

otoDom WebAPI 2.0
dokumentacja usługi sieciowej serwisu otodom.pl

wersja 110920



Spis treści

1. Wstęp.....	5
2. Historia zmian dokumentacji i usługi.....	5
3. Usługa dla biur nieruchomości i deweloperów - zarządzanie ogłoszeniami i inwestycjami.....	9
3.1. Dostęp do usługi.....	9
3.2. Dodawanie/edycja ogłoszeń.....	9
3.3. Dodawanie/edycja zdjęć w ogłoszeniach.....	9
3.4. Aktywacja/przedłużanie ogłoszeń.....	10
3.5. Aktywacja/przedłużanie promocji w ogłoszeniach.....	10
3.6. Ustalanie geolokalizacji.....	10
3.7. Dezaktywacja ogłoszeń.....	10
3.8. Usuwanie ogłoszeń.....	11
3.9. Dodawanie/edycja inwestycji.....	11
3.10. Dodawanie/edycja zdjęć w inwestycji.....	11
3.11. Usuwanie inwestycji.....	11
3.12. Przykłady.....	11
3.12.1. Logowanie.....	12
3.12.2. Pobieranie słowników.....	12
3.12.3. Dodawanie/edycja ogłoszenia.....	13
4. Usługa dla partnerów - transfer ogłoszeń.....	16
4.1. Dostęp do usługi.....	16
4.2. Mechanizm transferu.....	16
4.3. Przykłady.....	16
4.3.1. Logowanie.....	17
4.3.2. Pobieranie słowników.....	17
4.3.3. Pobieranie transferowanych ogłoszeń.....	18
5. Obiekty i pola.....	19
5.1. Lista pól wymaganych przy edycji.....	19
5.2. Ograniczenia dotyczące wybranych pól.....	19
5.3. Lista pól wykluczonych przy edycji.....	19
5.4. Lista pól.....	20
5.4.1. InsertionObject.....	20
5.4.2. SellerInfoObject.....	21
5.4.3. FlatDetailsObject.....	21
5.4.4. HouseDetailsObject.....	22
5.4.5. TerrainDetailsObject.....	22
5.4.6. RoomDetailsObject.....	23
5.4.7. CommercialPropertyDetailsObject.....	23
5.4.8. HallDetailsObject.....	24
5.4.9. GarageDetailsObject.....	24
5.4.10. PhotoObject.....	25
5.4.11. ContactInfoObject.....	25
5.4.12. TransferUnitObject.....	25
5.4.13. InvestmentObject.....	25
5.4.14. AdditionalContactObject.....	26
5.4.15. AdditionalAddressObject.....	26
5.4.16. PhoneNumberObject.....	26
6. Słowniki.....	26
6.1. Aktualizacja słowników.....	26

6.2. Pola ogólne.....	27
6.2.1. Country.....	27
6.2.2. Province.....	27
6.2.3. District.....	28
6.2.4. PriceCurrency.....	28
6.2.5. Status.....	28
6.2.6. PriceType.....	28
6.2.7. MarketType.....	28
6.2.8. ObjectName.....	29
6.2.9. OfferType.....	29
6.3. Pola obiektu FlatDetails.....	29
6.3.1. BuildingType.....	29
6.3.2. BuildingMaterial.....	29
6.3.3. BulidingFloorsNum.....	29
6.3.4. BuildingOwnership.....	30
6.3.5. FloorNo.....	30
6.3.6. ExtrasMask.....	30
6.3.7. Heating.....	30
6.3.8. ConstructionStatus.....	31
6.3.9. WindowsType.....	31
6.3.10. SecurityMask.....	31
6.3.11. MediaMask.....	31
6.3.12. EquipmentMask.....	31
6.4. Pola obiektu HouseDetails.....	32
6.4.1. BuildingType.....	32
6.4.2. BuildingMaterial.....	32
6.4.3. ConstructionStatus.....	32
6.4.4. RoofType.....	32
6.4.5. Type.....	32
6.4.6. FloorsNum.....	33
6.4.7. GarretType.....	33
6.4.8. WindowsType.....	33
6.4.9. MediaMask.....	33
6.4.10. HeatingMask.....	33
6.4.11. FenceMask.....	34
6.4.12. Location.....	34
6.4.13. ExtrasMask.....	34
6.4.14. SecurityMask.....	34
6.5. Pola obiektu TerrainDetails.....	34
6.5.1. Type.....	34
6.5.2. MediaMask.....	35
6.5.3. AccessMask.....	35
6.5.4. VicinityMask.....	35
6.6. Pola obiektu RoomDetails.....	35
6.6.1. PersonsMask.....	35
6.6.2. Availability.....	36
6.6.3. BuildingType.....	36
6.6.4. MediaMask.....	36
6.6.5. EquipmentMask.....	36
6.6.6. PreferredSex.....	36

6.6.7. PreferredProffesion.....	36
6.7. Pola obiektu CommercialPropertyDetails.....	37
6.7.1. PropertyUseMask.....	37
6.7.2. ExtrasMask.....	37
6.7.3. Floor.....	37
6.7.4. BuildingType.....	37
6.7.5. ConstructionStatus.....	38
6.7.6. MediaMask.....	38
6.7.7. SecurityMask.....	38
6.8. Pola obiektu GarageDetails.....	38
6.8.1. Localization.....	38
6.8.2. Structure.....	38
6.9. Pola obiektu HallDetails.....	39
6.9.1. AccessMask.....	39
6.9.2. Structure.....	39
6.9.3. MediaMask.....	39
6.9.4. UseMask.....	39
6.9.5. ParkingType.....	40
6.9.6. Flooring.....	40
6.9.7. ConstructionStatus.....	40
6.9.8. SecurityMask.....	40
6.9.9. Fence.....	40
6.10. Pola obiektu Investment.....	40
6.10.1. State.....	40

1. Wstęp

Serwis otoDom.pl udostępnia usługę sieciową (web service) opartą na protokole SOAP, służącą do komunikacji z serwisem i realizacji usług takich jak transfer ogłoszeń do portali partnerskich oraz zamieszczanie i edycja ogłoszeń. Usługa zdefiniowana jest za pomocą języka WSDL i może być łatwo implementowana niezależnie od wykorzystywanej platformy po stronie klienta.

W dalszej części dokumentu usługa nazwana jest **otoDom WebAPI 2.0**.

