

# **API Datový sklad MUNIS**

**API Datový sklad MUNIS**  
**popis rozhraní MUNIS DS, verze 0.2**

Copyright © 2021 Triada, spol. s r. o.

Triada, spol. s r. o.  
U Svobodárny 1110/12  
190 00 Praha 9-Libeň

tel.: 284 001 284  
fax: 284 818 027

e-mail: [info@triada.cz](mailto:info@triada.cz)  
WWW: <http://www.triada.cz>

## Verzování dokumentu

<b>Datum</b>	<b>Verze</b>	<b>Popis</b>
<b>2. 9. 2021</b>	<b>verze 0.2</b>	<b>Počáteční dokument</b>

# Obsah

<b>1. Funkce DS Munis přístupné pro napojení portálu občana</b>	<b>5</b>
1.1. Funkce pro spisovou službu .....	5
1.1.1. NajdiPodatele .....	5
1.1.2. DejPodaniPodatele .....	5
1.1.3. DejDetailPodani .....	6
1.1.4. DejHistoriiPodani .....	6
1.1.5. DejPrilohuPodani .....	6
1.1.6. VysledekPozadavku .....	6
1.1.7. Příklady použití funkcí pro spisovou službu .....	6
1.2. Funkce pro volby .....	7
1.2.1. NajdiVolice .....	7
1.2.2. DejUdajeVolice .....	7
1.2.3. Příklady použití funkcí pro volby .....	7
1.3. Funkce pro poplatky .....	7
1.3.1. NajdiPlatce .....	8
1.3.2. DejUdajePlatce .....	8
1.3.3. Příklady použití funkcí pro volby .....	8
1.4. On-line verze dokumentace .....	8
<b>2. Přílohy</b>	<b>9</b>
2.1. Strojově generovaná dokumentace API MUNIS DS .....	10

## 1. Funkce DS Munis přístupné pro napojení portálu občana

Datový sklad Munis obsahuje údaje z IS Munis, které jsou ukládány buď jako kompletní data vybraných agend (např. kategorie pohledávek z poplatků, stálý seznam voličů) nebo jako sledovaná data (sledování podatelů, kteří si vyžádali informaci o svých podáních). Datový sklad je vždy stranou volanou (z Portálu občana, z IS Munis), nikdy stranou volající. IS Munis je stranou volající.

Datový sklad Munis obsahuje funkce umožňující ztotožnění FO přihlášené přes portál občana s bezvýznamovými identifikátory FO v jednotlivých agendách uložených v datovém skladu. Tyto identifikátory jsou vždy specifické pro danou agendu (ID PODATELE, ID VOLICE, ID PLATCE) a nelze je tak mezi agendami zaměňovat. Dále se mohou též lišit podle parametrů dané instalace datového skladu, avšak vždy splňují obecný typ NVARCHAR 50.

Funkce pro nalezení uvedených identifikátorů mohou vrátit jedno nebo více ID. Protože ztotožnění vychází z identifikace FO dané správním řádem a zahrnuje jméno, příjmení, datum narození a místo trvalého bydliště (dle struktury vycházející z návratové struktury ROB), má se za to, že je vždy jednoznačné. V případě více návratových ID tak nejde o různé FO, ale o duplicitní evidenci stejné osoby v primárních datech agendy.

Datový sklad Munis dále obsahuje funkce na získání detailních dat k těmto FO (získaných identifikátorům) vedených v daných agendách.

Navázané agendy:

- Spisová služba
- Volby
- Poplatky

### 1.1. Funkce pro spisovou službu

Data v datovém skladu se aktualizují buď požadavkem na nové sledování podatele, nebo operacemi ve spisové službě. Kontrola požadavků na sledování nových podatelů se provádí jednou za 24 hod v časovém okně dle nastavení.

#### 1.1.1. NajdiPodatele

Funkce projde data podatelů již uložená v datovém skladu. Pokud najde plnou shodu, vrátí seznam ID těchto podatelů. Pokud nenajde žádná data založí požadavek na doplnění datového skladu a vrátí token tohoto požadavku.

- Vstup údaje o fyzické osobě
- Výstup seznam ID sledovaných podatelů nebo token z fronty požadavků

#### 1.1.2. DejPodaniPodatele

Funkce vrátí seznam všech podání daného podatele.

