



# API DOKUMENTACE

XML-RPC ROZHRANÍ

V 1.6

Všechny požadavky zasílejte na ENDPOINT  
<https://www.pronajmy.cz/api/>

Pro testovací účely použijte následující přístupové údaje:  
Pro import dat do našeho systému doporučujeme použít knihovnu **PhpXmlRpc**  
<https://packagist.org/packages/phpxmlrpc/phpxmlrpc>  
<https://github.com/gggeek/phpxmlrpc>

## LOGIN DO KLIENTSKÉ SEKCE

Přihlášení do KLIENTSKÉ SEKCE je možné provést na [www.pronajmy.cz](http://www.pronajmy.cz) (tlačítko v header - Přihlášení / Registrace)

**e-mail:** [apitest@pronajmy.cz](mailto:apitest@pronajmy.cz)  
**Heslo:** apitest2023!

V tomto účtu je aktivní neomezené předplatné a lze testovat přidávání, editaci, mazání inzerátů, přidávání makléřů a všechny ostatní implementované funkce.

Inzeráty vytvořené pod tímto účtem se nezobrazí ve veřejné části portálu Pronajmy.cz  
Je však možné si přidat inzerát zobrazit z klientské sekce (část - MOJE INZERÁTY)  
Inzeráty jsou automaticky po 30 dnech smazány

## API KLÍČE PRO KOMUNIKACI PŘES API

**api\_key:** [apitest@pronajmy.cz](mailto:apitest@pronajmy.cz)  
**api\_password:** boruvka7  
**hashed\_api\_password :** f037a40a55c3a26efcf88b9cb7177fa8  
**client\_id :** 67002

(Tyto informace získá náš zákazník v KLIENTSKÉ SEKCI -- NASTAVENÍ ÚČTU)  
Je tedy na místě zajistit ve vašem CRM systému možnost (formulář) kam zákazník vloží tyto přístupové údaje na základě kterých bude probíhat identifikace a komunikace (import inzerátů).

## PREAMBULE

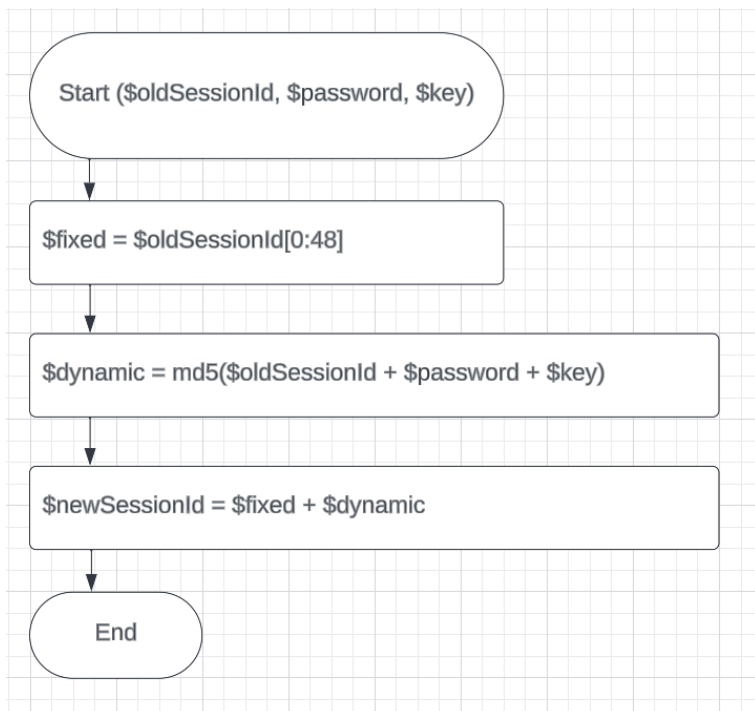
Předpokladem každého dotazu (vyjma úvodního getHash) na API server je použití správného sessionId. V úvodu komunikace se serverem sessionId není známo a proto je nutné si jej nechat skrze metodu getHash vygenerovat. Pro každou další aplikaci sessionId je nutné jej přegenerovat pomocí speciální funkce (například

označenou *computeSessionId*). *SessionId* se po každém dotazu změní a proto je třeba jej přegenerovat i na Vaší straně (straně klienta).

Pro vysvětlení přikládáme obecný diagram komunikace v kombinaci s generováním *sessionId* pro účely přidání inzerátu:



Obecný postup pro přegenerování *SessionId* je následující:



Skládá se z 48 znaků původního sessionId + se připojí hash z původního sessionId, Vašeho api hesla a Vašeho api klíče. Obecně lze konstatovat, že každé sessionId může být využité pro účely komunikace s importním rozhraním **POUZE JEDNOU**, následně je nutné pro další dotaz **VŽDY PŘEGENEROVAT NOVÉ SESSIONID**.