Dostęp do otoDom WebAPI 2.0 w wersji testowej
oraz opis wykorzystywanych metod i typów danych:

<http://beta.otodom.pl/webapi2/>

specyfikacja WSDL:

<http://beta.otodom.pl/webapi2/?wsdl>

Dostęp do otoDom WebAPI 2.0 w wersji produkcyjnej
oraz opis wykorzystywanych metod i typów danych:

<http://webapi2.otodom.pl/>

specyfikacja WSDL:

<http://webapi2.otodom.pl/?wsdl>

2. Historia zmian dokumentacji i usługi

110920

- dodanie pól w obiekcie HouseDetails: VicinityMask, AccessMask, SecurityMask
- dodanie do pola HouseDetails:HeatingMask wartości kominkowe oraz piece kaflowe
- dodanie do pola HouseDetails:ExtrasMask wartości klimatyzacja oraz umeblowanie
- dodanie pól w obiekcie FlatDetails: ConstructionStatus, WindowsType, SecurityMask, MediaMask, EquipmentMask, RentToStudents, Deposit, DepositCurrency
- dodanie wartości w polu FlatDetails:Heating: kotłownia
- dodanie do pola FlatDetails:ExtrasMask wartości klimatyzacja oraz tylko dla niepalących
- dodanie pól w obiekcie CommercialPropertyDetails: BuildingType, TerrainArea, ConstructionStatus, BuildYear, MediaMask, SecurityMask
- dodanie do CommercialPropertyDetails:ExtrasMask wartości: dojazd asfaltowy, winda, klimatyzacja, ogrzewanie
- dodanie do HallDetails pól: ConstructionStatus, OfficeSpace, SocialFacilities, Ramp, SecurityMask, Fence
- dodanie do HallDetails:AccessMask wartości: asfalt

- dodanie do `HallDetails:MediaMask` wartości: gaz, internet, szambo, oczyszczalnia
- dodanie do `TerrainDetails` pól: Location
- dodanie do `TerrainDetails:VicinityMask` wartości: góry, morze
- dodanie do `RoomDetails` pól: RoomPlace, RoomsNum, Rent, RentCurrency, Deposit, DepositCurrency, Balcony, MediaMask, EquipmentMask, WomenInFlat, MenInFlat, WomenInRoom, MenInRoom, PreferredSex, PreferredProffesion, NonSmokersOnly

110616

- możliwość ustawienia geolokalizacji dla ogłoszeń

110418

- możliwość aktywacji promocji dla ogłoszeń

110329

- poprawienie numeracji;

100601

- dodanie nowych nieobowiązkowych pól tylko do odczytu: Latitude, Longitude, AllegroId, ActivationDate

100107

- skasowanie pól Fence, Type, Heating, Height, Lighting, Heating, Structure z listy pól obowiązkowych

100104

- poprawienie błędów w przykładach: dodanie pola `ObjectName` oraz usunięcie pola `Object Type` w obiekcie `InsertionObject`;

091208

Poprawienie błędów w przykładach:

- zmiana nazwy pola `ObjectName` na ID w obiekcie `InsertionObject` (przykład 1.10.3);
- rozszerzenie obiektów `InsertionObject`, `FlatDetailsObject` o brakujące pola (przykład 1.10.3);
- opakowanie obiektu `InsertionObject` w `SoapVar` (przykład 1.10.3) – zabezpiecza to przed występowaniem błędu:

```
PHP Fatal error: SOAP-ERROR: Encoding: object hasn't 'SomeProperty' property.
```

w przypadku gdy definicja obiektu ulegnie zmianie po stronie webserwisu.

091120

- dodanie pola `Title` do obiektu `InsertionObject`;

090929

- poprawienie błędu w dokumentacji dla pola: `Furnished`;
- dodanie w specyfikacji formatu pola: `FreeFrom`;

090902

- dodanie do FlatDetails:BuildingType wartości loft
- dodanie do FlatDetails pól: Heating, FreeFrom, Furnished, Rent, RentCurrnecy, PriceIncludeRent;
- dodanie do HouseDetails pól: Roofing, FreeFrom, Furnished, Rent, RentCurrnecy, PriceIncludeRent;
- dodanie do CommercialPropertyDetails pól: FreeFrom, Furnished;
- dodanie do TerrainDetails:Type wartości rolno-budowlana;
- dodanie do Rooms pola: FreeFrom;

090828

- dodanie do FlatDetails:BuildingMaterial wartości: silikat oraz beton komórkowy

090112

- dodanie do FlatDetails:BuildingType wartości apartamentowiec
- dodanie do HouseDetails:BuildingType wartości gospodarstwo
- dodanie do TerrainDetails:Type wartości siedliskowa
- dodanie do FlatDetails pól BuildingMaterial oraz BuildingYear
- dodanie do HouseDetails pola BuildingMaterial

080716

- dodanie do FlatDetails:BuildingType wartości szeregowiec
- dodanie do FlatDetails:ExtrasMask wartości ogródek

080703

- rozszerzenie funkcjonalności usługi o możliwość zarządzania ogłoszeniami oraz inwestycjami przez biura nieruchomości i deweloperów
- nowe pola w obiekcie InsertionObject - AdditionalContact typu integer oraz AdditionalAddress typu integer
- nowe pole w obiekcie InvestmentObject - Name typu string
- likwidacja pola Keywords w obiekcie InsertionObject
- nowy obiekt AdditionalContactObject reprezentujący dodatkowe dane kontaktowe
- nowy obiekt AdditionalAddressObject reprezentujący dodatkowe dane adresowe
- nowy obiekt PhoneNumberObject reprezentujący numer telefonu
- nowa metoda getValidCityNames zwracająca listę prawidłowych nazw miejscowości
- nowa metoda getValidQuarterNames zwracająca listę prawidłowych nazw dzielnic

080610

- zmiana wartości w polach: FlatDetails:FloorNo oraz CommercialPropertyDetails:Floor

080529

- wsparcie transferu ogłoszeń inwestycji
- nowy obiekt InvestmentObject reprezentujący ogłoszenie inwestycji
- nowe pola w obiekcie InsertionObject - Investment typu integer oraz LocationDescription typu string
- nowe pole w obiekcie TransferUnitObject - Investment typu InvestmentObject
- zmiana typu pól RoomsNum oraz Height ze słownikowego na całkowitoliczbowy
- aktualizacja słowników

3. Usługa dla biur nieruchomości i deweloperów - zarządzanie ogłoszeniami i inwestycjami

3.1. Dostęp do usługi

Aby uzyskać dostęp do interfejsu otoDom WebAPI niezbędne jest posiadanie własnego klucza dostępowego oraz aktywne konto biura nieruchomości lub dewelopera w serwisie otoDom. Konto testowe należy zarejestrować na serwerze <http://beta.otodom.pl>. W celu aktywacji konta i otrzymania klucza dostępu należy skontaktować się z Działem Technicznym otoDom telefonicznie lub wysyłając w tej sprawie email na adres: tech@otodom.pl.

3.2. Dodawanie/edycja ogłoszeń

W celu dodawania lub edycji ogłoszenia należy w pierwszej kolejności zalogować się wywołując metodę `doLogin()`. Wywołanie metody zwróci identyfikator sesji – `sessionKey`. Identyfikator sesji będzie wykorzystany w wywołaniach innych metod np. `doInsertionEdit()` pozwalającej dodawać/edytować ogłoszenia.

Kolejnym krokiem jest przygotowanie obiektu reprezentującego pojedyncze ogłoszenie (typ `InsertionObject`). Po przygotowaniu obiektu można już wywołać metodę `doInsertionEdit()` przekazując do niej następujące parametry: identyfikator sesji oraz obiekt ogłoszenia. Każde wywołanie `doInsertionEdit()` zwraca identyfikator ogłoszenia w serwisie otoDom lub też informacje o ewentualnych błędach.

Metoda `doInsertionEdit()` pozwala zarówno dodawać oraz edytować ogłoszenie. Jeśli w obiekcie ogłoszenia nie będzie podany atrybut `ID` wtedy zostanie dodane nowe ogłoszenie natomiast gdy atrybut ten zostanie podany wtedy odbędzie się edycja ogłoszenia i będzie możliwa korekta danych ogłoszenia.

