**Chomik - wyszukiwarka mieszkań**

**Spis treści**

1. Informacje o autorach
2. Opis działania
3. GUI
   * Wymagania dla aplikacji klienckiej Chomik
   * Uruchamianie aplikacji
   * Konfiguracja
   * Sposób użytkowania aplikacji klienckiej Chomik
   * Podstawowe błędy mogące wystąpić podczas działania aplikacji Chomik
4. PARSER
   * Składniki parsera
   * Uruchamianie parsera
   * Opis działania
   * Konfiguracja parsera
   * Tabele

**1. Informacje o autorach**

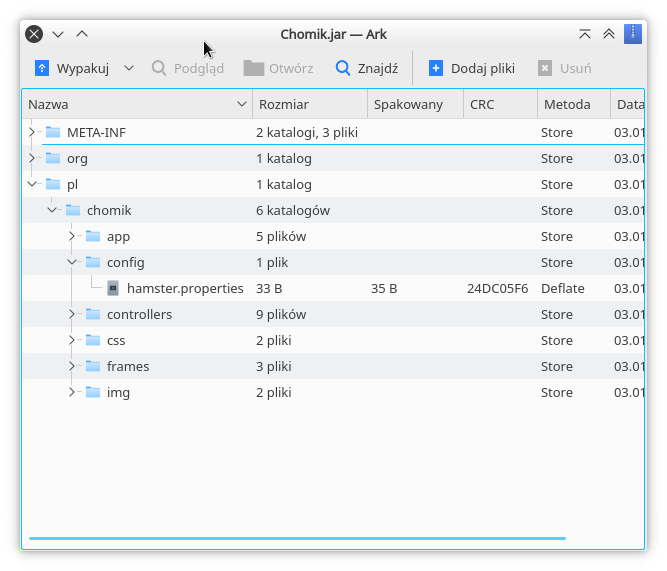
Nad projektem pracują: Borek Kamil i Kozubek Dariusz.

**2. Opis działania**

Projekt działa na zasadzie klient-serwer. Klientem jest aplikacja napisana w języku Java zawarta w katalogu GUI. Skrypt parsujący uruchomiany jest na serwerze. Aktualizuje dane co godzinę. Przeciętny czas aktualizacji danych wynosi ok. 5-10min. Klient łączy się za pomocą sieci Internet z serwerem pobierając dane o ogłoszeniach uwzględniając kryteria podane przez użytkownika. Po pobraniu informacji analizuje je i wyświetla umożliwiając użytkownikowi szybki przegląd ofert z danego serwisu.

**3. GUI**

* **Wymagania dla aplikacji klienckiej Chomik**
  1. Środowisko uruchomieniowe Java Runtime w wersji 8
  2. Połączenie internetowe
* **Uruchamianie aplikacji**
  1. Po pobraniu aplikacji (plik *Chomik.jar* zawarty w katalogu *GUI/Chomik/out/artifacts/Chomik\_jar*) uruchamiamy go za pomocą *Java Runtime Environment* (jest to domyślny program do obsługi plików *\*.jar* więc w większości przypadków wystarczy podwójnie kliknąć na ikonę aplikacji).
  2. Można również uruchomić aplikację z poziomu wiersza poleceń, aby otrzymać dodatkowe informacje, które aplikacja wysyła na konsolę. w tym celu należy:
     + Uruchomić *"Wiersz polecenia"* w systemie Windows lub dowolny terminal/konsolę w systemach Linux/macOS.
     + Przejść do katalogu, w którym znajduje się plik *"Chomik.jar"*.
     + Wykonać polecenie *"java -jar Chomik.jar"*.
     + Opcja ta przydatna jest szczególnie, gdy używamy trybu debugowania.
* **Konfiguracja**
* Aplikacja do normalnego działania nie wymaga żadnej konfiguracji. Gdyby jednak nastąpił problem z połączeniem lub wystąpił by błąd użytkownik może zmienić domyślną konfigurację otwierając plik *Chomik.jar* w dowolnym programie do obsługi plików *\*.zip, \*.rar* itp. W katalogu *'pl/chomik/config'* znajduje się plik *'hamster.properties'*. Jego zawartość jest ładowana do aplikacji jako konfiguracja. Otwierając go dowolnym edytorem tekstowym możemy zmienić następujące opcje:
  + Adres serwera parsującego projektu Chomik.
  + Stan trybu debugowania (zaleca się używanie go tylko w celu wykrycia błędu).



