblioteki
- Utworzyć Virtual Hosta w Apache2 skierowanego do folderu `$HOME/public`
- Utworzyć indywidualny klucz Google Maps API v3 (<https://code.google.com/apis/console>)
- Podmienić wygenerowany klucz w pliku `$HOME/module/Application/view/application/index/index.twig` zamiast ciągu znaków `###API_KEY###`
- Uzupełnić dane do połączenia z bazą danych w pliku konfiguracyjnym: `$HOME/config/autoload/global.php`

Instalacja bazy danych

- Rozpakować i zaimportować spakowane (*.tar.gz) pliki SQL z `$HOME/data/oracle`

Opis architektury i działania aplikacji

Od strony języka PHP najważniejszymi plikami są (w folderze `module/Application/src`):

- Application\Controller\IndexController - odpowiada za przekazanie parametrów POST wysyłanych AJAXem przez Javascript do repozytorium encji, aby uzyskać wyniki i zwrócić je (obiekty i tablice) w formacie JSON.
 - akcja `get-markers` pobiera markery dla danych parametrów wyświetlania (zoom, okno)
 - akcja `get-issue-types` pobiera typy (nazwy) zgłoszeń do pola combo z autocomplete
- Application\Entity* - encje reprezentujące zgłoszenia, stany i hrabstwa USA
- Application\Repository\Issue - repozytorium z metodami do pobierania danych z bazy danych
 - metoda `getMarkersForCountry` pobiera markery dla poziomu kraju
 - metoda `getMarkersForStates` pobiera markery dla poziomu stanu
 - metoda `getMarkersForCounties` pobiera markery dla poziomu hrabstwa

- metoda `getMarkersForLocalArea` pobiera markery dla poziomu lokalnego (widoczne okno przy największym zoomie)
- metody `*qb` zwracają wygenerowany kod SQL dla powyższych metod
- metoda `sdoRect` generuje reprezentację prostokąta o zadanych współrzędnych w formacie `SDO_GEOMETRY`
- metoda `getIssueTypes` pobiera typy (nazwy) zgłoszeń do pola combo z autocomplete

Od strony języka Javascript (w folderze `public/js`):

- main.js - odpowiada za:
 - inicjalizację Google Maps API v3
 - obsługę zdarzeń (events) GM API
 - obsługę komunikacji XHR (AJAX) z serwerem PHP
 - obsługę pozostałych elementów interfejsu użytkownika (pole combo, komunikaty)