Aby uprościć i przyspieszyć dodawanie/edycję ogłoszeń metodę `doInsertionEdit()` należy wywoływać wielokrotnie w ramach jednej sesji podstawiając do kolejnych wywołań kolejne ogłoszenia.

Dodanie nowego ogłoszenia powoduje wygenerowanie wystawienia wliczającego się do abonamentu za usługę.

Po dodaniu ogłoszenie jest automatycznie aktywowane.

3.3. Dodawanie/edycja zdjęć w ogłoszeniach

Zdjęcia należy dodawać i edytować przy użyciu dedykowanych metod `doInsertionPhotoEdit`, `doInsertionPhotoRemove`. W celu edycji istniejącego zdjęcia konieczne jest przekazanie aktualnej jego pozycji. Przy dodawaniu nowych zdjęć pola pozycji nie należy przekazywać. Pola `Url` i `UrlMini` zawierają wyłącznie informację zwrotną i są ignorowane w procesie edycji.

3.4. Aktywacja/przedłużanie ogłoszeń

Aktywacja ogłoszenia nieaktywnego lub przedłużanie okresu aktywności ogłoszenia aktywnego realizowane jest przez metodę `doInsertionActivate(string sessionId, int insertionId[, int periodCount])`. Jako parametry podajemy klucz sesji, obiekt ogłoszenia, w którym należy wypełnić wyłącznie pole `ID` lub `RemoteId` oraz opcjonalnie liczbę okresów, na które ma zostać przedłużone ogłoszenie.

Aktywacja/przedłużenie ogłoszenia powoduje wygenerowanie wystawienia wliczającego się do abonamentu za usługę.

Możliwe jest wystawienie ogłoszenia na maksymalnie cztery okresy. Rzeczywista data wygaśnięcia ogłoszenia zwracana jest jako wynik metody aktywującej.

3.5. Aktywacja/przedłużanie promocji w ogłoszeniach

Aktywacja bądź przedłużenie promocji realizowane jest z użyciem metody `doPromotionActivate($sessionId, InsertionObject $insertionObject, $promotion, $periodCount = 1)`. Jako parametry podajemy klucz sesji, obiekt ogłoszenia, w którym należy wypełnić wyłącznie pole `ID` lub `RemoteId`, typ promocji oraz opcjonalnie liczbę okresów, na które ma zostać aktywowana/przedłużona promocja.

Aktywacja/przedłużenie promocji powoduje wygenerowanie opłat doliczanych proporcjonalnie do jednej lub wielu kolejnych faktur w zależności od ilości okresów promocji.

Obsługiwane są następujące typy promocji:

- PRO1 - podświetlenie kolorem,
- PRO2 - początek listy wyników,
- PRO3 - strona główna,
- PRO123 - pakiet „3 w 1” – podświetlenie kolorem, początek listy wyników, strona główna

Możliwe jest aktywowanie promocji maksymalnie na taki okres na jaki aktywne jest ogłoszenie. Rzeczywista liczba doliczonych okresów jest zwracana jako wynik metody aktywującej.

3.6. Ustalanie geolokalizacji

Ustalenie geolokalizacji ogłoszenia możliwe jest z użyciem dedykowanej metody `WebAPI` – `doInsertionGeoMarkerEdit`. Zmiany dokonywane bezpośrednio na polach `Latitude` i `Longitude` są ignorowane – pola te służą wyłącznie do odczytu informacji. Lokalizacje ustawione z użyciem `WebAPI` traktowane są jako precyzyjne i prezentowane w ogłoszeniu z dużym poziomem zbliżenia

3.7. Dezaktywacja ogłoszeń

Dezaktywacja ogłoszenia jest realizowana przez metodę `doInsertionInactivate()`. Jako

parametry podajemy klucz sesji oraz obiekt ogłoszenia, w którym należy wypełnić wyłącznie pole `ID` lub `RemoteId`.

Ogłoszenie dezaktywowane można później aktywować.

3.8. Usuwanie ogłoszeń

Usuwanie ogłoszenia jest realizowane przez metodę `doInsertionRemove()`. Jako parametry podajemy klucz sesji oraz obiekt ogłoszenia, w którym należy wypełnić wyłącznie pole `ID` lub `RemoteId`.

Ogłoszenie jest usuwane nieodwracalnie.

3.9. Dodawanie/edycja inwestycji

Zarządzanie inwestycjami dostępne jest wyłącznie dla deweloperów. Proces dodawania/edycji inwestycji jest analogiczny do procesu [dodawania/edycji ogłoszeń](#). Różnica polega na wywołaniu metody dedykowanej do zarządzania inwestycjami – `doInvestmentEdit()`.

Dodanie nowej inwestycji powoduje wygenerowanie wystawienia wliczającego się do abonamentu za usługę.

Dodanie inwestycji powoduje jej automatyczną publikację w serwisie.

3.10. Dodawanie/edycja zdjęć w inwestycji

Zdjęcia należy dodawać i edytować przy użyciu dedykowanych metod `doInvestmentPhotoEdit`, `doInvestmentPhotoRemove`. W celu edycji istniejącego zdjęcia konieczne jest przekazanie aktualnej jego pozycji. Przy dodawaniu nowych zdjęć pola pozycji nie należy przekazywać. Pola `Url` i `UrlMini` zawierają wyłącznie informację zwrotną i są ignorowane w procesie edycji.

3.11. Usuwanie inwestycji

Usuwanie inwestycji jest realizowane przez metodę `doInvestmentInactivate()`. Jako parametry podajemy klucz sesji oraz obiekt inwestycji, w którym należy wypełnić wyłącznie pole `ID`.

Inwestycja jest usuwana nieodwracalnie.

3.12. Przykłady

W przykładach wykorzystano testowe konto agencji z aktywnym kluczem `WebAPI2`:

webapiKey: ABSDFB34FI651N4MJAXY564F

login: twojemail@domena.pl

hasło: twojehaslo

Przykłady napisane są w języku PHP w wersji 5 i wykorzystują wbudowaną klasę SoapClient.

