

# API LEK13v6 – Návod pro vývojáře SW

## 1. Úvod

API LEK13v6 (dále jen “hlášení”) používá synchronní REST API webové služby, využívající JSON datový formát. API je zabezpečené autentičným certifikátem lékárny vydaným SÚKL.

Datový model operací a popis operací webové služby je zveřejněn pomocí Swagger a nadstavby Swagger UI na adrese <https://api2.sukl.cz/lek13>. Nadstavba Swagger UI je přímo spojená s API a dovoluje pomocí autentičného certifikátu SÚKL testovat hlášení a uživatelsky prohlížet datové rozhraní hlášení.

Webová služba je dostupná na totožné adrese <https://testapi.sukl.cz/lek13/v3/{navez operace}>.. Produkční přístupové body budou vystaveny na stejné URL adrese bez uvedení slova “test”.

## 2. Aktualizace dokumentu

26.9.2018 pokyn k testovacím přístupům. Všechny testovací pracoviště mají ICZ “00000066”. Pro testování lze použít testovací přístupy lékárny k systému eReceptu napojené na testovací identity managementu SÚKL, tzn. poslední vydané přístupy. Validní přístup lze identifikovat pomocí O a OU v certifikátu, v podobě min. 6místného číselného ID.

## 3. Aktualizace datového rozhraní

12.7.2018 byla provedena aktualizace předchozí verze datového modelu. Popis provedených úprav je popsán v XLS souboru, umístěném ve stejné sekci portálu. Veškeré změny byly provedeny v souladu s vyhláškou 84/2008 Sb.

1.11.2018 byla provedena aktualizace předchozí verze datového modelu. Popis provedených úprav je popsán v XLS souboru, umístěném ve stejné sekci portálu. Veškeré změny byly provedeny v souladu s vyhláškou 84/2008 Sb.

10.12.2018 opravena chyba v názvu produkční API.

## 4. Výčet operací hlášení

### 4.1. Recept

API poskytuje čtyři operace pro práci s hlášením receptů. Hlášení se provádí pro oba způsoby výdeje na recept. Datový model obsahuje tři typy léčiv (reglp, nreglp, iplp). V případě absence jakéhokoliv typu léčiva v hlášení je nutné uvést prázdný element, např. „nreglp“:[ ].

#### recept : API pro hlášení výdejů léčivých přípravků na recept

Show/Hide | List Operations | Expand Operations

POST	/recept	Založí hlášení výdeje na Recept.
DELETE	/recept/{podani}	Zruší nahlášený výdej na Recept.
GET	/recept/{podani}	Vrátí nahlášený výdej na Recept.
PUT	/recept/{podani}	Změní nahlášený výdej na Recept.

### 4.1.1. Založení hlášení

Operace POST `"/recept"` založí hlášení výdeje na recept. Hlášení může obsahovat libovolné množství registrovaných, neregistrovaných a IPLP léčivých přípravků, resp. vše co je vydáno na předpis.

### 4.1.2. Smazání hlášení

Operace DELETE `"/recept/{podani}"` smaže hlášení výdeje na recept. Vstupní parametr je UUID podaniID. Smazané UUID hlášení nelze dále použít, např. pokud chci hlášení přepsat formou smazání a vložení nového opraveného hlášení.

### 4.1.3. Načtení hlášení

Operace GET `"/hlaseni/{podani}"` načte hlášení výdeje na recept. Vstupní parametr je UUID podaniID.

### 4.1.4. Změna hlášení

Operace PUT `"/recept/{podani}"` upraví hlášení výdeje na recept. Vstupní parametr je UUID podaniID původního hlášení. Operace kompletně přepíše původní údaje hlášení, zůstává pouze UUID. Pro ID položky je vždy nutné vygenerovat nové UUID.

## 4.2. Žádanka

API poskytuje čtyři operace pro práci s hlášením na žádanku. Datový model obsahuje tři typy léčiv (reglp, neregplp, iplp). V případě absence jakéhokoliv typu léčiva v hlášení je nutné uvést prázdný element, např. „neregplp“: [ ].

žadanka : API pro hlášení výdejů léčivých přípravků na žádanku		<a href="#">Show/Hide</a>	<a href="#">List Operations</a>	<a href="#">Expand Operations</a>
POST	/žadanka			Založí hlášení výdeje na Žádanku.
DELETE	/žadanka/{podani}			Zruší nahlášený výdej na Žádanku.
GET	/žadanka/{podani}			Vrátí nahlášený výdej na Žádanku.
PUT	/žadanka/{podani}			Změní nahlášený výdej na Žádanku.

