Mesto Košice

Príloha č.3

Konsolidácia elektronických služieb mesta Košice (KESK) - Konvencia vytvárania REST-API

REST API, OPEN API 3, SWAGGER

|  |  |
| --- | --- |
| Zmluva |  |
| Verzia dokumentu | 0.2 |
| Autor dokumentu | Peter Stavný |
| Vlastník dokumentu | Mesto Košice |
| Dátum vydania |  |

Denník zmien:

| Verzia | Dátum | Revidoval | Popis | Označenie zmien |
| --- | --- | --- | --- | --- |
| 0.1 | 01.01.2023 | Peter Stavný | Iniciálna verzia dokumentu | Bez revízie |
| 0.2 | 10.01.2023 | Peter Stavný | Autorizácia aplikácii | Bez revízie |

Obsah

[1 Slovník pojmov 3](#_Toc158213893)

[2 Zoznam obrázkov 3](#_Toc158213894)

[3 Zoznam tabuliek 3](#_Toc158213895)

[4 Návrh riešenia 5](#_Toc158213896)

[5 Pravidlá 5](#_Toc158213897)

[5.1 Všeobecné pravidlá 5](#_Toc158213898)

[5.2 Pravidlá pre URI 5](#_Toc158213899)

[5.3 Pravidlá pre verejnú časť API GW 6](#_Toc158213900)

[5.4 Pravidlá pre privátnu časť API rozhraní 7](#_Toc158213901)

[5.5 Pravidlá pre verziovanie API 7](#_Toc158213902)

[5.6 Hlavička požiadavky 7](#_Toc158213903)

[5.6.1 Autorizácia osôb 8](#_Toc158213904)

[5.6.2 Autorizácia aplikácii 8](#_Toc158213905)

[5.6.3 Identifikačné údaje aplikácie 8](#_Toc158213906)

[5.7 Dopytové metódy HTTP 9](#_Toc158213907)

[5.7.1 GET metóda 9](#_Toc158213908)

[5.7.2 POST metóda 9](#_Toc158213909)

[5.7.3 PUT metóda 10](#_Toc158213910)

[5.7.4 DELETE metóda 10](#_Toc158213911)

[5.8 Spracovanie chýb (Error Handling) 11](#_Toc158213912)

[5.8.1 Úspešná operácia 11](#_Toc158213913)

[5.9 Neúspešná operácia 11](#_Toc158213914)

[5.10 OpenAPI 3.0 (Swagger) 12](#_Toc158213915)

[5.10.1 Definícia security schémy a typu autorizácie 13](#_Toc158213916)

[5.10.2 Definícia serverov príklad 13](#_Toc158213917)

[5.10.3 Verzionovanie implementácie kontroléra 14](#_Toc158213918)

[6 Skratky a značky 15](#_Toc158213919)

# Slovník pojmov

|  |  |  |
| --- | --- | --- |
| Pojem | Vysvetlenie | |
| **Aplikácia** | Samostatne prevádzkovateľný komponent informačného systému implementujúci aplikačné služby. | |
| **Application ID** | Identifikátor, ktorý jednoznačne identifikuje práve jednu aplikáciu v procese riadenie prístupov cez API rozhrania. Vzniká v procese registrácie aplikácie oprávnenou osobou pre používanie služieb a rozhraní publikovaných na Azure AD B2C. | |
| **Tenant** | Entita s nezávislými informáciami a dátami. Používa služby a zdroje pridelené poskytovateľom služby. Riadenie tenantov rieši služba. | |
| **Autentifikácia aplikácie** | Aplikácia identifikovaná cez Application ID sa autentifikuje voči určenej autentifikačnej službe určeným spôsobom a pomocou vopred pridelených *credentials*. Po overení identity a oprávnení môže overená aplikácia využívať zdroje a služby (technické prepojenie). | |
| **Riadenie oprávnení** | Opakovane použiteľné služby, ktoré umožňujú prístup k svojim chráneným zdrojom bezpečným spôsobom sa riadia oprávneniami nastavenými administrátorom príslušného poskytovateľa služby pre konkrétnu skupinu oprávnení a konkrétny chránený zdroj. | |
| **Dokument** | Reprezentuje jednu vec, jednu entitu (alebo logický agregát).Ide podstatné meno v jednotnom čísle, ale pokiaľ je naň referované cez kolekciu, uvádza sa názov v množnom čísle.  Príklady URI:  /applications/6ba7b810-9dad-11d1-80b4-00c04fd430c8  /navody/ako-vybavit-obciansky | |
| **Kolekcia** | **P**redstavuje množinu (viacerých) zdrojov. Ide o podstatné meno v množnom čísle. Príklady URI:  /applications  /applications/6ba7b810-9dad-11d1-80b4-00c04fd430c8/platforms | |
| **Kontrolér** | Predstavuje generický zdroj reprezentujúci „vykonateľnú akciu“. Ide o sloveso reprezentujúce danú akciu.  Príklady URI:  /applications/6ba7b810-9dad-11d1-80b4-00c04fd430c8/feedback/ |
|  |  | |

# Zoznam obrázkov

**No table of figures entries found.**

# Zoznam tabuliek

[Tabuľka 1 Všeobecné pravidlá pre API. 5](#_Toc158213759)

[Tabuľka 2 Pravidlá pre URI identifikátori. 6](#_Toc158213760)