3.12.1. Logowanie

```
<?php
try {
    // klucz dostępu do WebAPI 2.0
    $accessKey = "ABSDFB34FI651N4MJAXY564F";

    // utworzenie obiektu klienta Soap
    $soapClient = new SoapClient(
        "http://beta.otodom.pl/webapi2/?wsdl");

    // zalogowanie się do WebAPI 2.0. Metoda doLogin zwraca klucz sesji.
    $sessionKey = $soapClient->doLogin(
        $accessKey, "twojemail@domena.pl", "twojehaslo" );
    echo "Logowanie powiodło się. Klucz sesji to $sessionKey.\n";
} catch (SoapFault $fault) {
    echo "Logowanie nie powiodło się. Kod błędu: [{"$fault->faultcode}]
        {"$fault->faultstring}\n";
}
...
?>
```

3.12.2. Pobieranie słowników

```
<?php
try {
    // klucz dostępu do WebAPI 2.0
    $accessKey = "ABSDFB34FI651N4MJAXY564F";

    // utworzenie obiektu klienta Soap
    $soapClient = new SoapClient(
        "http://beta.otodom.pl/webapi2/?wsdl");

    $dictionaries = $soapClient->getDictionaries($accessKey);
    echo "Słowniki:<br/><pre>";
}
```

```

print_r($dictionaries);

echo "</pre>";
} catch (SoapFault $fault) {

    echo "Pobranie słowników nie powiodło się. Kod błędu:
    [{$fault->faultcode}] {$fault->faultstring}\n";

}

...

?>

```

3.12.3. Dodawanie/edycja ogłoszenia

```

<?php

//przykładowy sessionId: 113f4d58048d5dd567f7947443efb9ae

...

// klasa obiektu ogłoszenia
class InsertionObject {
    public $ID;
    public $Status;
    public $Country;
    public $Province;
    public $District;
    public $City;
    public $Quarter;
    public $Street;
    public $LocationDescription;
    public $ExpirationDate;
    public $ModificationDate;
    public $RemoteId;
    public $Price;
    public $PriceType;
    public $PriceCurrency;
    public $Area;
    public $MarketType;
    public $ObjectName;
    public $OfferType;
    public $Description;
    public $Investment;
    public $SellerInfo;
    public $AdditionalContact;
    public $AdditionalAddress;
    public $FlatDetails;
    public $HouseDetails;
    public $TerrainDetails;
    public $RoomDetails;
    public $CommercialPropertyDetails;
    public $HallDetails;
    public $GarageDetails;
}

```

```

public $Photos;
public $Url;
public $Title;
}

// klasa obiektu danych mieszkania
class FlatDetailsObject {
public $BuildingType;
public $BuildingMaterial;
public $BuildingFloorsNum;
public $BuildingOwnership;
public $FloorNo;
public $RoomsNum;
public $ExtrasMask;
public $Heating;
public $FreeFrom;
public $Furnished;
public $Rent;
public $RentCurrency;
public $PriceIncludeRent;
}

// klasa obiektu zdjęcia
class PhotoObject {
public $Position;
public $Url;
public $UrlMini;
public $Data;
}

// edycja ogłoszenia
try {
// tworzenie obiektu ogłoszenia
$insertionObject = new InsertionObject();

// aby edytować istniejące ogłoszenie konieczne jest wysłanie jego
// identyfikatora w otoDom
// $insertionObject->ID = 67235329;

// wypełnianie obiektu ogłoszenia danymi
$insertionObject->ID = 0;
$insertionObject->ObjectName = 0;
$insertionObject->OfferType = 0;
$insertionObject->Country = 1;
$insertionObject->Province = 15;
$insertionObject->City = "Poznań";
$insertionObject->PriceCurrency = 1;
$insertionObject->PriceType = 0;
$insertionObject->Price = 100000;
$insertionObject->Area = 25;

// tworzenie powiązanego obiektu z danymi mieszkania
$insertionObject->FlatDetails = new FlatDetailsObject();
// wypełnianie obiektu danymi
$insertionObject->FlatDetails->RoomsNum = 1;
$insertionObject->FlatDetails->ExtrasMask = array(0,2);

//tworzenie zlozonego typu soap, InsertionObject jest nazwa
//zlozonego typu w wsdl

```

```

$insertionObjectWSDL = new SoapVar($insertionObject,
                                   SOAP_ENC_OBJECT, "InsertionObject");

// wykonanie edycji ogłoszenia i pobranie identyfikatora oferty w otoDom
$insertionID = $soapClient->doInsertionEdit($sessionKey,
                                           $insertionObjectWSDL);

echo "Dodanie ogłoszenia powiodło się. Identyfikator ogłoszenia to:
      $insertionID\n";
} catch (SoapFault $fault) {
    echo "Dodanie ogłoszenia nie powiodło się. Kod błędu:
        [{$fault->faultcode}] [{$fault->faultstring}\n";
}

// edycja zdjęć w ogłoszeniu
try {

    // tworzenie obiektu ogłoszenia zawierającego wyłącznie identyfikator
    // otoDom
    $insertionObject = new InsertionObject();
    $insertionObject->ID = $insertionID;
    //tworzenie zlozonego typu soap, InsertionObject jest nazwa
    //zlozonego typu w wsdl
    $insertionObjectWSDL = new SoapVar($insertionObject,
                                       SOAP_ENC_OBJECT, "InsertionObject");

    // tworzenie obiektu zdjęcia
    $photoObject = new PhotoObject();

    // aby edytować konkretne istniejące zdjęcie, należy przekazać jego
    // pozycję
    //$photoObject->Position = 3;

    // dołączenie ciała zdjęcia zakodowanego w formacie BASE64
    $photoObject->Data = base64_encode(@file_get_contents("test1.jpg"));

    // dodanie pierwszego zdjęcia
    $photoPosition = $soapClient->doInsertionPhotoEdit($sessionKey,
                                                       $insertionObjectWSDL, $photoObject);
    echo "Dodanie zdjęcia powiodło się. Pozycja zdjęcia to: $photoPosition\n";

    // dołączenie ciała zdjęcia zakodowanego w formacie BASE64
    $photoObject->Data = base64_encode(@file_get_contents("test2.jpg"));

    // dodanie drugiego zdjęcia
    $photoPosition = $soapClient->doInsertionPhotoEdit($sessionKey,
                                                       $insertionObjectWSDL, $photoObject);
    echo "Dodanie zdjęcia powiodło się. Pozycja zdjęcia to: $photoPosition\n";
} catch (SoapFault $fault) {
    echo "Dodanie zdjęć nie powiodło się. Kod błędu: [{$fault->faultcode}]
        [{$fault->faultstring}\n";
}

...

?>

```

4. Usługa dla partnerów - transfer ogłoszeń

4.1. Dostęp do usługi

Aby uzyskać dostęp do interfejsu otoDom WebAPI 2.0 niezbędne jest posiadanie własnego klucza dostępowego oraz aktywne konto partnera w serwisie otoDom. W celu otrzymania klucza należy skontaktować się z Działem Technicznym otoDom telefonicznie lub wysyłając w tej sprawie email na adres: tech@otodom.pl.

4.2. Mechanizm transferu

Listę transferowanych ogłoszeń można uzyskać przy pomocy metody `getTransferUnits`. Lista ta składa się z jednostek transferu - obiektów `TransferUnitObject`. Każda jednostka transferu posiada swój identyfikator, typ akcji, datę oraz dane ogłoszenia lub inwestycji. Jednostka transferu może reprezentować jedną z 3 akcji na ogłoszeniu:

- START – pojawienie się nowego ogłoszenia,
- CHANGE – zmiana danych ogłoszenia (jeżeli ogłoszenie nie występuje w systemie, akcję należy traktować tak samo jak akcję START),
- STOP – usunięcie ogłoszenia (w przypadku tej akcji nie jest przekazywany obiekt ogłoszenia).

Unikalną jednostkę transferu należy przetwarzać tylko raz. Po przetworzeniu konieczne jest wysłanie potwierdzenia przy pomocy metody `doTransferInsertionsAck`. Dopiero po potwierdzeniu przetworzenia jednostka transferu zostanie usunięta z kolejki.