- Vstup ID podatele
- Výstup ID všech podání daného podatele

### 1.1.3. DejDetailPodani

Funkce vrací detailní strukturované informace o daném podání, včetně jeho aktuálního stavu, ČJ, zařazení do spisu a SZ, aktuální vyřizující osobu a útvar a seznam dostupných digitálních komponent podání.

- Vstup ID podání
- Výstup základní informace o podání a seznam ID digitálních komponent

### 1.1.4. DejHistoriiPodani

Funkce vrací strukturovanou historii podání ve struktuře časová značka, jméno osoby a popis akce.

- Vstup ID podání
- Výstup strukturovaná historie vyřizování podání ve stylu časová značka, jméno úředníka, provedená akce. Řazení je chronologické dle operací s dokumentem.

### 1.1.5. DejPrilohuPodani

Funkce vrací přílohu podání. Po předání může dle nastavení dojít k vymazání této přílohy, tedy opakovaný dotaz již vrací „neznámá příloha“.

- Vstup ID přílohy podání
- Výstup soubor přílohy podání

### 1.1.6. VysledekPozadavku

Pokud funkce NajdiPodatele v aktuálním datovém skladu nikoho nenajde, zapíše požadavek na dohledání v datech spisové služby, který se jednou za 24 hod zpracuje. Pokud se tento podatel ztotožní v datech spisové služby, předají se všechny informace o jeho podáních a jejich komponentách do datového skladu a přidá se ke sledovaným podatelům, jejichž data se v datovém skladu průběžně aktualizují. Informace o tomto nalezení se zapíše a lze jí touto funkcí ověřit.

- Vstup token z fronty
- Vystup stav zpracování požadavku
  - STAVTOKENU z možností zapsáno, zpracováno a seznam ID podatelů, zpracováno a nic nenalezeno
  - Množina případně nově sledovaných podatelů

### 1.1.7. Příklady použití funkcí pro spisovou službu

Pokud se na portál přihlásí nová fyzická osoba, volá portál funkci NajdiPodatele.

Pokud funkce vrátí ID podatele (podatelů), lze se pomocí funkce DejPodaniPodatele zeptat na seznamy ID její podání a k těmto podáním pak zobrazit detailní informace funkcí DejDetailPodani, popř. historii jejich oběhu na úřadě pomocí funkce DejHistoriiPodani. Tyto údaje jsou vždy aktuální, nejvýše se zpožděním několika minut.

Pokud primární volání funkce NajdiPodatele nenajde v datovém skladu sledovaného podatele, zapíše požadavek do fronty nových sledování. Ten je zpracován spisovou službou v následujícím spuštění procesu zpracování této fronty, k němuž dochází jednou za 24 hod. Portál by měl tedy ohlásit, že zatím nezná žádná podání, ale uživatel to může znovu zkusit za 24 hod. Portál se může dotazovat na stav zpracování požadavku ve frontě pomocí funkce VysledekPozadavku, avšak neměl by tuto funkci volat častěji než 1 za hod.

Jestliže je v datech spisové služby nalezen příslušný podatel, jsou jednorázově ihned doplněny všechny informace o jeho podání a jejich historii i digitálních komponentách do datového skladu. Informace o nalezených podatelích a jejich ID lze získat buď dotazem přes funkci `VysledekPozadavku`, nebo opakovaným postupem od začátku, tedy voláním funkce `NajdiPodatele`.

Jestliže v datech spisové služby není žádný odpovídající podatel nalezen, změní se pouze stav požadavku na „zpracováno a nic nenalezeno“. Podatel není nalezen, pokud nemá evidované podání, nebo pokud údaje evidované ve spisové službě neodpovídají ROB, což může nastat.

## 1.2. Funkce pro volby

Data v datovém skladu se aktualizují vždy následně, když je v modulu `Seznam obyvatel` a volby aktualizován stálý seznam voličů, avšak nejvýše jednou za 24 hod. Aktualizace je prováděna v časovém okně dle nastavení.

### 1.2.1. NajdiVolice

Funkce projde data voličů uložená v datovém skladu. Pokud najde plnou shodu, vrací seznam ID těchto voličů.