Rysunek . Zawartość pakietu Chomik.jar

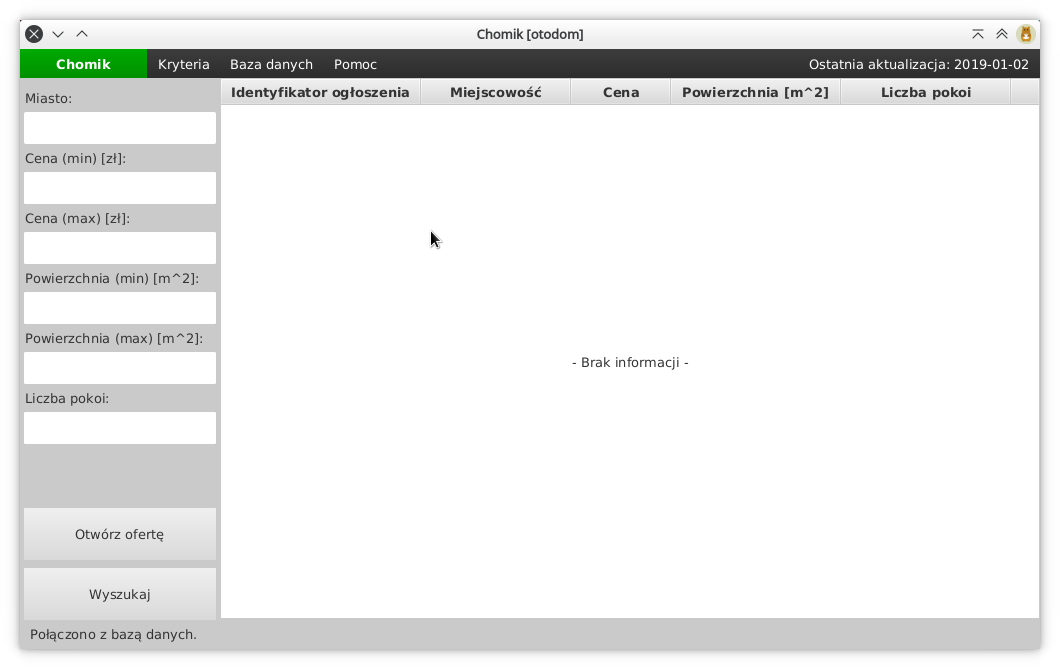
Prawidłowa zawartość pliku *'hamster.properties':*

*ip=<adres\_serwera>*

*debugmode=<stan\_trybu\_debugowania>*

Opis opcji:

* ip -> adres serwera zewnętrznego na którym działa oprogramowanie parsera
* debugmode -> umożliwia włączenie/wyłączenie trybu debugowania, zalecany tylko do wykrywanie błędów, testowania oraz do uzyskania szczegółowych informacji na temat działań wykonywanych przez Chomika (możliwe wartości to:
  + *true* -> włącz tryb debugowania
  + *false* -> wyłącz tryb debugowania
* **Sposób użytkowania aplikacji klienckiej Chomik**
* Po uruchomieniu aplikacji wyświetlone zostanie okno jak na przykładzie poniżej
* U góry okna znajduje się menu, w którym znajdziemy opcje takie jak:
  + *Rozpocznij nową sesję* - Odnawia połączenie, czyści krysteria
  + *Zamknij Chomika* - zamyka aplikację kliencką Chomik
  + *Wyczyść kryteria* - czyści kryteria wprowdzone przez użytkownika
  + *Odśwież połączenie z bazą dancyh* - łączy ponownie z serwerem danych
  + *O projekcie ...* - pokazuje krótką informację o Chomiku
* Użytkownik może wypełnić następujące kryteria (w dowolnej konfiguracji i ilości):
  + *Miasto* - miasto dla szukanego mieszkania
  + *Cena (min)[zł]* - cena minimalna w zł
  + *Cena (max)[zł]* - cena maksymalna w zł
  + *Powierzchnia (min)[m^2]* - najmniejsza powierzchnia szukanego mieszkania
  + *Powierzchnia (max)[m^2]* - maksymalna powierzchnia szukanego mieszkania
  + *Liczba pokoi*
* Po wpisaniu kryteriów, klikając w *'Wyszukaj'* po chwili otrzymamy listę ofert
* Po zaznaczeniu interesującej nas oferty możemy użyć przycisku *'Otwórz ofertę* lub dwukrotnie kliknąć na nią, aby otworzyć okno szczegółów oferty.
* Znajdziemy tam podstawowe informacje oraz link do oferty, który możemy skopiować lub otworzyć bezpośrednio ofertę w domyślnej przeglądarce używając przycisku *'Otwórz w przeglądarce'*.