W zależności od tego, czy jednostka transferu dotyczy ogłoszeń czy inwestycji dane przekazywane są w polu `Insertion` (dla ogłoszeń) lub `Investment` (dla inwestycji). Jednostka transferu nie może zawierać jednocześnie obu informacji.

4.3. Przykłady

W przykładach wykorzystano testowe konto partnera z aktywnym kluczem WebAPI 2.0:

```
webapiKey: 505939D49F904D392770A20308B0E9AA
login:     twojlogin
hasło:     twojehaslo
```

Przykłady napisane są w języku PHP w wersji 5 i wykorzystują wbudowaną klasę `SoapClient`.

4.3.1. Logowanie

```
<?php
try {
    // klucz dostępu do WebAPI 2.0
    $accessKey = "505939D49F904D392770A20308B0E9AA";

    // utworzenie obiektu klienta Soap
    $soapClient = new SoapClient(
        "http://beta.otodom.pl/webapi2/?wsdl");

    // zalogowanie się do WebAPI 2.0. Metoda doLogin zwraca klucz sesji.
    $sessionKey = $soapClient->doLogin(
        $accessKey, "twojlogin", "twojehaslo" );
    echo "Logowanie powiodło się. Klucz sesji to $sessionKey.\n";
} catch (SoapFault $fault) {
    echo "Logowanie nie powiodło się. Kod błędu: [{$fault->faultcode}]
        {$fault->faultstring}\n";
}
...
?>
```

4.3.2. Pobieranie słowników

```
<?php
try {
    // klucz dostępu do WebAPI 2.0
    $accessKey = "505939D49F904D392770A20308B0E9AA";

    // utworzenie obiektu klienta Soap
    $soapClient = new SoapClient(
        "http://beta.otodom.pl/webapi2/?wsdl");

    $dictionaries = $soapClient->getDictionaries($accessKey);
    echo "Słowniki:<br/><pre>";

    print_r($dictionaries);

    echo "</pre>";
} catch (SoapFault $fault) {
    echo "Pobranie słowników nie powiodło się. Kod błędu:
        [{$fault->faultcode}] {$fault->faultstring}\n";
}
```

```
}  
...  
?>
```

4.3.3. Pobieranie transferowanych ogłoszeń

```
<?php  
//przykładowy sessionId: 113f4d58048d5dd567f7947443efb9ae  
...  
try {  
    // pobieranie listy transferowanych ogłoszeń  
    $transferUnits = $soapClient->getTransferUnits($sessionId,10);  
  
    $transferUnitsAck = array();  
  
    // przetwarzanie jednostek transferu  
    foreach($transferUnits as $transferUnit) {  
        ...  
        // przetwarzanie danych  
        ...  
        // tworzenie listy przetworzonych jednostek transferu  
        $transferUnitsAck[] = $transferUnit->ID;  
    }  
  
    // potwierdzenie przetworzenia odebranych jednostek transferu  
    $soapClient->doTransferUnitsAck($sessionId,$transferUnitsAck);  
  
} catch (SoapFault $fault) {  
    echo "Pobranie transferowanych ogłoszeń nie powiodło się. Kod błędu:  
    [{$fault->faultcode}]  
    {$fault->faultstring}\n";  
}  
...  
?>
```

5. Obiekty i pola

5.1. Lista pól wymaganych przy edycji

Flat	House	Terrain	Room	CommercialProperty	Hall	Garage
Price	Price	Price	Price	Price	Price	Price
PriceType	PriceType	PriceType	PriceType	PriceType	PriceType	PriceType
PriceCurrency	PriceCurrency	PriceCurrency	PriceCurrency	PriceCurrency	PriceCurrency	PriceCurrency
Area	Area	Area	MarketType	Area	Area	MarketType
MarketType	MarketType	MarketType		MarketType	MarketType	

Poza polami wymienionymi powyżej konieczne jest ustalenie lokalizacji. W przypadku Polski oznacza to obowiązkowe wypełnienie pól `Country`, `Province`, `District`, `City`. W przypadku zagranicy `Country` oraz `City`.

5.2. Ograniczenia dotyczące wybranych pól

- `City` – maksymalna długość 32 znaki
- `Quarter` – maksymalna długość 64 znaki
- `Description` – nie ma ograniczenia długości, z treści wycinane są wszystkie tagi HTML
- `Pictures` – liczba zdjęć jest ograniczona do 9ciu, dozwolone formaty plików to JPEG i GIF, maksymalny rozmiar pojedynczego pliku to 300 KB

5.3. Lista pól wykluczonych przy edycji

- `SellerInfo` – informacja zwrotna wyłącznie dla partnerów pobierających ogłoszenia
- `Photos` – zdjęcia należy dodawać i edytować przy użyciu dedykowanych metod `doInsertionPhotoEdit`, `doInsertionPhotoRemove`
- `Status` – informacja zwrotna
 - `ExpirationDate` – informacja zwrotna
 - `ModificationDate` – informacja zwrotna
 - `Url` – informacja zwrotna

5.4. Lista pól

5.4.1. InsertionObject

- ID (integer) - Identyfikator ogłoszenia
- Status (integer/słownik) - Status ogłoszenia
- Country (integer/słownik) - Kraj
- Province (integer/słownik) - Województwo
- District (integer/słownik) - Powiat
- City (string) - Miasto
- Quarter (string) - Dzielnica
- Street (string) - Ulica
- LocationDescription (string) - Dodatkowy opis lokalizacji
- ExpirationDate (string) - Data wygaśnięcia ogłoszenia
- ModificationDate (string) - Data ostatniej modyfikacji ogłoszenia
- RemoteId (string) - Numer oferty w biurze
- Price (float) - Cena
- PriceType (integer/słownik) - Rodzaj ceny
- PriceCurrency (integer/słownik) - Waluta
- Area (float) - Powierzchnia
- MarketType (integer/słownik) - Typ rynku
- ObjectName (integer/słownik) – Rodzaj oferty
- OfferType (integer/słownik) - Typ oferty
- Description (string) - Opis
- Investment (integer) – Identyfikator powiązanej inwestycji
- SellerInfo (SellerInfoObject) - Dane sprzedawcy
- FlatDetails (FlatDetailsObject) - Dane szczegółowe dotyczące mieszkania
- HouseDetails (HouseDetailsObject) - Dane szczegółowe dotyczące domu
- TerrainDetails (TerrainDetailsObject) - Dane szczegółowe dotyczące gruntu, działki
- RoomDetails (RoomDetailsObject) - Dane szczegółowe dotyczące pokoju
- CommercialPropertyDetails (CommercialPropertyDetailsObject) - Dane szczegółowe dotyczące lokalu/obiektu użytkowego
- HallDetails (HallDetailsObject) - Dane szczegółowe dotyczące magazynu, hali
- GarageDetails (GarageDetailsObject) - Dane szczegółowe dotyczące garażu
- Photos (ArrayOfPhotoObject) - Zdjęcia

- AdditionalContact (integer) – Identyfikator powiązanych danych kontaktowych
- AdditionalAddress (integer) – Identyfikator powiązanych danych adresowych
- Title (string) – Tytuł ogłoszenia
- Latitude (string) – szerokość geograficzna – pole tylko do odczytu
- Longitude (string) – długość geograficzna – pole tylko do odczytu
- ActivationDate (string) – data pierwszej aktywacji ogłoszenia – tylko do odczytu
- AllegroId (integer) – id ogłoszenia w serwisie Allegro (tylko dla aukcji pochodzących z Allegro) – dla pozostałych ogłoszeń wartość zawsze 0 (zero) – pole tylko do odczytu

