Instrukcja obsługi

1. Aby zainstalować aplikację na serwerze potrzebujemy maszynę z zainstalowanym Apache HTTP Server, PHP oraz bazą danych MySQL. W praktyce posłużyć nam do tego może dowolny hosting internetowy, nawet darmowy.
   Wszystkie potrzebne pliki znajdują się w repozytorium GIT'a pod adresem https://github.com/klapaucius4/ETL-Process
   Możemy pliki te pobrać w formacie .zip, rozpakować a następnie poprzez klienta FTP (FileZilla, Total Commander) wgrywamy. Jeśli aplikację instalujemy na fizycznej maszynie (komputer, serwer) lub posiadamy dostęp SSH do naszego hostingu, możemy po prostu w wybranych katalogu wywołać komendę:
   git clone https://github.com/klapaucius4/ETL-Process
   Oczywiście zakładamy, że Apache, PHP i MySQL mamy na tej maszynie zainstalowane i skonfigurowane.
   Następnie tworzymy bazę danych. Na postawionej już stronce którą umieściliśmy na serwerze, znajduje się link do pliku .sql zawierającego strukturę danych tabeli (files/struktura_bazy_danych.sql). Możemy go zaimportować do naszej bazy danych, np. za pomocą phpMyAdmin lub jakiegokolwiek klienta bazodanowego. Możemy też za pomocą tych narzędzi ręcznie stworzyć stworzyć strukturę naszej bazy danych, gdyż tak naprawdę jest to tylko jedna tabela. Wystarczy użyć jednego zapytania MySQL:

```
-- Zrzut struktury tabela palkora_etl.job_adverts
CREATE TABLE IF NOT EXISTS `job_adverts` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `gumtree_id` varchar(255) NOT NULL,
  `title` varchar(255) NOT NULL,
  `date` varchar(255) NOT NULL,
  `location` varchar(255) NOT NULL,
  `by` varchar(255) NOT NULL,
  `type_of_work` varchar(255) NOT NULL,
  `type_of_contact` varchar(255) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=186 DEFAULT CHARSET=latin2;
```

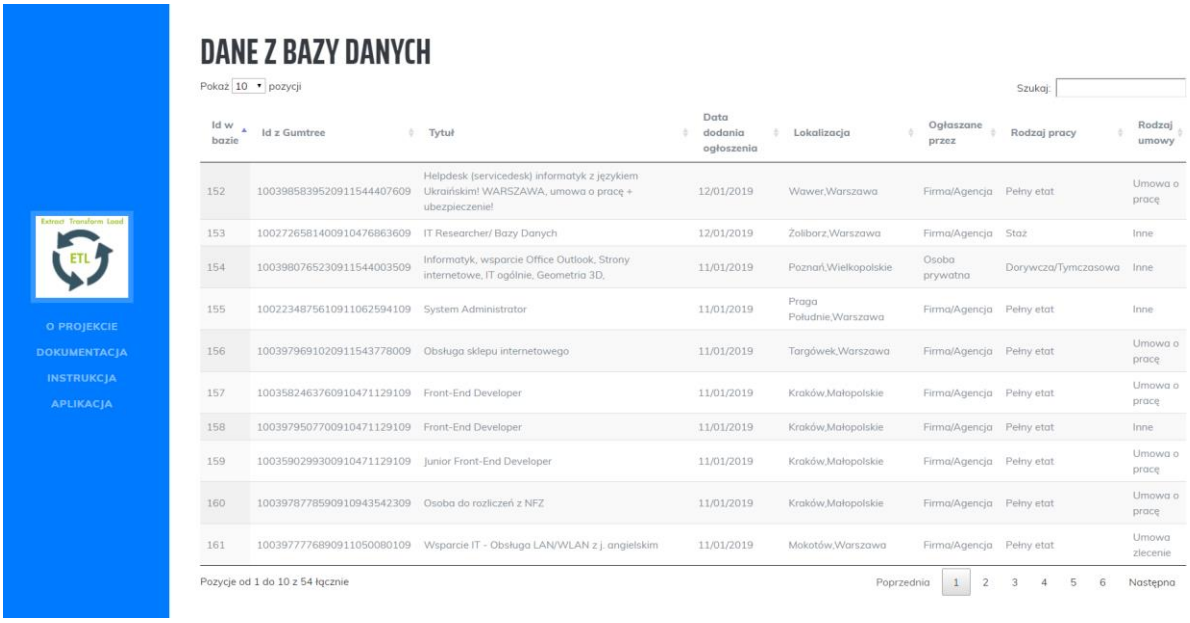
   Posiadając bazę, możemy w pliku config.php ustalić parametry połączeniowe z bazą danych, takie jak: host, nazwa bazy, nazwa użytkownika i hasło.
   Gdy to już zrobimy, mamy aplikację poprawnie skonfigurowaną na serwerze.
   Jako klient, posłuży nam już w tym wypadku dowolna przeglądarka internetowa.