[Tabuľka 3 Pravidlá pre publikovanie na verejnom API rozhraní. 7](#_Toc158213761)

[Tabuľka 4 Pravidlá pre publikovanie na privátne API rozhranie. 7](#_Toc158213762)

[Tabuľka 5 Identifikačné údaje aplikácie. 8](#_Toc158213763)

[Tabuľka 6 Povolené hodnoty pre X-APP-PLATFORM. 9](#_Toc158213764)

[Tabuľka 7 Pravidlá pre dopytovú metódu GET 9](#_Toc158213765)

[Tabuľka 8 Pravidlá pre dopytovú metódu POST 10](#_Toc158213766)

[Tabuľka 9 Pravidlá pre dopytovú metódu PUT 10](#_Toc158213767)

[Tabuľka 10 Pravidlá pre dopytovú metódu DELETE 11](#_Toc158213768)

[Tabuľka 11 Návratové kódy pre úspešné operácie (HTTP status codes) 11](#_Toc158213769)

[Tabuľka 12 Návratové kódy pre neúspešné operácie (HTTP status codes) 12](#_Toc158213770)

[Tabuľka 13 Pravidlá pre popis API podľa štandardu OpenAPI 3.0 13](#_Toc158213771)

# Návrh riešenia

Tento dokument popisuje konvencie požadované pri vytváraní REST volaní (paths, query, headers) a jeho následnom vystavení na verejnom (PUBLIC) a privátnom (PRIVATE) API rozhraní.

Východiskom sú dva kľúčové strategické dokumenty informatizácie, a to Strategická priorita: Integrácia a orchestrácia a Referenčná architektúra integrovaného informačného systému verejnej správy, zadefinovali pre všetky agendové systémy, poskytujúce elektronické služby verejnej správy, a ich správcov, povinnosť publikovať (formou web služieb) služby pre spracovanie elektronických podaní, a zároveň všetky súvisiace (pomocné) služby pre úspešné vyplnenie a prípravu podania (napr. doťahovanie údajov, validácia vstupov) do multikanálového prostredia, ktorá je súčasťou modulu procesnej integrácie a integráciu údajov definovaného zákonom 305/2013 Z.z. o eGovernmente.

Celkový návrh je orientovaný na potreby vystavenia API v OAS3 špecifikácii ekosystéme mesta Košíc. Ako základ pre špecifikáciu sa berú **Pravidlá publikovania elektronických služieb do multikanálového prostredia verejnej správy** (<https://www.mirri.gov.sk/wp-content/uploads/2018/10/Pravidla_Publikovania_Sluzieb_v1_0-1.pdf>).

# Pravidlá

## Všeobecné pravidlá

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API00 | REST/Restful ako jediný podporovaný prístup implementácie web služieb. |
| API01 | Predpísaný štandard pre popis API je OpenAPI3.0 (alebo vyšší). |
| API02 | Na modelovanie obsahu dostupného cez API sú využívané Chránené zdroje aj Nechránené zdroje. |
| API03 | Každý chránený zdroj je identifikovaný pomocou svojho jednoznačného URI. |
| API04 | Všeobecný predpis pre API je: /{version}/{component-name}/{hierarchy-identifiers} |

Tabuľka 1 Všeobecné pravidlá pre API.

## Pravidlá pre URI

Pre definície URI platia tieto pravidlá:

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API05 | URI je vždy v anglickom jazyku. |
| API06 | URI je vždy malými písmenami bez diakritiky. |
| API07 | Viacslovné názvy v URI sú spojené pomlčkami (*kebab-case*). |
| API08 | URI nesmie používať podčiarkovník (underscore) a medzery (space). |
| API09 | Na modelovanie variabilných častí URI sú využívané URI šablóny ([RFC 6570](https://tools.ietf.org/html/rfc6570)).  Príklad:  .../platforms/{platformId}/configurations |
| API10 | URI je hierarchické. |
| API11 | URI musí obsahovať verziu API.  Príklad:  /v1/applications/6ba7b810-9dad-11d1-80b4-00c04fd430c8 |
| API12 | Návrh URI pre API musí zohľadňovať biznisovú funkcionalitu. |
| API13 | Každé URI musí mať označenie verzie v tvare v1, v2, v3,... vN, kde N je prirodzené číslo. |
| API14 | Označenie názvu viacslovného komponentu je jednoznačné a používa *kebab-case* konvenciu. |

Tabuľka 2 Pravidlá pre URI identifikátori.

## Pravidlá pre verejnú časť API GW

Verejné API rozhranie smeruje API volania zvyčajne do privátnej zóny, kde ich nasledovne smeruje k chráneným zdrojom – mikroslužbám, službám a monolitným systémom. API sú zvyčajne vytvárané na doménovom modely, napr. Registračná služba, Konfiguračná služba, Služba Profilu Košičana alebo EKR.