Vzorový PHP kód takové funkce může vypadat následovně:

```
/**
 * @param string $oldSessionId Staré sessionId
 * @param string $password Vaše heslo k importnímu rozhraní
 * @param string $key Váš klíč k importnímu rozhraní
 * @return string Nové sessionId pro použití v následujícím dotazu
 */
function computeSessionId(string $oldSessionId, string $password, string $key):
string
{
    return substr($oldSessionId, 0, 48) . md5($oldSessionId . $password . $key);
}
```

## AUTORIZACE

1. Zavolejte funkci **getHash** pouze s **client\_id**  
a vrátí se **sessionId** (pro pochopení si ji pojmenujeme jako první sessionId 1)

Ukázka PHP kódu:

```
use PhpXmlRpc\Encoder;
use PhpXmlRpc\Client;
use PhpXmlRpc\Request;

$clientId = 67002;
$passwordHash = '33b8ec7ffb030b8a2d52a028b728a7d3';

$encoder = new Encoder();
$client = new Client('https://www.pronajmy.cz/api/');

$request = new Request('getHash', [
    $encoder->encode($clientId),
]);
$response = $client->send($request);
$data = $encoder->decode($response->value());

$sessionId = $data['output'][0]['sessionId'];
echo 'Session ID = ' . $sessionId;
```

3. Zavoláme funkci **login**

Tato funkce slouží k přihlášení (aktivaci) Vaší session pro účely volání dalších metod importního rozhraní. Bez volání této funkce nebude Vaše sessionId přihlášené a nebudete moci provádět další operace skrze importní rozhraní. Vstupním parametrem je pouze **PŘEGENEROVANÉ** sessionId na základě odpovědi z metody getHash.

4. Pro další operace jako **addAdvert**, **delAdvert**, **listAdvert** atd. je nutné provést přepočítání **sessionId** podle  **bodu 2.**

5. po provedení potřebných operací zavoláme metodu **logout** (opět s nově vypočteným **sessionId**)

- **sessionId** má pro další výpočet platnost 15 minut

## FUNKCE API

Naše API podporuje tyto funkcionality:

- **getHash** (získání řetězce pro hashování hesla) - slouží pro přihlášení k API
- **login** - pro provedení loginu za pomoci potvrzení session jako autorizované
- **logout** - odhlášení klienta z API
- **addAdvert** - metoda pro přidání inzerátů přes API
- **delAdvert** - vymazání inzerátu přes API
- **listAdvert** - výpis všech inzerátů přihlášené realitní kanceláře
- **addPhoto** - vložení fotografie k již přidanému inzerátu
- **delPhoto** - vymazání fotografie
- **listPhoto** - výpis fotografií u konkrétního existujícího inzerátu
- **addSeller** - přidání realitního makléře k realitní kanceláři
- **delSeller** - odebrání existujícího makléře
- **listSeller** - výpis makléřů realitní kanceláře
- **listInquiry** - Výpis unikátních identifikátorů inquiry\_id všech zpráv pro všechny inzeráty, které byly v zadané datum odeslány RK.

- **getInquiry** - Obsah zprávy pro RK pro identifikátor zprávy inquiry\_id.
- **listFullInquiry** - Výpis celých zpráv, ze všech inzerátů, odeslaných na RK v zadané datum.

Pozn. TOPOVÁNÍ INZERÁTU musí makléř nebo realitní kancelář provést z klientské sekce Pronájmy.cz

## Metody importního rozhraní

### Základní metody

*version(): struct*

Informace o aktuální verzi API.

### Autorizační metody

*getHash(int \$clientId): struct*

Vygenerování sessionId pro klienta.

*login(string \$sessionId): struct*

Přihlášení (aktivace) sessionId.

*logout(string \$sessionId): struct*

Odhlášení (deaktivace) sessionId.

## Metody maklérů

*addSeller(string \$sessionId, int \$sellerId, string \$sellerRkId, struct \$data): struct*

Vytvoření nového či editace existujícího makléře ze zadaných dat (\$data). Zda má jít o aktualizaci či vytvoření nového, určují parametry *sellerId* (ID pronájmy.cz) či *sellerRkId* (Vaše ID) – pokud jsou vyplněné, předpokládá importní rozhraní, že jde o aktualizaci.

Makléře je možné přidat i v případě, že nemá vytvořený účet na Pronájmy.cz  
V takovém případě se účet automaticky vytvoří, spáruje s realitní kanceláří a přístupy do klientské sekce se odešlou na e-mail makléře.