### 4.2.1. Založení hlášení

Operace POST `"/žadanka"` založí hlášení výdeje na recept. Hlášení může obsahovat libovolné množství registrovaných, neregistrovaných a IPLP léčivých přípravků, resp. vše co je vydáno na žádanku.

### 4.2.2. Smazání hlášení

Operace DELETE `"/žadanka/{podani}"` smaže hlášení výdeje na žádanku. Vstupní parametr je UUID podaniID. Smazané UUID hlášení nelze dále použít, např. pokud chci hlášení přepsat formou smazání a vložení nového opraveného hlášení.

### 4.2.3. Načtení hlášení

Operace GET “/zadanka/{podani}” načte hlášení výdeje na žádanku. Vstupní parametr je UUID podaniID.

### 4.2.4. Změna hlášení

Operace PUT “/zadanka/{podani}” upraví hlášení výdeje na žádanku. Vstupní parametr je UUID podaniID původního hlášení. Operace kompletně přepíše původní údaje hlášení, zůstává pouze UUID. Pro ID položky je vždy nutné vygenerovat nové UUID.

## 4.3. Volný prodej s omezením

API poskytuje čtyři operace pro práci s hlášením volného prodeje s omezením. Datový model obsahuje pouze registrovaná léčiva.

**volny prodej : API pro hlášení výdejů volně prodejných registrovaných léčivých přípravků s omezením**

		<a href="#">Show/Hide</a>   <a href="#">List Operations</a>   <a href="#">Expand Operations</a>
POST	/volny	Založí hlášení výdeje léčivých přípravků volného prodeje s omezením.
DELETE	/volny/{podani}	Zruší nahlášený výdej léčivých přípravků volného prodeje s omezením.
GET	/volny/{podani}	Vrátí nahlášený výdej léčivých přípravků volného prodeje s omezením..
PUT	/volny/{podani}	Změní nahlášený výdej z volného prodeje.

### 4.3.1. Založení hlášení

Operace POST “/volny” založí hlášení výdeje OTC s omezením. Hlášení může obsahovat libovolné množství registrovaných léčivých přípravků, resp. vše co je vydáno během jednoho výdeje.

### 4.3.2. Smazání hlášení

Operace DELETE “/volny/{podani}” smaže hlášení výdeje OTC s omezením. Vstupní parametr je UUID podaniID. Smazané UUID hlášení nelze dále použít, např. pokud chci hlášení přepsat formou smazání a vložení nového opraveného hlášení.

### 4.3.3. Načtení hlášení

Operace GET “/volny/{podani}” načte hlášení výdeje OTC s omezením. Vstupní parametr je UUID podaniID.

### 4.3.4. Změna hlášení

Operace PUT “/volny/{podani}” upraví hlášení výdeje OTC s omezením. Vstupní parametr je UUID podaniID původního hlášení. Operace kompletně přepíše původní údaje hlášení, zůstává pouze UUID. Pro ID položky je vždy nutné vygenerovat nové UUID.

## 5. Načtení JSON schématu

JSON schéma je možné načíst na uvedených URL:

<https://api2.sukl.cz/lek13.swagger.json>

<https://testapi.sukl.cz/lek13.swagger.json>

## 6. Swagger UI

Swagger UI je grafické rozšíření popisu API Swagger. UI obsahuje popis datového modelu každé operace a umožňuje testovat jednotlivé operace.

### 6.1. Popis datového modelu

Jednotlivé položky hlášení každé operace jsou popsány v UI po kliknutí na záložku "model". Popis obsahuje i výčet číselníkových položek.

Popis testovacího rozhraní (model) může obsahovat drobné nepřesnosti. V případě nesrovnalosti modelu s JSON schématem, nebo s response/request daty nám prosím tuto informaci sdělte na kontaktním centru <https://kc.sukl.cz>.

Kompletní výčet datových položek je možné získat pomocí Swagger definice na adrese:

<https://testapi.sukl.cz/lek13.swagger.json>

## 7. Testování API

API lze pomocí Swagger UI testovat. Každá operace obsahuje example data, která se po kliknutí překlápí do části pro odeslání API. Operaci lze následně testovat pomocí tlačítka "Try out". Operace bez elementu "Body" neobsahují example data, volají se přímo, případně pouze s parametry.

Jednotlivá volání ve Swagger UI jsou interně volána pomocí příkazu CURL, který může klient využít pro volání API.

Testování lze provádět pouze pomocí nového testovacího certifikátu SÚKL, který je nutné mít importovaný v úložišti certifikátů operačního systému, případně přímo v prohlížeči (Firefox).

Další možnost testování hlášení nabízí jakýkoliv REST API klient, vhodný je např. Postman.