2. Interfejs aplikacji jest intuicyjny i nie powinno być raczej większych problemów ze zrozumieniem działania aplikacji.
   W pierwszej, tytułowej sekcji mamy możliwość przejścia do repozytorium na Github'ie, pobrania strony tytułowej projektu lub pobrania struktury bazy danych w pliku .sql.
   W kolejnych sekcjach mamy możliwość pobrania dokumentacji technicznej i niniejszej instrukcji obsługi.
   Główny interfejs aplikacji znajdziemy po kliknięciu w menu na odnośnik „Aplikacja". Możemy wywoływać odpowiednie operacje za pomocą widocznych przycisków:

Oto ich funcjonalności:

- „Wykonaj proces ETL” – zostaje wykonany pełen proces ETL – trzy etapy – Extract, Transform, Load. Wykonane zostanie to samo, gdybyśmy wywoływali poniżej znajdują się 3 przyciski ręcznie. Przejdźmy więc do nich w kolejnych punktach.
- „E – extract” – na tym etapie wywoływana jest funkcja „extractStep()”.
  Na samym początku sprawdzamy w pliku etlStatus.json, na jakim ostatnio etapie skończyliśmy proces ETL. Jeśli literka w tym pliku to „L”, oznacza to, że był to proces Load, czyli zakończyliśmy cały proces ETL. W tej sytuacji możemy przejść dalej. W przeciwnym razie pojawi się komunikat o niepowodzeniu.
  Zadaniem funkcji extractStep() jest połączenie się za pomocą funkcji file_get_html z wybraną stroną (definiowaną wcześniej w pliku config.php) i pobranie całej treści html ze strony. Następnie za pomocą biblioteki „PHP Simple HTML DOM Parser” przekształcamy ten „tekst” do postaci DOM (Obiektowy model dokumentu). Biblioteka dostarcza nam wiele przydatnych funkcje, dzięki którym możemy swobodnie przemieszczać się po strukturze drzewa DOM i swobodnie pobierać wybrane przez nas elementy ze strony. Z jej pomocą tworzymy tablicę z ogłoszeniami z portalu Gumtree, w której dane są walidowane i sprawdzane, czy posiadają odpowiedni format itd. Oprócz tego usuwane są z nich zbędne znaki, takie jak tzw. „white spaces” czy też znaczniki HTML. Ostatecznie dane te zapisywanie są do pliku w formacie .csv o nazwie data_from_extract.csv. Na samym końcu w pliku etlStatus.json wstawiamy literkę „E” oraz obecną godzinę. Ma to za zadanie informować aplikację, na jakim etapie procesu ETL skończyliśmy.
- „T – transform” – Na tym etapie uruchamiamy fuynkcję „transformStep()”, która sprawdzi status procesu, a następnie, jeśli się okaże, że poprzednio wywołano proces Extract – przejdzie dalej. Na tym etapie otwarty zostaje plik data_from_extract.csv, z którego dane są pobrane do tablicy, przekształcone, a następnie zapisane do pliku data_from_transform.csv. Jednocześnie usunięty zostaje plik z poprzedniego procesu o nazwie data_from_exctract.csv. Na końcu oczywiście ustalamy status aplikacji w pliku etlStatus.json.

- „L - Load” – ostatni etap procesu ETL, czyli zapisanie danych do bazy. Sprawdzany jest status aplikacji, następnie dane są pobierane z pliku data_from_transform.csv, przekształcane na tablicę i następnie zapisywane do bazy danych MySQL za pomocą zapytania „INSERT”. Zapis jest możliwy wyłącznie wtedy, gdy w bazie danych nie znajduje się jeszcze ogłoszenie o danym ID ogłoszenia. Jeśli jest inaczej, ogłoszenie zostaje pominięte. Oczywiście o każdym takim przypadku użytkownik zostaje poinformowany.
  Na końcu następuje usunięcie pliku data_from_transform.csv i ustawienie statusu aplikacji na status „L”. Dzięki temu jesteśmy w stanie wykonać proces ETL od początku.
- „Zobacz dane w bazie” – opcja ta wyświetla nam listę wszystkich rekordów z bazy w tabeli. Dzięki bibliotece JavaScript – DataTable – mamy możliwość swobodnego sortowania danych, paginacji itd.
- „Pobierz dane w formacie CSV” – pobieramy całą zawartość rekordów z bazy danych do pliku w formacie .CSV.



- „Wyczyść bazę danych” – wyczyszcza wszystkie istniejące rekordy w bazie danych.

Dodatkowo należy zaznaczyć, że wykonując proces ETL w całości, jak i dzieląc go ręcznie na etapy, nie jesteśmy w stanie tych właśnie etapów wykonać w różnej kolejności. Zawsze jest to schemat Extract -> Transform -> Load. Przy próbie zmienienia kolejności otrzymamy komunikat o niepowodzeniu. Dzieje się tak za sprawą pliku etlStatus.json, który przechowuje informacje o ostatnio wykonywanym etapie procesu ETL.

Wszelkiego rodzaju „produkty uboczne” procesu ETL (takie jak pliki .csv generowane w etapach Extract i Transform) zostają automatycznie usuwane.

3. Przykładowym scenariuszem wykorzystania aplikacji może być chęć analizy ogłoszeń o pracę związanych z branżą IT – aby np. wiedzieć jakie stanowiska, technologie, miejsca zatrudnienia itd. są popularne.
   Można by też traktować aplikację jako pewne API, z którego moglibyśmy pobierać ogłoszenia do innych portali, aplikacji itd.