Seznam atributů makléře (parametr \$data):

Název	Typ	Popis	Povinné
client_login	String	E-mail	Vždy
client_name	String	Jméno	Vždy
contact_gsm	String	Mobil	Vždy
client_ic	Int	IČ	Vždy
client_is_employee	Int	makléř je zaměstnanec na HPP	Vždy
contact_gsm_code	String	předvolba telefonu	Ne
contact_phone	String	telefon	Ne
contact_phone_code	String	předvolba telefonu	Ne
contact_email	String	kontaktní e-mail	Ne
makler_note	String	poznámka	Ne

broker_specialization	String	specializace	Ne
suppress_notification	Boolean	Potlačí zaslání e-mailové notifikace při vytváření nového makléře	Ne, výchozí hodnota je false

*delSeller(string \$sessionId, int \$sellerId, string \$sellerRkid): struct*

Odstranění existujícího makléře.

*listSeller(string \$sessionId): struct*

Výpis existujících makléřů.

## Metody inzerátů

*addAdvert(string \$sessionId, struct \$data): struct*

Vytvoření nového či editace existujícího inzerátu ze zadaných dat (\$data). Zda má jít o aktualizaci či vytvoření nového inzerátu, určuje data atribut *advert\_id* či *advert\_rkid* – pokud je vyplněný, předpokládá importní rozhraní, že jde o aktualizaci.

Seznam atributů inzerátu (parametr \$data):

Název	Typ	Popis	Povinné
publish_status	Codebook 0 1 3	Stav inzerátu Zpracovává se Zveřejněno Vypnuto administrátorem	Pouze vrací hodnotu
advert_function	Codebook  1 2 3 4	Typ inzerátu  Prodej Pronájem Dražby Podíly	Vždy
advert_type	Codebook  1 2 3 4 5 20	Kategorie inzerátu  Byty Domy Pozemky Komerční Ostatní Projekty	Vždy
advert_room_count	Codebook  1 2 3 4 5 6	Počet pokojů  1 pokoj 2 pokoje 3 pokoje 4 pokoje 5 a více pokojů Atypický	Vždy
advert_subtype	Codebook	Podtyp kategorie	





basin	Boolean 1 0	Bazén Ano Ne	
building_condition	Codebook 1 2 3 4 5 6 7 8 9 10	Stav objektu Velmi dobrý Dobrý Špatný Ve výstavbě Projekt Novostavba K demolici V rekonstrukci Před rekonstrukcí Po rekonstrukci	
building_type	Codebook 1 2 3 4 5 6 7	Budova Dřevěná Cihlová Kamenná Montovaná Panelová Skeletová Smišená	Povinné pro byty, domy, komerční
cellar	Boolean 1 0	Sklep Ano Ne	
easy_access	Codebook 1 2	Bezbariérový přístup Ano Ne	
electricity	Multiselect 1 2 3	Elektřina 120V 230V 400V	
elevator	Codebook 1 2	Výtah Ano Ne	
energy_efficiency_rating	Codebook 1 2 3 4 5 6 7	Třída energetické náročnosti A - Mimořádně úsporná B - Velmi úsporná C - Úsporná D - Méně úsporná E - Nehospodárná F - Velmi nehospodárná G - Mimořádně nehospodárná	
energy_performance_certificate	Codebook 1 2 3	Podle vyhlášky Č. 148/2007 Sb. Č. 78/2013 Sb. Č. 264/2020 Sb.	
extra_info	Codebook 1 2	Stav inzerátu Rezervováno Prodáno	
flat_class	Codebook	Typ bytu	

	1 2 3	Loft Mezonet Podkrovní	
furnished	Codebook 1 2 3	Zařízeno Ano Ne Částečně	
garage	Boolean 1 0	Garáž Ano Ne	
gas	Multiselect 1 2	Rozvod plynu Individuální Plynovod	
gully	Multiselect 1 2 3 4 5	Typ odpadu Veřejná kanalizace ČOV pro celý objekt Septik Jímka Trativod	
heating	Multiselect 1 2 3 4 5 6 7 8 9	Rozvod topení Veřejná Lokální plynové Lokální tuhá paliva Lokální elektrické Ústřední plynové Ústřední tuhá paliva Ústřední elektrické Ústřední dálkové Jiné Podlahové	
locality_inaccuracy_level	Int 1 2 3	Úroveň přesnosti ukazatele Přesná adresa O stupěň zneřesněná adresa (ulice nebo část města) Pouze konkrétní obec	
loggia	Boolean 1 0	Lodžie Ano Ne	
object_kind	Codebook 1 2 3 4	Typ objektu Řadový Rohový V bloku Samostatný	
object_location	Codebook 1 2 3 4 5 6 7	Umístění Centrum obce/města Klidná část obce Rušná část obce Okraj obce Sídliště Polo-samota Samota	
object_type	Codebook 1	Typ stavby Přízemní	Povinné u domů