Rysunek . Interface aplikacji klienckiej Chomik

* **Podstawowe błędy mogące wystąpić podczas działąnia aplikacji klienckiej Chomik**
* *'Musisz zaznaczyć ogłoszenie, aby je otworzyć'*
* *'Pole może przyjmować tylko litery'* - aktualnie wypełniane pole kryterium nie przyjmuje wprowadzonego znaku
* *'Pole może przyjmować tylko wartości numeryczne'* - aktualnie wypełniane pole kryterium nie przyjmuje wprowadzonego znaku
* *'Nie można sprawdzić daty ostatniej aktualizacji'* - nie można połączyć z bazą danych, sprawdź konfigurację

**4. PARSER**

* Obecna wersja parsera obsługuje bazy danych serwera *PostgreSQL*. Obsługa serwisów ogłoszeniowych została rozdzielona do osobnych plików by umożliwić wybór poszczególnych serwisów, co powoduje konieczność oddzielnego uruchamiania pojedynczych skryptów. Czas działania każdego ze skryptów został ustawiony na 12 minut.  
  Skrypt został wyposażony w:
  + mechanizm cyklicznego usuwania rekordów starszych niż 1 miesiąc
  + zapis ostatnio pobranych danych do pliku
  + zapis błędów do pliku *error\_log*
* Uruchomienie parsera.  
  By uruchomić parser należy posiadać dostęp do serwara umożliwiającego wykonywanie skryptów *php* oraz dostępu do serwera *postgresql* z odpowiednią strukturą bazodanową. Samo uruchomienie skryptu możliwe jest przez otworzeniu lokalnego adresu w przeglądarce lub wywołanie z konsoli poprzez "*php -parametr nazwa\_skryptu.php*" (np: *php -q curl-otodom.php*). Zalecamy utworzenie automatycznego wywoływania skryptu, dla serwera z systemem linux z użyciem np. podstawowego narzędzia jakim jest *cron*. Dodając do harmonogramu cron'a wpis w postaci (*0 \*/1 \* \* \* php -q /usr/share/nginx/php/curl-otodom.php*) ustalamy, iż skrypt będzie uruchamiany w 0-wej minucie każdej godziny czasu serwera.
* Opis działania  
  Skrypt obsługujący serwis *otodom* pobiera m.in. następujące dane: id\_ogłoszenia, miasto, cenę, ilość pokoi; a następnie zapisuje je do odpowiednich kolumn danej tabeli w naszej bazie.
* Konfiguracja parsera (oraz serwera).  
  + Wszelkie opcje serwera wymagane do poprawnego działania parsera zostały podane w pliku *"dbSampleAndServerSettings.txt"*, jak również umieszczone zostały definicje wymaganych tabel bazy danych.
  + Konfiguracja polega na edycji pliku *"pgsqlbdconfig.php"* gdzie w odpowiednich polach podajemy dane potrzebne do połączenia z serwerem sql oraz odpowiednie nazwy tabel i kolumn - wszelkie informacje zostały podane w plikach jako komentarze.
* Tabela główna posiada kolumny które przechowują dane:
  + *#kolumna nie jest używana w konfiguracji#* - id ogłoszenia w naszej bazie, jest to Primary\_key, auto\_increment
  + *tab1col1* - znakowe id ogłoszenia w serwisie
  + *tab1col2* - liczbowe id ogłoszenia w serwisie
  + *tab1col3* - nazwa miasta
  + *tab1col4* - cena dla danego ogłoszenia
  + *tab1col5* - wielkość oferowanej nieruchomości
  + *tab1col6* - liczba pokoi
  + *tab1col7* - adres ogłoszenia
  + *tab1col8* - wstawiana jest zawsze aktualna data i czas dodania ogłoszenia do naszej bazy
* Tabela dla przechowywania daty ostatniej aktualizacji tabeli głównej
  + *datatabcol1* - id dla danego serwisu
  + *datatabcol2* - nazwa serwisu aktualizowanego
  + *datatabcol3* - czas ostatniej aktualizacji.