5.4.2. SellerInfoObject

- Name (string) - Nazwa
- Street (string) - Ulica
- Postcode (string) - Kod pocztowy
- City (string) - Miasto
- Country (string) - Kraj
- Contacts (ArrayOfContactInfoObject) - Nazwa

5.4.3. FlatDetailsObject

- BuildingType (integer/słownik) - Rodzaj zabudowy
- BuildingMaterial (integer/słownik) – Materiał budowy
- BuildingFloorsNum (integer/słownik) - Liczba pięter w budynku
- BuildingOwnership (integer/słownik) - Forma własności
- FloorNo (integer/słownik) - Piętro
- RoomsNum (integer) - Liczba pokoi
- BuildYear (integer) - Rok budowy
- ExtrasMask (ArrayOfinteger/słownik) - Informacje dodatkowe
- Heating (integer/słownik) – Rodzaj ogrzewania
- FreeFrom (string) – Od kiedy wolne (data w formacie YYYY-MM-DD)
- Furnished (string) – Czy umeblowane (y/n)
- Rent (float) – Wysokość czynszu
- RentCurrency (integer/słownik) – waluta czynszu
- PriceIncludeRent (string) – Cena zawiera czynsz (y/n)
- ConstructionStatus (integer/słownik) - Stan wykończenia
- WindowsTypes (integer/słownik) – Typ okien

- SecurityMask (ArrayOfinteger/słownik) – Zabezpieczenia
- MediaMask (ArrayOfinteger/słownik) - Media
- EquipmentMask (ArrayOfinteger/słownik) – Wyposażenie
- RentToStudents (boolean) – Wynajmę również studentom
- Deposit (float) – Wysokość kaucji
- DepositCurrency (integer/słownik) – waluta kaucji

5.4.4. HouseDetailsObject

- BuildingType (integer/słownik) - Rodzaj zabudowy
- BuildingMaterial (integer/słownik) – Materiał budowy
- TerrainArea (integer) - Powierzchnia działki m2
- ConstructionStatus (integer/słownik) - Stan wykończenia
- BuildYear (integer) - Rok budowy
- RoofType (integer/słownik) - Dach
- Type (integer/słownik) - Typ
- FloorsNum (integer/słownik) - Liczba pięter
- RoomsNum (integer) - Liczba pokoi
- GarretType (integer/słownik) - Poddasze
- WindowsType (integer/słownik) - Okna
- MediaMask (ArrayOfinteger/słownik) - Media
- HeatingMask (ArrayOfinteger/słownik) - Ogrzewanie
- FenceMask (ArrayOfinteger/słownik) - Ogrodzenie
- Location (integer/słownik) - Położenie
- Roofing (integer/słownik) – Rodzaj dachu
- FreeFrom (string) – Od kiedy wolne (data w formacie YYYY-MM-DD)
- Furnished (string) – Czy umeblowane (y/n)
- Rent (float) – Wysokość czynszu
- RentCurrency (integer/słownik) – waluta czynszu
- PriceIncludeRent (string) – Cena zawiera czynsz
- VicinityMask (ArrayOfinteger/słownik) – Okolica
- AccessMask (ArrayOfinteger/słownik) – Dojazd
- SecurityMask (ArrayOfinteger/słownik) – Zabezpieczenia

5.4.5. TerrainDetailsObject

- Type (integer/słownik) - Typ działki
- Dimensions (string) - Wymiary
- MediaMask (ArrayOfinteger/słownik) - Media
- AccessMask (integer/słownik) - Dojazd
- Fence (boolean) - Ogrodzenie
- VicinityMask (ArrayOfinteger/słownik) - Sąsiedztwo
- AccessMask (integer/słownik) - Dojazd

5.4.6. RoomDetailsObject

- PersonsMask (ArrayOfinteger/słownik) - Ilość osób
- Availability (integer/słownik) - Wolny od
- Furnished (string) - Umeblowany (y/n/pusty string)
- BuildingType (integer/słownik) - Rodzaj zabudowy
- RoomPlace (boolean[0/1]) – Miejsce w pokoju
- RoomsNum (integer) – Liczba pokoi w mieszkaniu
- Rent (float) – Wysokość czynszu / dodatkowych opłat
- RentCurrency (integer/słownik) – waluta czynszu
- Deposit (float) – Wysokość kaucji
- DepositCurrency (integer/słownik) – waluta kaucji
- Balcony (boolean[0/1]) – Pokój z balkonem
- MediaMask (ArrayOfinteger/słownik) - Media
- EquipmentMask (ArrayOfInteger/słownik) – Wyposażenie
- WomenInFlat (integer) – Liczba kobiet w mieszkaniu
- MenInFlat (integer) – Liczba mężczyzn w mieszkaniu
- WomenInRoom (integer) – Liczba kobiet w pokoju
- MenInRoom (integer) – Liczba mężczyzn w pokoju
- PreferredSex (integer/słownik) – Preferowana płeć
- PreferredProfession (integer/słownik) – Preferowana sytuacja zawodowa
- NonSmokersOnly (boolean[0/1]) – Tylko dla niepalących

5.4.7. CommercialPropertyDetailsObject

- PropertyUseMask (ArrayOfinteger/słownik) - Przeznaczenie

- ExtrasMask (ArrayOfinteger/słownik) - Inform. dodatkowe
- Floor (integer/słownik) - Piętro
- FreeFrom (string) – Od kiedy wolne (data w formacie YYYY-MM-DD)
- Furnished (string) – Czy umeblowane (y/n)
- BuildingType (integer/słownik) Umiejscowienie lokalu
- TerrainArea – (integer) – Powierzchnia działki
- ConstructionStatus (integer/słownik) - Stan wykończenia
- BuildYear (integer) – Rok budowy
- MediaMask (ArrayOfinteger/słownik) - Media
- SecurityMask (ArrayOfInteger/słownik) – Zabezpieczenia

5.4.8. HallDetailsObject

- Height (integer/słownik) - Wysokość
- AccessMask (ArrayOfinteger/słownik) - Dojazd
- Parking (boolean) - Usytuowanie
- Structure (integer/słownik) - Konstrukcja
- MediaMask (ArrayOfinteger/słownik) - Media
- Heating (boolean) - Ogrzewanie
- UseMask (ArrayOfinteger/słownik) - Przeznaczenie
- ParkingType (integer/słownik) - Parking
- Flooring (integer/słownik) - Posadzka
- ConstructionStatus (integer/słownik) - Stan wykończenia
- OfficeSpace (boolean) – Pomieszczenia biurowe
- SocialFacilities (boolean) – Zaplecze socjalne
- Ramp (boolean) – Rampa
- SecurityMask (ArrayOfInteger/słownik) – Zabezpieczenia
- Fence (integer/słownik) - Ogrodzenie

5.4.9. GarageDetailsObject

- Heating (boolean) - Ogrzewanie
- Lighting (boolean) - Oświetlenie
- Localization (integer/słownik) - Lokalizacja
- Structure (integer/słownik) - Konstrukcja