- Vstup údaje o fyzické osobě
- Výstup seznam ID voličů

### 1.2.2. DejUdajeVolice

Funkce vrací informace o zařazení voliče do volebního okrsku a informace k tomuto okrsku.

- Vstup ID voliče
- Výstup číslo okrsku, název okrsku, popisné údaje okrsku, např. adresu volební místnosti

### 1.2.3. Příklady použití funkcí pro volby

Pokud se na portál přihlásí fyzická osoba, volá portál funkci `NajdiVolice`. Je-li volič nalezen, lze se pomocí funkce `DejUdajeVolice` dotázat na číslo okrsku, název okrsku, popisné údaje okrsku, např. adresu volební místnosti.

Není-li volič nalezen, pak není zařazen ve stálém seznamu voličů, avšak není možné zjistit důvod nezařazení. Může to být dáno jiným občanstvím nebo tím, že občan dosud nedovršil volební věk s ohledem na aktuální datum.

V datovém skladu nejsou údaje ze zvláštního seznamu voličů ani údaje z dodatku seznamu voličů. Na portál se také nepřenáší informace o voličském průkazu nebo důvodu nevolit, protože tyto nejsou součástí stálého seznamu voličů, ale váží se ke konkrétním volbám.

Protože údaje o zařazení voliče ve volební seznamu se mění velmi málo, nepředpokládá se opakované volání zmíněných funkcí.

## 1.3. Funkce pro poplatky

V datové skladu jsou uloženy údaje o všech plátcích poplatků, které jsou sledované. Úřad tedy určí sledované poplatky (např. poplatek za psa, poplatek za komunální odpad) a informace o splátkách (pohledávkách úřadu) k těmto poplatkům jsou pak přenášeny dávkově do datového skladu. Aktualizace dat je uživatelsky řízena z důvodu dalších návazných procesů, a

proto nelze obecně stanovit interval aktualizace ani aktualizací okno. Typicky však k aktualizaci dochází jednou za 24 hod.

Přenášeny a uloženy jsou vždy aktuální údaje o všech splátkách (zaplacených i nezaplacených pohledávkách úřadu) ke sledovaným poplatkům, tedy pro všechny plátce a jejich zaplacené i nezaplacené závazky.

### 1.3.1. NajdiPlatce

Funkce projde data plátců uložená v datovém skladu. Pokud najde plnou shodu, vrátí seznam ID těchto plátců.

- Vstup údaje o fyzické osobě
- Výstup seznam ID nalezených plátců

### 1.3.2. DejUdajePlatce

Funkce vrátí informace o závazcích daného plátce vůči úřadu, přičemž návratový seznam může být úplný, nebo obsahuje pouze neuhrazené závazky, nebo pouze uhrazené závazky.

- Vstup ID plátce + požadavek (všechny, nezaplacené, zaplacené)
- Výstup seznam závazků daného plátce odpovídající dané podmínce

### 1.3.3. Příklady použití funkcí pro volby

Pokud se na portál přihlásí fyzická osoba, volá portál funkci NajdiPlatce. Je-li plátce nalezen, lze se pomocí funkce DejUdajePlatce dotázat na zaplacené, nezaplacené nebo všechny (bez ohledu na zaplacení) závazky tohoto plátce vůči úřadu vše všech sledovaných kategoriích poplatků (např. poplatek za psa, poplatek za komunální odpad, ...).

Není-li plátce nalezen, pak buď není přihlášen k předmětnému poplatku, nebo jeho evidované údaje neodpovídají ROB, což může nastat.

Je-li v portálu integrovaná platební brána, změní se stav zaplacení závazku plátce až v okamžiku, kdy je informace dostupná z platební brány (seznam VS provedených transakcí) spárována s příslušnou splátkou v modulu Poplatky a následně je tato informace propagována do datového skladu dalším hromadným přenosem. Tato operace probíhá typicky se zpožděním (min. 1 den) a je prováděna pracovníky ekonomického odboru. Od doby provedení úhrady uživatelem portálu do doby této změny (ke které nemusí vůbec dojít, pokud splátka nebude spárována, nebo bude úhrada použita na zaplacení staršího dluhu) si musí případně portál sám udržovat informace o této úhradě navázané na ID splátky, jehož neměnnost při opakovaných aktualizací dat v datovém skladu garantujeme, aby uživatel portálu neprovedl úhradu opakovaně. Informace o provedení úhrady platební bránou portál do datového skladu neposílá a datový sklad tyto údaje neudrží.