Príklady verejných URI :

* **Profil Košicana mikroslužba**: /v1/kk/profiles
* **Konfiguračná mikroslužba**: /v1/kk/configurations/profiles/6ba7b810-9dad-11d1-80b4-00c04fd430c8
* **EKR sprístupnené cez REST API rozhranie**: /v1/messagebox-egov /messages/6ba7b810-9dad-11d1-80b4-00c04fd430c8

Význam jednotlivých častí všeobecného predpisu /{version}/{component-name}/{hierarchy-identifiers} pre API je:

* **version –** verzia API. *Príklad* ***v1***
* **component-name** –názov komponentu, ktorý musí byť jednoznačný medzi ostatnými komponentami, napríklad: /**profiles**, /**calendar**, /**proceeding**,… podľa tohto názvu je nastavené smerovanie na API GW (routing), viacslovné alebo zložené názvy komponentov, ktoré označujú dvojicu modul a komponent, sú spojené pomlčkami.
* **hierarchy-identifiers** - hierarchický dizajn identifikátorov zdroja, napríklad*:* /processes/*6ba7b810-9dad-11d1-80b4-00c04fd430c8*

| **Pravidlo** | **Popis pravidla** |
| --- | --- |
| **API15** | Na verejnom API rozhraní URI musí obsahovať *business-namespace*. |
| **API16** | Na verejnom API rozhraní URI nesmie obsahovať technický názov služby alebo modulu. |
| **API17** | Na verejnom API rozhraní je povinný hierarchický dizajn časti *hierarchy-identifiers*. |

Tabuľka 3 Pravidlá pre publikovanie na verejnom API rozhraní.

## Pravidlá pre privátnu časť API rozhraní

Privátne API rozhranie smeruje API volania k zdrojom – mikroslužbám, službám a monolitným systémom. API sú zvyčajne vytvárané na doménovom modely, napr. Registračná služba, Konfiguračná služba, Služba Profilu Košičana alebo EKR.

API dostupné z privátneho (PRIVATE) API rozhrania sú navrhované predovšetkým ako administračné (CRUD) resp. biznisové. Ide o služby, ktoré nie sú dostupné verejne z internetu ale iba zo privátnej siete alebo zabezpečeného spojenia.

Rozloženie biznisových URI na privátnom API rozhraní zrkadlí API návrh na verejnom API rozhraní. API na privátnom API rozhraní povoľuje vloženie doplňujúceho identifikátor chráneného zdroja. Rozloženie pre /{version}/{business-name}/{source-identifier}/{hierarchy-identifier} je nasledovné:

* **version** – verzia API, napríklad /v1
* **business-name** – biznisový názov chráneného zdroja, nie názov modulu, napríklad: /profile-ad
* **source-identifier –** názov chráneného zdroja. Napríklad pri moduloch zložených z viacerých servisov, resp. mikroslužieb, ide o ich názov. Príklad: /profile-ad/identity
* **hierarchy-identifiers** - hierarchický dizajn identifikátorov chráneného zdroja, napríklad/identities/6ba7b810-9dad-11d1-80b4-00c04fd430c8

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API18 | Na privátnom API rozhraní pre biznisovo chránené zdroje URI musí obsahovať *business-namespace***.** |
| API19 | Na privátnej API rozhraní je povinný hierarchický dizajn časti *hierarchy-identifiers*. |

Tabuľka 4 Pravidlá pre publikovanie na privátne API rozhranie.

## Pravidlá pre verziovanie API

Verzionovanie API zabezpečiť pomocou zakomponovania označenia verzie do URI v rámci elementu API na najvyššej úrovni v podobe skratky verzia 1 – v1. Napríklad:  
https://api.kosice.sk/v1/applications/6ba7b810-9dad-11d1-80b4-00c04fd430c8/.

Pre zachovanie jednoduchosti pri používaní, je vyžadované aby API bolo verziované ako celok, nie po jeho jednotlivých častiach.

V prípade spätne nekompatibilných (major) zmien je potrebné tieto zmeny urobiť v rámci novej verzie API, pričom nová verzia nesie typicky v označení inkrementované celé číslo (z v1 sa stane v2 a pod.).

Zároveň je potrebné pre nevyhnutnú dobu podporovať a udržiavať funkčné obe verzie API (starú aj novú, ktorá vznikla po spätne nekompatibilnej zmene.

Jasne komunikovať, do kedy bude stará verzia udržiavaná a dokedy je potrebné, aby používatelia API migrovali na novšiu verziu.

## Hlavička požiadavky

Jednotlivé typ autentizácie osôb sú popísane v dokument [**MK-KKaKES-DNR-PR-02-Autentifikacia.docx**](MK-KKaKES-DNR-PR-02-Autentifikacia.docx)

### Autorizácia osôb

Samotný autentifikačný prostriedok (JWT token - <MK-KKaKES-DNR-PR-02-Autentifikacia.docx>) bude naďalej využívať štandardný atribút HTTP hlavičky **Authorization**.

**HTTP/S HEADER**

Authorization: BEARER <token>

### Autorizácia aplikácii

Pre potreby rozhodovania API rozhrania sa do hlavičky HTTP volania zavádza povinný **X-APP-ID** pre prenos **applicationId** vygenerovaného pri registrácii aplikácie v systéme Azure Microsoft Entre.

Ako autorizačný prostriedok je určený JWT access token aplikácie [MK-KKaKES-DNR-PR-05-Autentifikacia-Aplikacii.docx](file:///Users/peter.stavny/Documents/Bussines/In-Watts/MADSolution/DNR/MK-KKaKES-DNR-PR-05-Autentifikacia-Aplikacii.docx).

Práve tento, ID token bude slúžiť:

* pre autorizáciu verejného REST API bez autorizácie osôb (ak nejde o Open Data endpointy)
* pre autorizáciu privátneho REST API, aj keď sa tam nachádza autorizácia

Pre túto autorizáciu sa zavádza nový atribút HTTP hlavičky **X-KK-APP-AUTH**.