5.4.10. PhotoObject

- Position (string) - Pozycja
- Url (string) - Odnośnik do zdjęcia
- UrlMini (string) - Odnośnik do miniaturki zdjęcia
- Data (string) - Ciało zdjęcia (Base64)

5.4.11. ContactInfoObject

- Type (string) - Typ kontaktu
- Number (string) - Numer

5.4.12. TransferUnitObject

- ID (integer) - Identyfikator jednostki transferu
- InputDate (string) - Data utworzenia
- Action (string) - Akcja (START|CHANGE|STOP)
- Insertion (InsertionObject) - Ogłoszenie
- Investment (InvestmentObject) – Ogłoszenie inwestycji

5.4.13. InvestmentObject

- ID (integer) - Identyfikator inwestycji
- Country (integer/słownik) - Kraj
- Province (integer/słownik) - Województwo
- District (integer/słownik) - Powiat
- City (string) - Miasto
- Quarter (string) - Dzielnica
- Street (string) - Ulica
- LocationDescription (string) - Dodatkowy opis lokalizacji
- Name (string) – Nazwa inwestycji
- BeginDate (string) - Data rozpoczęcia inwestycji
- FinishDate (string) - Data planowanego zakończenia inwestycji
- State (integer/słownik) - Stan inwestycji
- Description (string) - Opis
- SellerInfo (SellerInfoObject) - Dane sprzedawcy

- Photos (ArrayOfPhotoObject) - Zdjęcia
- Url (string) - Odnośnik do ogłoszenia

5.4.14. AdditionalContactObject

- ID (int) - Id (int)
- Name (string) - Nazwa / opis
- Phone (string) - Telefon
- Email (string) – E-mail

5.4.15. AdditionalAddressObject

- ID (int) - Id
- Name (string) - Nazwa firmy
- Country (int) - Kraj
- City (string) - Nazwa miasta
- Postcode (string) - Kod pocztowy
- Street (string) - Ulica
- Phone (PhoneNumberObject) - Telefon
- Fax (PhoneNumberObject) - Fax
- Mobile (PhoneNumberObject) - Telefon komórkowy
- Email (string) - E-mail

5.4.16. PhoneNumberObject

- CountryCode (string) - Kod kraju
- AreaCode (string) - Kod regionu
- Number (string) - Numer

6. Słowniki

6.1. Aktualizacja słowników

Aktualne słowniki należy pobierać z wykorzystaniem metody `getDictionaries`. Słowniki z serwera testowego różnią się od słowników produkcyjnych. Poniższe zestawienie jest wyłącznie

poglądowne i może różnić się od aktualnego zbioru słownikówn.

6.2. Pola ogólna

6.2.1. Country

- [26] Zjed. Emiraty Arabskie
- [13] Austria
- [5] Belgia
- [17] Bułgaria
- [30] Brazylia
- [12] Kanada
- [25] Cypr
- [3] Czechy
- [20] Niemcy
- [27] Egipt
- [6] Hiszpania
- [4] Francja
- [9] Wielka Brytania
- [19] Grecja
- [15] Chorwacja
- [28] Węgry
- [18] Irlandia
- [10] Włochy
- [31] Litwa
- [8] Luxemburg
- [21] Czarnogóra
- [7] Holandia
- [1] Polska
- [14] Portugalia
- [16] Rumunia
- [22] Rosja
- [2] Słowacja
- [24] Turcja
- [23] Ukraina
- [11] Stany Zjednoczone

6.2.2. Province

- [1] dolnośląskie
- [2] kujawsko-pomorskie
- [3] lubelskie
- [4] lubuskie
- [5] łódzkie
- [6] małopolskie

- [7] mazowieckie
- [8] opolskie
- [9] podkarpackie
- [10] podlaskie
- [11] pomorskie
- [12] śląskie
- [13] świętokrzyskie
- [14] warmińsko-mazurskie
- [15] wielkopolskie
- [16] zachodniopomorskie

6.2.3. District

Ze względu na dużą ilość informacji słownik dostępny jest wyłącznie za pośrednictwem usługi.

6.2.4. PriceCurrency

- [1] PLN
- [2] EUR
- [3] GBP
- [4] USD
- [5] AED
- [6] EGP

6.2.5. Status

- [0] tymczasowe (nieaktywne)
- [1] aktywne
- [2] nieaktywne
- [3] zablokowane (nieaktywne)

6.2.6. PriceType

- [0] brutto
- [1] netto

6.2.7. MarketType

- [0] pierwotny
- [1] wtórny

6.2.8. ObjectName

- [0] Flat (*Mieszkanie*)
- [1] House (*Dom*)
- [2] Terrain (*Grunt, działka*)
- [3] Room (*Pokój*)
- [4] CommercialProperty (*Lokal/obiekt użytkowy*)
- [5] Hall (*Magazyn, hala*)
- [6] Garage (*Garaż*)

6.2.9. OfferType

- [0] sprzedaż
- [1] wynajem

6.3. Pola obiektu FlatDetails

6.3.1. BuildingType

- [0] blok
- [1] kamienica
- [2] dom wolnostojący
- [3] plomba
- [4] szeregowiec
- [5] apartamentowiec
- [6] loft

6.3.2. BuildingMaterial

- [0] cegła
- [1] drewno
- [2] pustak
- [3] keramzyt
- [4] wielka płyta
- [5] beton
- [6] inne
- [7] silikat
- [8] beton komórkowy

6.3.3. BuildingFloorsNum

- [0] do 5 pięter
- [1] 6-11 pięter

- [2] powyżej 11 pięter

6.3.4. BuildingOwnership

- [0] spółdzielcze własnościowe
- [1] spółdzielcze wł. z KW
- [2] pełna własność

6.3.5. FloorNo

- [0] suterena
- [1] parter
- [2] 1
- [3] 2
- [4] 3
- .
- .
- .
- [48] 47
- [49] 48
- [50] 49
- [51] 50
- [52] powyżej 50

6.3.6. ExtrasMask

- [0] balkon
- [1] pom. użytkowe
- [2] garaż
- [3] piwnica
- [4] ogródek
- [5] taras
- [6] winda
- [7] dwupoziomowe
- [8] oddzielna kuchnia
- [9] klimatyzacja
- [10] tylko dla niepalących

6.3.7. Heating

- [0] miejskie
- [1] gazowe
- [2] piece kaflowe
- [3] elektryczne

- [4] inne
- [5] kotłownia

6.3.8. ConstructionStatus

- [0] do zamieszkania
- [1] do wykończenia
- [2] do remontu

6.3.9. WindowsType

- [0] plastikowe
- [1] drewniane
- [2] aluminiowe

6.3.10. SecurityMask

- [0] rolety antywłamaniowe
- [1] drzwi / okna antywłamaniowe
- [2] domofon / wideofon
- [3] monitoring / ochrona
- [4] system alarmowy
- [5] teren zamknięty

6.3.11. MediaMask

- [0] internet
- [1] telewizja
- [2] telefon

6.3.12. EquipmentMask

- [0] pralka
- [1] zmywarka
- [2] lodówka
- [3] kuchenka
- [4] piekarnik
- [5] telewizor