## 1.4. On-line verze dokumentace

On-line verze této dokumentace:

- <http://dstest.triada.cz/swagger/index.html?urls.primaryName=API%200.2> (HTML)
- <http://dstest.triada.cz/swagger/v0.2/swagger.json> (JSON)



## 2. Přílohy

# Datový sklad MUNIS API

## Overview

## Paths

**GET /api/v0.2/Poplatky/NajdiPlatce [Externi] Funkce** projde data plátců uložená v datovém skladu. Pokud najde plnou shodu, vrací seznam ID těchto plátců.

### Parameters

Type	Name	Description	Schema
query	<b>cisloOrientacni</b> <i>optional</i>	Číslo orientační	integer (int32)
query	<b>cisloDomovni</b> <i>optional</i>	Číslo domovní	integer (int32)
query	<b>obec</b> <i>optional</i>	Obec	string
query	<b>castObce</b> <i>optional</i>	Část obce	string
query	<b>typCislaDomovniho</b> <i>optional</i>	Typ čísla domovního	integer (int32)
query	<b>ulice</b> <i>optional</i>	Ulice	string
query	<b>prijmeni</b> <i>optional</i>	Příjmení	string
query	<b>jmeno</b> <i>optional</i>	Jméno	string

Type	Name	Description	Schema
query	<b>cisloOrientacniPismeno</b> <i>optional</i>	Číslo orientační písmeno	string
query	<b>datumNarozeni</b> <i>optional</i>	Datum narození	string (date-time)

### Responses

Code	Description	Links
200	OK  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	
apiKey	subjekt-query	

**GET /api/v0.2/Poplatky/DejUdajePlatce [Externi] Funkce** vrací informace o závazcích daného plátce vůči úřadu, přičemž návratový seznam může být úplný, nebo obsahuje pouze neuhrazené závazky, nebo pouze uhrazené závazky.

#### Parameters

Type	Name	Description	Schema
query	<b>platceID</b> <i>optional</i>	ID plátce	string
query	<b>nezaplacene</b> <i>optional</i>	Nezaplacené	boolean
query	<b>zaplacene</b> <i>optional</i>	Zaplacené	boolean

#### Responses

Code	Description	Links
200	OK  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	
apiKey	subjekt-query	

**GET /api/v0.2/SpisovaSluzba/NajdiPodatele [Externi]**  
**Funkce projde data podatelů již uložená v datovém skladu. Pokud najde plnou shodu, vrátí seznam ID těchto podatelů. Pokud nenajde žádná data založí požadavek na doplnění datového skladu a vrátí token tohoto požadavku.**

#### Parameters

Type	Name	Description	Schema
query	<b>cisloOrientacni</b> <i>optional</i>	Číslo orientační	integer (int32)
query	<b>cisloDomovni</b> <i>optional</i>	Číslo domovní	integer (int32)
query	<b>obec</b> <i>optional</i>	Obec	string
query	<b>castObce</b> <i>optional</i>	Část obce	string
query	<b>typCislaDomovniho</b> <i>optional</i>	Typ čísla domovního	integer (int32)
query	<b>ulice</b> <i>optional</i>	Ulice	string

Type	Name	Description	Schema
query	<b>prijmeni</b> <i>optional</i>	Příjmení	string
query	<b>jmeno</b> <i>optional</i>	Jméno	string
query	<b>cisloOrienta cniPismeno</b> <i>optional</i>	Číslo orientační písmeno	string
query	<b>datumNaroz eni</b> <i>optional</i>	Datum narození	string (date-time)

### Responses

Code	Description	Links
200	Success  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	

Type	Name	Scopes
apiKey	subjekt-query	

**GET /api/v0.2/SpisovaSluzba/DejPodaniPodatele [Externi]**  
**Funkce vrací seznam ID všech podání daného podatele.**

#### Parameters

Type	Name	Description	Schema
query	podatelID <i>optional</i>	ID podatele	string

#### Responses

Code	Description	Links
200	Success  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	