**HTTP/S HEADER**

X-APP-ID: <UUID>

X-KK-APP-AUTH: BEARER <token>

### Identifikačné údaje aplikácie

|  |  |  |
| --- | --- | --- |
| # | Hodnota | Poznámka |
| 1 | correlationId | Jedinečný identifikátor požiadavky resp. série požiadaviek, UUID v string formáte, [RFC 4122](https://datatracker.ietf.org/doc/html/rfc4122)  *Príklad* : 6ba7b810-9dad-11d1-80b4-00c04fd430c8 |
| 2 | X-APP-ID | Jedinečný identifikátor registrovanej aplikácie v Microsoft Entra, UUID v string formáte, [RFC 4122](https://datatracker.ietf.org/doc/html/rfc4122)  *Príklad* : 6ba7b810-9dad-11d1-80b4-00c04fd430c8 |
| 3 | X-APP-VERSION | Verzia aplikácie podľa *Semantic versioning* 2.0 (<https://semver.org>)  *Príklad* : 1.0.0 |
| 4 | X-APP-PLATFORM | Aplikačná platforma. Hodnota môže nadobúdať hodnoty z číselníka [X\_APP\_PLATFORM](#_X_APP_PLATFORM) |
| 5 | X-DEVICE-ID | Povinný atribút len pre mobilné platformy. UUID v string formáte, [RFC 4122](https://datatracker.ietf.org/doc/html/rfc4122)  *Príklad* : 6ba7b810-9dad-11d1-80b4-00c04fd430c7 |

Tabuľka 5 Identifikačné údaje aplikácie.

**HTTP/S HEADER**

correlationId: 6ba7b810-9dad-11d1-80b4-00c04fd430c8

X-APP-ID: 6ba7b810-9dad-11d1-80b4-00c04fd430c7

X-APP-VERSION: 1.0.0

X-APP-PLATFORM: ios

X-DEVICE-ID: 6ba7b810-9dad-11d1-80b4-00c04fd430c7

##### X-APP-PLATFORM

|  |  |  |
| --- | --- | --- |
| # | Hodnota | Poznámka |
| 1 | ios | iOS aplikácia. |
| 2 | android | Android aplikácia . |
| 3 | web | Web aplikácia. |
| 4 | native | Natívna desktopová alebo embedded aplikácia. |
| 5 | service | Backend servis. |
| 6 | spa | Single point application - web |

Tabuľka 6 Povolené hodnoty pre X-APP-PLATFORM.

## Dopytové metódy HTTP

Pre dopyty je potrebné využívať HTTP dopytové metódy[[1]](#footnote-1) pričom ak tieto nestačia, využíva sa *Kontrolér:*

### GET metóda

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API20 | GET metóda musí byť využívaná na získanie reprezentácie webového zdroja, teda získanie Dokumentu alebo získanie Kolekcie. |
| API21 | GET metóda dotazu nemá telo. Odpoveď musí v tele obsahovať reprezentáciu zdroja. GET dotaz nesmie mať „vedľajšie“ účinky na daný zdroj - iné ako súvisiace so získaním zdroja. Ani pri viacnásobnom volaní a jej výsledok je potenciálne kešovateľný. |
| API22 | GET metóda dotazu môže obsahovať parametre, ktoré sa využívajú na filtrovanie, alebo stránkovanie v prípade kolekcií. Odporúčame využívať jednotný štýl cez celé API. Parametre dotazu môžu slúžiť aj na filtrovanie, ktoré atribúty budú (alebo nebudú) poskytované v odpovedi (reprezentácia zdroja). |

Tabuľka 7 Pravidlá pre dopytovú metódu GET

### POST metóda

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API23 | POST sa využíva na posielanie informácií z klienta smerom na server. |
| API24 | POST dotaz musí obsahovať telo. Odpoveď naň môže obsahovať telo. |
| API25 | POST dotaz môže mať „vedľajšie účinky“, aj pri viacnásobnom volaní, môže byť potencionálne kešovateľný. |
| API26 | POST dotaz je využívaný na vloženie nového webového zdroja do kolekcie. |
| API27 | POST je metódou, ktorá sa používa aj pri práci so zdrojom typu kontrolér. |
| API28 | POST dotaz môže byť využitý aj na filtrovanie a stránkovanie kolekcií, pokiaľ nie je možné, alebo je nepraktické využiť parametre dotazu (využíva sa v takom prípade podriadený kontrolér, pretože priame využitie kolekcie by viedlo ku kolízii s významom metódy pre pridávanie nového zdroja do kolekcie). |
| API29 | POST metóda sa nesmie využívať na aktualizáciu, resp. vymazanie webového zdroja, na to je potrebné využívať metódy PUT a DELETE. |