	2	Patrový	
ownership	Codebook 1 2 3	Vlastnictví  Osobní Družstevní Státní/Obecní	Povinné u bytů
parking_lots	Boolean  1 0	Parkování  Ano Ne	
personal	Codebook  1 2	Převod do osobního vlastnictví  Ano Ne	
protection	Codebook  1 2 3	Ochrana  Ochranné pásmo Národní park CHKO	
road_type	Multiselect  1 2 3 4 5 6 7	Příjezdová cesta  Betonová Dlážděná Asfaltová Neupravená Zpevněná Šterková Šotolinová	
surroundings_type	Multiselect  1 2 3 4 5 6 7 8	Zástavba  Obytná Obchodní a obytná Obchodní Komerční Průmyslová Venkovská Rekreační Rekreačně nevyužitá	
telecommunication	Multiselect  1 2 3 4 5 6	Telekomunikace  Telefon Internet Satelit Kabelová televize Kabelové rozvody Ostatní	
terrace	Boolean  1 0	Terasa  Ano Ne	
transport	Multiselect  1 2 3 4 5	Doprava  Vlak Dálnice Silnice MHD Autobus	
water	Multiselect  1 2 4	Vodní zdroj  Místní zdroj Dálkový vodovod Studna	

	5	Vrt	
advert_price	int	Cena	Vždy
description	String	Popis	
estate_area	Int		Povinné pro pozemky a domy
floor_number	Int		Povinné pro byty
locality_latitude	Double	Zeměpisná šířka	
locality_longitude	Double	Zeměpisná délka	
locality_ruian		RUIAN lokality	
locality_ruian_level		RUIAN level lokality, aktuálně podporované úrovně: Okres (1), Obec (2), Část obce (5), Ulice (7), Adresa (11), Městská část (17)	
locality_uir		UIR lokality	
locality_uir_level		UIR-level lokality	
seller_id		ID makléře	
seller_rkid		Váš identifikátor makléře	
usable_area		Užitná plocha	
advert_code		ID inzerátu	
advert_price_text_note		Popis ceny	
building_area	Int	Zastavěná plocha	
extra_info	Codebook		
first_tour_date	Datetime	Datum prohlídky	
floor_area	Int	Podlahová plocha	
floors	Int	Počet podlaží	
object_age	Int	Rok výstavby	
parking		Počet parkovacích míst	
ready_date	Datetime	K dispozici od	Povinné pro pronájem
reconstruction_year	Int	Rok rekonstrukce	
garden_area	Int	Plocha zahrady	
underground_floors	Int	Počet podzemních podlaží	
user_status	Boolean	Aktivní	

advert_id	int	ID inzerátu pronajmy.cz (pro aktualizaci)	Ne
advert_rkid	String	Vaše ID inzerátu (pro aktualizaci)	Ne

*delAdvert(string \$sessionId, int \$advertId, string \$advertRkid): struct*

Odstranění existujícího inzerátu.

*listAdvert(string \$sessionId): struct*

Výpis existujících inzerátů.

## Metody fotografií inzerátů

*addPhoto(string \$sessionId, int \$advertId, string \$advertRkid, struct \$data): struct*

Přidání či editace fotografie k inzerátu, maximální počet fotografií u inzerátu je 30, minimální počet u platného inzerátu je 5. Zda má jít o aktualizaci či vytvoření nové fotografie inzerátu, určuje data atribut *photo\_id* či *photo\_rkid* – pokud je vyplněný, předpokládá importní rozhraní, že jde o aktualizaci.

Seznam atributů fotografie (parametr \$data):

Název	Typ	Popis	Povinné
data	String	binární data obrázku zakódovaná pomocí base64	Vždy
main	Int	hlavní fotka = 1', vedlejší fotky = 0	Vždy
order	Int	řazení v rámci vedlejších fotek	Vždy
alt	Int	popisek obrázku	Ne
photo_id	Int	ID fotky pronajmy.cz (pro aktualizaci)	Ne

photo_rkid	String	Vaše ID fotky (pro aktualizaci)	Ne
------------	--------	---------------------------------	----

*delPhoto(string \$sessionId, int \$photoId, string \$photoRkid): struct*

Odstranění existující fotografie inzerátu.