Type	Name	Scopes
apiKey	subjekt-query	

**GET /api/v0.2/SpisovaSluzba/DejDetailPodani [Externi]**  
**Funkce vrací detailní strukturované informace o daném podání, včetně jeho aktuálního stavu, ČJ, zařazení do spisu a SZ, aktuální vyřizující osobu a útvar a seznam dostupných digitálních komponent podání.**

#### Parameters

Type	Name	Description	Schema
query	podaniID <i>optional</i>	ID podání	string

#### Responses

Code	Description	Links
200	Success  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	



Type	Name	Scopes
apiKey	apikey-query	
apiKey	subjekt-header	
apiKey	subjekt-query	

**GET /api/v0.2/SpisovaSluzba/DejHistoriiPodani [Externi]**  
**Funkce vrací strukturovanou historii podání ve struktuře časová značka, jméno osoby a popis akce.**

#### Parameters

Type	Name	Description	Schema
query	podaniID <i>optional</i>	ID podání	string

#### Responses

Code	Description	Links
200	Success  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	

Type	Name	Scopes
apiKey	apikey-query	
apiKey	subjekt-header	
apiKey	subjekt-query	

**GET /api/v0.2/SpisovaSluzba/DejPrilohuPodani [Externi]**  
**Funkce vrací přílohu podání. Po předání může dle nastavení dojít k vymazání této přílohy, tedy opakovaný dotaz již vrací „neznámá příloha“.**

#### Parameters

Type	Name	Description	Schema
query	<b>prilohaID</b> <i>optional</i>	ID přílohy	string

#### Responses

Code	Description	Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	

Type	Name	Scopes
apiKey	subjekt-query	

**GET /api/v0.2/SpisovaSluzba/VysledekPozadavku [Externi]**  
Pokud funkce NajdiPodatele v aktuálním datovém skladu nikoho nenajde, zapíše požadavek na dohledání v datech spisové služby, který se jednou za 24 hod zpracuje. Pokud se tento podatele ztotožní v datech spisové služby, předají se všechny informace o jeho podání do datového skladu a přidá se ke sledovaným podatelům, jejichž data se v datovém skladu průběžně aktualizují. Informace o tomto nalezení se zapíše a lze jí touto funkcí ověřit.

#### Parameters

Type	Name	Description	Schema
query	požadavekID <i>optional</i>	ID požadavku	string

#### Responses

Code	Description	Links
200	Success  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links

Code	Description	Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	
apiKey	subjekt-query	

**GET /api/v0.2/Volby/NajdiVolic** [Externí] Funkce projde data voličů uložená v datovém skladu. Pokud najde plnou shodu, vrací seznam ID těchto voličů.

#### Parameters

Type	Name	Description	Schema
query	<b>cisloOrientacni</b> <i>optional</i>	Číslo orientační	integer (int32)
query	<b>cisloDomovni</b> <i>optional</i>	Číslo domovní	integer (int32)
query	<b>obec</b> <i>optional</i>	Obec	string
query	<b>castObce</b> <i>optional</i>	Část obce	string
query	<b>typCislaDomovniho</b> <i>optional</i>	Typ čísla domovního	integer (int32)
query	<b>ulice</b> <i>optional</i>	Ulice	string

Type	Name	Description	Schema
query	<b>prijmeni</b> <i>optional</i>	Příjmení	string
query	<b>jmeno</b> <i>optional</i>	Jméno	string
query	<b>cisloOrientacniPismeno</b> <i>optional</i>	Číslo orientační písmeno	string
query	<b>datumNarozeni</b> <i>optional</i>	Datum narození	string (date-time)

### Responses

Code	Description	Links
200	OK  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	

Type	Name	Scopes
apiKey	subjekt-query	

**GET /api/v0.2/Volby/DejUdajeVolice [Externi] Funkce**  
**vrací informace o zařazení voliče do volebního okrsku**  
**a informace k tomuto okrsku.**

#### Parameters

Type	Name	Description	Schema
query	<b>volicID</b> <i>optional</i>	ID voliče	string