6.4. Pola obiektu HouseDetails

6.4.1. BuildingType

- [0] wolnostojący
- [1] bliźniak
- [2] szeregowiec
- [3] kamienica czynszowa
- [4] dworek/pałac
- [5] gospodarstwo

6.4.2. BuildingMaterial

- [0] cegła
- [1] drewno
- [2] pustak
- [3] keramzyt
- [4] wielka płyta
- [5] beton
- [6] inne
- [7] silikat
- [8] beton komórkowy

6.4.3. ConstructionStatus

- [0] do zamieszkania
- [1] do wykończenia
- [2] do remontu
- [3] stan surowy otwarty
- [4] stan surowy zamknięty

6.4.4. RoofType

- [0] brak
- [1] płaski
- [2] skośny

6.4.5. Type

- [0] mieszkalny
- [1] rekreacyjny

6.4.6. FloorsNum

- [0] parter
- [1] 1 piętro
- [2] 2 piętra
- [3] więcej

6.4.7. GarretType

- [0] brak
- [1] użytkowe
- [2] nieużytkowe

6.4.8. WindowsType

- [0] brak
- [1] plastikowe
- [2] drewniane
- [3] aluminiowe

6.4.9. MediaMask

- [0] prad
- [1] woda
- [2] gaz
- [3] telefon
- [4] kanalizacja
- [5] szambo
- [6] oczyszczalnia

6.4.10. HeatingMask

- [0] gazowe
- [1] węglowe
- [2] biomasa
- [3] pompa ciepła
- [4] kolektor słoneczny
- [5] geotermika
- [6] olejowe
- [7] elektryczne
- [8] miejskie
- [9] kominkowe
- [10] piece kaflowe

6.4.11. FenceMask

- [0] murowane
- [1] siatka
- [2] metalowe
- [3] drewniane
- [4] betonowe
- [5] żywopłot
- [6] inne

6.4.12. Location

- [0] miasto
- [1] pod miastem
- [2] wieś

6.4.13. ExtrasMask

- [0] piwnica
- [1] strych
- [2] garaż
- [3] basen
- [4] Internet
- [5] telewizja kablowa
- [6] klimatyzacja
- [7] umeblowanie

6.4.14. SecurityMask

- [0] rolety antywłamaniowe
- [1] drzwi / okna antywłamaniowe
- [2] domofon / wideofon
- [3] monitoring / ochrona
- [4] system alarmowy
- [5] teren zamknięty

6.5. Pola obiektu TerrainDetails

6.5.1. Type

- [0] budowlana

- [1] rolna
- [2] rekreacyjna
- [3] pod inwestycję
- [4] leśna
- [6] siedliskowa
- [5] inna
- [7] rolno-budowlana

6.5.2. MediaMask

- [0] prad
- [1] woda
- [2] gaz
- [3] telefon
- [4] kanalizacja
- [5] szambo
- [6] oczyszczalnia

6.5.3. AccessMask

- [0] utwardzony
- [1] polny
- [2] asfaltowy

6.5.4. VicinityMask

- [0] las
- [1] jezioro
- [2] otwarty teren
- [3] góry
- [4] morze

6.6. Pola obiektu RoomDetails

6.6.1. PersonsMask

- [0] 1-osobowy
- [1] 2-osobowy
- [2] 3-osobowy
- [3] 4-osobowy

6.6.2. Availability

- [0] od zaraz
- [1] w ciągu m-ca
- [2] później niż miesiąc

6.6.3. BuildingType

- [0] blok
- [1] kamienica czynszowa
- [2] dom wolnostojący

6.6.4. MediaMask

- [0] internet
- [1] telewizja
- [2] telefon

6.6.5. EquipmentMask

- [0] pralka
- [1] zmywarka
- [2] lodówka
- [3] kuchenka
- [4] piekarnik
- [5] telewizor

6.6.6. PreferredSex

- [0] mężczyzna
- [1] kobieta
- [2] para

6.6.7. PreferredProfession

- [0] pracujący
- [1] student

6.7. Pola obiektu CommercialPropertyDetails

6.7.1. PropertyUseMask

- [0] usługowy
- [1] biurowy
- [2] handlowy
- [3] gastronomiczny
- [4] przemysłowy
- [5] hotelowy

6.7.2. ExtrasMask

- [0] witryna
- [1] parking
- [2] dojazd asfaltowy
- [3] winda
- [4] klimatyzacja
- [5] ogrzewanie

6.7.3. Floor

- [0] suterena
- [1] parter
- [2] 1
- [3] 2
- [4] 3
- .
- .
- .
- [48] 47
- [49] 48
- [50] 49
- [51] 50
- [52] powyżej 50

6.7.4. BuildingType

- [0] w centrum handlowym
- [1] w biurowcu
- [2] w bloku
- [3] w kamienicy
- [4] w domu prywatnym
- [5] w budynku zabytkowym

- [6] osobny obiekt

6.7.5. ConstructionStatus

- [0] gotowy
- [1] do wykończenia
- [2] do remontu

6.7.6. MediaMask

- [0] prąd
- [1] woda
- [2] gaz
- [3] telefon
- [4] internet
- [5] telewizja kablowa
- [6] kanalizacja
- [7] szambo
- [8] oczyszczalnia

6.7.7. SecurityMask

- [0] rolety antywłamaniowe
- [1] drzwi / okna antywłamaniowe
- [2] domofon / wideofon
- [3] monitoring / ochrona
- [4] system alarmowy
- [5] teren zamknięty

6.8. Pola obiektu GarageDetails

6.8.1. Localization

- [0] w budynku
- [1] samodzielny
- [2] przy domu

6.8.2. Structure

- [0] murowany

- [1] blaszak
- [2] drewniany
- [3] wiata

6.9. Pola obiektu HallDetails

6.9.1. AccessMask

- [0] utwardzony betonowy
- [1] nieutwardzony
- [2] asfalt

6.9.2. Structure

- [0] stalowa
- [1] murowana
- [2] wiata
- [3] namiotowa
- [4] drewniana
- [5] szklana

6.9.3. MediaMask

- [0] woda
- [1] prąd
- [2] siła
- [3] kanalizacja
- [4] telefon
- [5] gaz
- [6] internet
- [7] szambo
- [8] oczyszczalnia

6.9.4. UseMask

- [0] magazynowe
- [1] produkcyjne
- [2] biurowe
- [3] handlowe

6.9.5. ParkingType

- [0] brak
- [1] utwardzony
- [2] betonowy
- [3] asfaltowy
- [4] nieutwardzony
- [5] kostka brukowa

6.9.6. Flooring

- [0] brak
- [1] pylna
- [2] niepylna

6.9.7. ConstructionStatus

- [0] gotowe
- [1] do wykończenia
- [2] do remontu
- [3] stan surowy otwarty
- [4] stan surowy zamknięty

6.9.8. SecurityMask

- [0] rolety antywłamaniowe
- [1] drzwi / okna antywłamaniowe
- [2] domofon / wideofon
- [3] monitoring / ochrona
- [4] system alarmowy
- [5] teren zamknięty

6.9.9. Fence

- [0] tak
- [1] nieogrodzony

6.10. Pola obiektu Investment

6.10.1. State

- [0] zakończona

- [1] w budowie
- [2] nie rozpoczęta