Tabuľka 8 Pravidlá pre dopytovú metódu POST

### PUT metóda

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API30 | PUT metóda je využívaná na vloženie, alebo aktualizáciu jedného dokumentu, alebo nahradenie celej kolekcie. |
| API31 | Vkladanie nového dokumentu pomocou PUT metódy implikuje, že klient je zároveň schopný navrhnúť URI pre nový zdroj (resp. nejakú identifikáciu). Ak to nedokáže, mal by vkladať nový webový zdroj cez POST metódu. |
| API32 | Pri vytváraní, server odpovedá pomocou **201 Created** s hlavičkou umiestnenia, kde je finálne URI (ak je iné, ako to čo bolo navrhnuté klientom). Odpoveď v tele môže niesť nejaký obsah. V prípade, že hlavičku s umiestnením neobsahuje, implicitne sa predpokladá, že URI dotazu je umiestnením novovzniknutého zdroja. |
| API33 | Pri aktualizácii, server odpovedá buď s kódom 200 OK (pričom vracia aktuálny stav entity), alebo 204 No Content (v tele odpovede sa nevracia žiadny obsah, klient musí v prípade potreby znovu načítať webový zdroj), alebo **303 See Other** s presmerovaním na zdroj s aktuálnym obsahom (čo je rovnaké URI ako bolo využité pri PUT metóde, ale je potrebné znovu načítať cez GET). |
| API34 | To, akým spôsobom očakáva klient komunikáciu úspešnej operácie, môže byť uvedené pomocou atribútu PUT dotazu. |

Tabuľka 9 Pravidlá pre dopytovú metódu PUT

### DELETE metóda

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API35 | DELETE metóda dotazu sa využíva na odstránenie zdroja. Výsledkom je, že dané URI už viac nie je platné a dotazy budú vracať **410 Gone**, alebo **404 Not Found**. |
| API36 | DELETE metóda nesmie byť využívaná na akékoľvek „predbežné“ vymazanie, teda operáciu, ktorej výsledok je možné cez API vrátiť, resp. odvolať. |
| API37 | Ak bol výsledok operácie DELETE úspešný, žiadny webový zdroj sa nebude dať pod daným URI ďalej využívať. |
| API38 | Implementácia DELETE nemusí striktne záznam fyzicky odstrániť zo svojho úložiska. Stav takto spracovaných zdrojov musí pre pôvodné URI produkovať odpovede s kódom **410 Gone**, alebo **404 Not Found**. |

Tabuľka 10 Pravidlá pre dopytovú metódu DELETE

## Spracovanie chýb (Error Handling)

Výsledky HTTP dopytov sú označované štandardnými kódmi (HTTP status codes).

### Úspešná operácia

V prípade úspechu operácie platia tieto pravidlá:

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API39 | Na komunikáciu úspešnej operácie využívajú kódy v rozsahu 2xx. |
| API40 | V prípade neúspechu operácie nie je povolené používať kódy z rozsahu 2xx. |
| API41 | **200 OK** komunikuje úspešný výsledok operácie v prípade, že výsledok obsahuje nejaké „telo“. Ak tomu tak nie je, potrebné použiť **204 No Content** |
| API42 | **201 Created** je kódom odpovede v prípade vytvárania nového zdroja, pričom obsahuje hlavičku *Location* s URI adresou novovzniknutého webového zdroja. V prípade, že hlavičku s umiestnením neobsahuje, implicitne sa predpokladá, že URI dotazu je umiestnením novovzniknutého zdroja. |
| API43 | **202 Accepted** je najvhodnejším kódom odpovede v prípade asynchrónnych operácií a výhradne pri operáciách nad kontrolérmi. Nesmie obsahovať *Location* hlavičku. |
| API44 | **301 Moved Permanently** sa použije na indikáciou, že zdroj sa presunul na iné URI. |
| API45 | **303 See Other** môže byť použitý po spustení asynchrónnej operácie na presmerovanie na URI, kde je možné vyzdvihnúť výsledok (alebo stav) danej operácie. Klient sa môže rozhodnúť, kedy bude presmerovanie nasledovať (napr. pri snahe nezahlcovať komunikačné pásmo). |
| API46 | **304 Not Modified** kód je indikáciou, že zdroj nebol zmenený a môže byť využívaná nakešovaná reprezentácia. |

Tabuľka 11 Návratové kódy pre úspešné operácie (HTTP status codes)

## Neúspešná operácia

V prípade neúspechu vyvolanej operácie sú výsledkom chybové odpovede. Chybové odpovede okrem uvádzaných štandardných kódov, budú obsahovať aj správy vysvetľujúce čo to znamená pre klienta.

Kódy v rozsahu **4xx** indikujú chybu na strane klienta (nesprávne volanie) a komunikuje sa očakávanie, že klient svoj dotaz upraví tak, aby operáciu bolo možné vykonať (pričom poslanie znovu toho istého dotazu nedáva význam).

Kódy v rozsahu **5xx** indikujú chybu na strane servera.