*listPhoto(string \$sessionId, int \$advertId, string \$advertRkid): struct*

Výpis existujících fotografií konkrétního inzerátu.

## Metody statistik inzerátů

*listStat(string \$sessionId, struct \$advertId, struct \$advertRkid): struct*

Vrátí statistiky všech inzerátů uvedených v seznamech parametrů *\$advertId* a *\$advertRkid*.

*listDailyStat(string \$sessionId, int \$advertId, string \$advertRkid): struct*

Vrátí denní statistiky konkrétního inzerátu.

*listAllDailyStat(string \$sessionId, DateTime \$date): struct*

Vrátí všechny denní všech inzerátů v požadovaný den.

*listSellerStat(string \$sessionId, int \$sellerId, string \$sellerRkid, DateTime \$from, DateTime \$till): struct*

Výpis denních statistik inzerátů jednoho makléře. Čas je zadán uzavřeným intervalem <from, till>

## Metody poptávek inzerátů

*listInquiry(string \$sessionId, DateTime \$date): struct*

Výpis unikátních identifikátorů inquiry\_id všech zpráv pro všechny inzeráty, které byly v zadané datum odeslány RK.

*getInquiry(string \$sessionId, int \$id): struct*

Obsah zprávy pro RK pro identifikátor zprávy inquiry\_id.

*listFullInquiry(string \$sessionId, DateTime \$date): struct*

Výpis celých zpráv, ze všech inzerátů, odeslaných na RK v zadané datum.

## Naše importní rozhraní vychází ze struktury importního rozhraní **sreality.cz**

### ODLIŠNOSTI:

- Nepodporujeme import projektů jako takových, projekt lze přidat jako klasický inzerát (např.: developerský projekt) viz. číselník **advert\_type** a **advert\_subtype**
- Minimální počet fotografií k inzerátu je **5** a maximální počet fotografií je **30**
- Nepodporujeme jazykové mutace inzerátů (pouze čeština)
- Inzeráty mají pouze 3 stavy v **publish\_status**
- Nelze provádět TOPOVÁNÍ skrze importní rozhraní (Tuto funkci připravujeme)
- Nepřijímáme VIDEO nebo 3D prohlídky

### POZNÁMKY K API ROZHRAŇÍ

- K adrese nám stačí zaslat RUIAN nebo GPS pozice, adresu zjistíme na základě RUIAN ID nebo GPS souřadnic
- Názvy polí a číselníky jsou shodné s API Import sreality.cz
- U datového typu **boolean** lze zasílat TRUE a FALSE, ale i logickou 1 jako TRUE nebo logickou 0 jako FALSE
- V případě UPDATU fotografie (např. změna pořadí) není nutné zasílat field **data** (tedy není nutné posílat fotografii znovu)
- Při přidání makléře (addSeller) který v našem systému ještě neexistuje, se tomuto makléři na e-mail automaticky odešlou přístupové údaje do naší klientské sekce, kde může inzeráty TOPOVAT, UPRAVOVAT nebo si nastavit svoji fotografii

V případě dotazů kontaktujte naše **IT oddělení** na e-mailu: [it@pronajmy.cz](mailto:it@pronajmy.cz)

## NÁVRATOVÉ STATUS KÓDY

KÓD a popis	POPIS STAVU
200; Advert has been created	Inzerát byl úspěšně přidán
200; Advert has been deleted	Inzerát byl úspěšně smazán
200; Advert has been updated	Inzerát byl úspěšně aktualizován
200; Image has been created	Obrázek byl úspěšně přidán
200; Image has been deleted	Obrázek byl úspěšně uložen
200; Image has been updated	Obrázek byl úspěšně aktualizován
200; List of adverts	Inzeráty byly úspěšně vypsány
200; List of photos	Fotografie byly úspěšně vypsány
200; List of sellers	Makléři byli úspěšně vypsáni
200; OK	Standardní status code

200; Seller has been created	Maklér byl úspěšně vytvořen
200; Seller has been deleted	Maklér byl úspěšně smazán
200; Seller has been updated	Maklér byl úspěšně aktualizován
402; Client not found	RK Agentura nebyla nalezena
404; Advert not found	Inzerát nebyl nalezen
404; Image not found	Obrázek nebyl nalezen
407; Session is bad	Session není korektní
407; There is no active membership	RK Agentura nemá aktivní předplatné
407; Your membership has reached the maximum number of adverts	RK Agentura vyčerpala limit aktivních inzerátů
425; Missing required parameter: {name}	Zaslaný parametr je chybný nebo nebyl zaslán vůbec
461; Seller not found	Maklér nebyl nalezen
462; {name} already exists	Vztahuje se na existující RKID u Advertu