#### Responses

Code	Description	Links
200	OK  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
404	Not Found  <i>Content</i> <b>text/plain</b> <b>application/json</b> <b>text/json</b>	No Links
400	Bad request	No Links
401	Unauthorized	No Links
403	Forbidden	No Links

Type	Name	Scopes
apiKey	apikey-header	
apiKey	apikey-query	
apiKey	subjekt-header	

Type	Name	Scopes
apiKey	subjekt-query	

# Components

## Schemas

### ProblemDetails

#### Properties

Name	Description	Schema
type <i>optional</i>	<i>nullable</i>	string
title <i>optional</i>	<i>nullable</i>	string
status <i>optional</i>	<i>nullable</i>	integer (int32)
detail <i>optional</i>	<i>nullable</i>	string
instance <i>optional</i>	<i>nullable</i>	string

### PlatceUdajeDto

#### Properties

Name	Description	Schema
poplCis <i>optional</i>		integer (int32)
poplKat <i>optional</i>		integer (int32)
nazevPoplatek <i>optional</i>	<i>nullable</i> <b>Maximum Length: 40</b>	string
nazevSplatka <i>optional</i>	<i>nullable</i> <b>Maximum Length: 20</b>	string
splatne <i>optional</i>		string (date-time)

Name	Description	Schema
suma <i>optional</i>		number (double)
vs <i>optional</i>	<i>nullable</i> <b>Maximum Length: 10</b>	string
specSymb <i>optional</i>	<i>nullable</i> <b>Maximum Length: 10</b>	string
dU1 <i>optional</i>	<i>nullable</i> <b>Maximum Length: 10</b>	string
dU2 <i>optional</i>	<i>nullable</i> <b>Maximum Length: 40</b>	string
ndU1 <i>optional</i>	<i>nullable</i> <b>Maximum Length: 20</b>	string
ndU2 <i>optional</i>	<i>nullable</i> <b>Maximum Length: 20</b>	string
pocet <i>optional</i>		integer (int32)
datumVznik <i>optional</i>		string (date-time)
kdoPredepsal <i>optional</i>		integer (int32)
sumaZaplac <i>optional</i>		number (double)
dokladZaplac <i>optional</i>	<i>nullable</i> <b>Maximum Length: 9</b>	string
datumZaplac <i>optional</i>		string (date-time)
kdoPotvrdil <i>optional</i>		integer (int32)



## PodateleDto

### Properties

Name	Description	Schema
podateleIDs <i>optional</i>	<i>nullable</i> <b>Maximum Length: 50</b>	< string > array
tokenPozadavk u <i>optional</i>	<i>nullable</i> <b>Maximum Length: 50</b>	string

## FormaEntity

## StavDokumentu

## PodaniDto

### Properties

Name	Description	Schema
datum <i>optional</i>		string (date-time)
forma <i>optional</i>		<a href="#">FormaEntity</a>
vec <i>optional</i>	<i>nullable</i> <b>Maximum Length: 512</b>	string
cj <i>optional</i>	<i>nullable</i> <b>Maximum Length: 50</b>	string
sz <i>optional</i>	<i>nullable</i> <b>Maximum Length: 50</b>	string
stav <i>optional</i>		<a href="#">StavDokumentu</a>
digitalniKomponentyIDs <i>optional</i>	<i>nullable</i>	< string > array

## HistoriePodaniDto

### Properties

Name	Description	Schema
poradi <i>optional</i>		integer (int32)
cas <i>optional</i>		string (date-time)
provedl <i>optional</i>	<i>nullable</i> <b>Maximum Length: 96</b>	string
popisAkce <i>optional</i>	<i>nullable</i> <b>Maximum Length: 1024</b>	string

## StavPozadavku

### VysledekPozadavkuDto

#### Properties

Name	Description	Schema
stavPozadavku <i>optional</i>		<a href="#">StavPozadavku</a>
podateleGuids <i>optional</i>	<i>nullable</i>	< string > array

## VolicUdajeDto

#### Properties

Name	Description	Schema
cisloOkrsku <i>optional</i>		integer (int32)
nazevOkrsku <i>optional</i>	<i>nullable</i> <b>Maximum Length: 96</b>	string
popis <i>optional</i>	<i>nullable</i> <b>Maximum Length: 1024</b>	string