|  |  |
| --- | --- |
| Pravidlo | Popis pravidla |
| API47 | **400 Bad Request** je najuniverzálnejším kódom komunikujúcim neúspešnú operáciu z dôvodu zlého dotazu zo strany klienta. Nižšie uvádzané špecifickejšie kódy by mali byť využívané len v prípadoch, kde vyplývajú z použitých špecifikácií, alebo štandardov. |
| API48 | **401 Unauthorized** by sa mal využívať v prípade, že v dotaze chýbajú autorizačné informácie. |
| API49 | **403 Forbidden** je potrebné použiť v prípade, ak zdroj nie je pre daného klienta (dotaz autorizačné informácie obsahuje) dostupný z bezpečnostných dôvodov (klient nemá naň z nejakého dôvodu prístup). |
| API50 | **404 Not Found** by sa mal využívať v prípade, že na danom URI neexistuje žiadny webový zdroj. |
| API51 | **406 Not Acceptable** indikuje, že klient vyžaduje v odpovedi „*content-type*“, ktorý server nemôže (alebo nevie) produkovať. |
| API52 | **409 Conflict** indikuje nekonzistentnú aktualizáciu (napr. kvôli paralelnému vyvolaniu vloženia, alebo aktualizácie zdroja). |
| API53 | **410 Gone** indikuje, že zdroj na danom URI už nie je dostupný (a ani v budúcnosti dostupný nebude – príklad vymazaného zdroja pomocou DELETE metódy). |
| API54 | **415 Unsupported Media Type** by sa mal využívať v prípade, že klient posla dotaz v nepodporovanej reprezentácii (*content-type*). |
| API55 | **422 Unprocessable Entity** môže byť využitý v prípade ak formálne je dotaz v poriadku, ale validácia našla chybu na inom mieste (Napríklad využíva WebDAV, RFC 4918). |
| API56 | **429 Too Many Requests** je nutné využívať v prípade obmedzovania počtu volaní na webový zdroj. |
| API57 | Je odporúčané využívať univerzálny kód 500 na komunikáciu všetkých chýb, okrem tých, ktorým z použitých protokolov, alebo štandardov vyplývajú iné kódy z rozsahu 5xx. |

Tabuľka 12 Návratové kódy pre neúspešné operácie (HTTP status codes)

## OpenAPI 3.0 (Swagger)

| **Pravidlo** | **Popis pravidla** |
| --- | --- |
| **API58** | Pre všetky dotazy a odpovede z web služieb využívať štandard Unicode, ak nie je určené inak, používa sa UTF-8. |
| **API59** | Telá odpovedí dotazov a odpovedí sú vo špecifikovanom formáte, teda ak ide o JSON, *content-type* sa uvádza ako application/json, pre XML sa uvádza application/xml. |
| **API60** | Každý vstupný a výstupný parameter je popísaný (description) a má svoj príklad (*example*). |
| **API61** | Každá požiadavka (request) má krátky popis (*summary*) a popis (*description*) a príklad. |
| **API62** | Každá požiadavka má tag definovaný opisnou formou podľa *business-name***.** Vnorené tagy musia obsahovať aj rodičovský tag ako prefix s pomlčkou. |
| **API63** | Každá správna odpoveď má popis a príklad. |
| **API64** | Každá chybová odpoveď má popis a príklad. |
| **API65** | Odpovede zodpovedajú HTTP chybovým kódom podľa HTTP špecifikácie. |
| **API66** | API definície musia mať ako celok, respektíve každé volanie vlastný, zadefinovaný server, kde je API nasadené a možné otestovať volanie. |
| **API67** | Atribúty s „prázdnou“ hodnotou (**null**) môžu byť pri dotazoch, alebo odpovediach z web služby vypustené, toto chovanie však musí byť zdokumentované v špecifikácii služby. |
| **API68** | Na poradí atribútov v JSON objekte nezáleží. |
| **API69** | Všade, kde to je možné, je podporovaný ľahko čitateľný formát a logické zoradenie JSON atribútov. |

Tabuľka 13 Pravidlá pre popis API podľa štandardu OpenAPI 3.0

### Definícia security schémy a typu autorizácie

security:

- bearerAuth: []

securitySchemes:

bearerAuth: # arbitrary name for the security scheme

type: http

scheme: bearer

bearerFormat: JWT # optional, arbitrary value for documentation purposes

### Definícia serverov príklad

description: TEST API NORRIS

url: '{protocol}://api.test.kosice.zone:{port}'

variables:

port:

enum: ['8443']

default: '8443'

protocol:

enum: ['http', 'https']

default: https

- description: PROD API NORRIS

url: https://api.kosice.sk

### Verzionovanie implementácie kontroléra

Verzionovanie API zabezpečiť pomocou SwaggerHub nástroja, označenia verzie podľa *Semantic versioning* 2.0 - <https://semver.org>, napríklad: 1.0.0

V prípade práce na novšej verzii ako je reálne publikovaná a nasadená na API GW, je nutné vytvoriť a zdvihnúť verziu, avšak neoznačovať ako default.

# Skratky a značky

V tomto dokumente sú použité nasledujúce skratky a značky:

| Skratka / Pojem | Vysvetlenie / Popis |
| --- | --- |
| AAA | Autentifikácia, autorizácia a audit. |
| Admin GUI | Rozhranie pre správu a konfiguráciu aplikácií, spravidla pre rolu Administrátor (Admin). |
| API | **Application Platform Interface**, rozhranie aplikačnej platformy spravidla na komunikáciu medzi systémami. |
| API First | Prístup, ktorý určuje spôsob návrhu a vývoja aplikačného softvéru, tak aby API vznikalo ešte pred samotnou implementáciou. |
| API GW | Verejná integračná platforma (API Gateway). |
| AS | Aplikačná služba (AS) je služba, ktorá sprístupňuje automatizované správanie aplikačnej funkcie, resp. aplikačného komponentu, ktorý je súčasťou informačného systému verejnej správy. |
| AS–IS | Aktuálny stav bez realizácie projektu. |
| BOK | Bezpečnostný osobný kód. |
| BPaaS | Business-process-as-a-Service. |
| BPMN | Business Process Model and Notation. |
| CEP | Centrálna elektronická podateľňa. |
| CI/CD | Kontinuálna integrácia a dodávka. |
| CIP | Centrálna integračná platforma. |
| CISSP | Certified Information Systems Security Professional. Certifikát pre oblasť informačnej bezpečnosti. |
| CMS | Systém na správu obsahu (Content Management System). |
| CPU | Centrálna procesorová jednotka. |
| CSRÚ | Centrálna Správa Referenčných Údajov - informačný systém. |
| CÚD | Centrálne úradné doručovanie - rozšírenie modulu MED. |
| DB | Databáza (Database). |
| DevOps | Prístup riadenia založený na spolupráci vývoja a prevádzky. |
| DNR | Detailný návrh riešenia. |
| DOS | Pokus útočníka zamedziť používateľom prístup k službám počítača alebo k sieti. Komunikácia medzi používateľmi je natoľko preťažená, že nemôže adekvátne prebiehať. Napadnutý počítač je zvyčajne potrebné reštartovať, aby mohol poskytovať plnohodnotné služby (Denial of Service).  Cieľom sa stávajú najčastejšie web servery a účelom útoku je vyradiť ich z činnosti. |
| EA | Enterprise architektúra. |
| eDesk | Modul elektronických komunikačných stránok ÚPVS. |
| eForm | Modul elektronických formulárov ÚPVS. |
| eGOV | eGovernment. |
| eID | Elektronická identifikačná karta. |
| EKR | Externé komunikačné rozhranie modulu eDesk. |
| ESB | Podniková zbernica služieb (Enterprise Service Bus). |
| FAQ | Často kladené otázky (Frequently Asked Questions). |
| FO | Fyzická osoba. |
| G-Cloud | Vládny cloud (Government cloud). |
| G2G | Government To Government. |
| Git | Systém slúžiaci na riadenie revízií. |
| GovNET | Zabezpečená štátna sieť spravovaná Národnou agentúrou pre sieťové a elektronické systémy. |
| GUI | Grafické používateľské rozhranie (Graphical User Interface). |
| GUID | Globally Unique Identifier. |
| HDD | Pevný disk (skratka HDD z angl. hard disk). |
| HTML | Hypertextový značkový obsah (Hypertext Markup Language). |
| HTTP | Hypertextový prenosový protokol (Hypertext Transfer Protocol). |
| HW | Hardvér (Hardware). |
| IaaS | Infraštruktúra ako služba (Infrastructure as a Service). |
| IAM | Modul pre autentifikáciu používateľa. |
| IAM STS | Služba na vydanie SAML tokenov cez integračné rozhranie. |
| IAM WS | Synchrónne služby modulu IAM poskytujúce funkcionalitu nad identitou. |
| IČO | Identifikačné číslo organizácie. |
| ID | Identifikačné číslo. |
| ID-SK | Jednotný dizajn manuál elektronických služieb Slovenska. |
| IDP | Identity provider. |
| IIS | Internet Information Services. |
| IOM | Integrované obslužné miesto. |
| IS | Informačný systém |
| IS CSRU | Informačný systém Centrálnej správy referenčných údajov. |
| IS VS | Informačný systém verejnej správy. |
| ISO | International Organization for Standardization. Organizácia vydávajúca medzinárodné štandardy. |
| IT | Informačné technológie. |
| IT TOGAF | Medzinárodne akreditovaný certifikát, potvrdzujúci znalosti v oblasti Enterprise Architektúry. |
| IT VS | Informačné technológie verejnej správy. |
| JSON | JavaScript Object Notation. Štandardný formát súborov a formát výmeny údajov. |
| KC | Kontaktné centrum. |
| EP | Elektronický podpis. |
| KEP | Kvalifikovaný elektronický podpis, používa sa aj QES. |
| KK | Konto Košičana |
| KP | Katalóg požiadaviek. |
| KPI | Kľúčový ukazovateľ výkonnosti (Key performance Indicator). |
| KS | Koncová služba (KS) je služba, ktorá napĺňa určitú potrebu používateľa pri komunikácii s verejnou správou. Prostredníctvom koncovej služby sa uskutočňuje komunikácia občana s inštitúciou verejnej správy (G2C), podnikateľa s inštitúciou verejnej správy G2B), inštitúcie verejnej správy s inštitúciou verejnej správy (G2G), ak sú v postavení aplikácie práva a účastníka konania a inštitúcie verejnej správy so zahraničnou inštitúciou verejnej správy (G2A). |
| LDAP | Lightweight Directory Access Protocol. Protokol pre ukladanie a prístup k dátam. |
| MEF / eForm | Modul elektronických formulárov. |
| MEP | Platobný modul (modul elektronických platieb). |
| META IS | Centrálny metainformačný systém verejnej správy. |
| MF SR | Ministerstvo financií SR. |
| MH SR | Ministerstvo hospodárstva SR. |
| mID | Mobilné ID reprezentované mobilnou aplikáciou Slovensko v mobile. |
| MIRRI | Ministerstvo investícií, regionálneho rozvoja a informatizácie SR . |
| MK SR | Ministerstvo kultúry SR. |
| MPSVaR SR | Ministerstvo práce, sociálnych vecí a rodiny SR. |
| MŠVVaŠ SR | Ministerstvo školstva, vedy, výskumu a športu SR. |
| MV SR | Ministerstvo vnútra SR. |
| MVC | **Model View Controller** - návrhový vzor, ktorý sa bežne používa na vývoj používateľských rozhraní. |
| NASES | Národná agentúra pre sieťové a elektronické služby vznikla 1. januára 2009 ako príspevková organizácia Úradu vlády Slovenskej republiky (ďalej ako „ÚV SR“) za účelom plnenia odborných úloh v oblasti informatizácie spoločnosti vyplývajúcich zo zákona č. 275/2006 Z. z. o informačných systémoch verejnej správy a o zmene a doplnení niektorých zákonov v znení neskorších predpisov, správy a prevádzkovania elektronických komunikačných sietí a služieb, pre Úrad vlády Slovenskej republiky (ktoré ÚV SR vyplývajú z § 24 ods. 3 zákona č. 575/2001 Z. z. o organizácii činnosti vlády a organizácii ústrednej štátnej správy, ako aj §5 ods. 2 zákona č. 275/2006 Z. z. o informačných systémoch verejnej správy a o zmene a doplnení niektorých zákonov v znení neskorších predpisov), a aj pre ostatné orgány štátnej správy, právnické osoby a fyzické osoby, ktoré požadujú informácie, údaje z informačných systémov, databáz a registrov verejnej správy. Od 1. januára 2019 bola NASES delimitovaná pod Úrad podpredsedu vlády pre investície a informatizáciu (súčasné Ministerstvo investícií, regionálneho rozvoja a informatizácie SR). |
| NBÚ | Národný bezpečnostný úrad. |
| NET | Platforma pre vývoj webových aplikácií. |
| Nginx | Webový server. |
| NKIVS | Národná koncepcia informatizácie verejnej správy. |
| OP | Občiansky preukaz. |
| OPZ | Opis predmetu zákazky. |
| Oracle | Systém riadenia bázy dát. |
| OS | Operačný systém. |
| OST | Object Storage, objektové úložisko ÚPVS. |
| OVM | Orgán verejnej moci. |
| OWASP | Open Web Application Security Project. Určuje základné opatrenia pre zabezpečenie webu. |
| PaaS | Platforma ako služba (Platform as a Service). |
| PK | Prístupový komponent. |
| PO | Právnické osoby. |
| RAM | Pamäť s priamym prístupom alebo RAM (angl. Random Access Memory). |
| RAW data | Surové (RAW) dáta. |
| Rest API | Representational State Transfer. Architektúra rozhraní. |
| RFO | Register fyzických osôb. |
| RPO | Register právnických osôb. |
| SaaS | Softvér ako služba (Software as a Service). |
| SAML | Security Assertion Markup Language - Standard založený na XML (). |
| SLA | Service level Agreement, servisná zmluva. |
| SOAP | Simple Object Access Protocol. |
| SP | Service provider – poskytovateľ služby, spravidla právnická osoba alebo inštitúcia. |
| SQL | Štruktúrovaný vyhľadávací jazyk (Structured Query Language). |
| SR | Slovenská republika. |
| SSL | Secure Sockets Layer. |
| SVM | Slovensko v mobile. |
| SW | Softvér (Software). |
| SZČO | Samostatne zárobkovo činná osoba. |
| TEST | Prostredie pre vykonanie funkčných testov na strane Zhotoviteľa. |
| TO-BE | Budúci stav. |
| UAT | Prostredie určené pre vykonanie používateľských akceptačných testov (user acceptance testing environment). |
| UI | Používateľské rozhranie (User Interface). |
| UIR | Univerzálne integračné rozhranie. |
| UIR | Univerzálne integračné rozhranie - synchrónne rozhranie modulu G2G pre príjem SkTalk správ. |
| UML | Unified Modeling Language. Modelovací jazyk určený na vizualizáciu, špecifikáciu, navrhovanie a dokumentáciu programových systémov. |
| ÚPVII / ÚPPVII / ÚPPVIaI | Úrad podpredsedu vlády pre investície a informatizáciu (súčasné Ministerstvo investícií, regionálneho rozvoja a informatizácie SR). |
| ÚPVS | Ústredný portál verejnej správy. |
| URL | Uniform Resource Locator. Definuje doménovú adresu servera, umiestnenie zdroja na serveri a protokol, ktorým je možné k zdroju pristupovať. |
| USR | Univerzálne synchrónne rozhranie ÚPVS |
| ÚV SR | Úrad vlády Slovenskej republiky. |
| UX | Používateľský zážitok ( User Experience ). |
| VS | Verejná správa. |
| WEB SSO | Jednorazová jednotná autentifikácia na jednom mieste cez webový prehliadač. |
| WSDL | Web Services Description Language. |
| WYSIWYG | Princíp verného prenosu vizuálnej informácie (What You See Is What You Get). |
| XML | Extensible Markup Language. |
| XPATH | Dotazovací jazyk na výber uzlov z dokumentu XML. |
| XSD | XML Schema Definition. |
| XSLT | Jazyk určený na transformovanie XML. |
| XSS | Metóda narušenia WWW stránok využitím bezpečnostných chýb v skriptoch. |
| Z.z. | Zbierka zákonov. |
| ZoD | Zmluva o dielo. |
| ŽS | Životná situácia. |

1. <https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol#Request_methods> [↑](#footnote-ref-1)