



OEP
Betegétút lekérdezés házi orvosok
és
vénytörténet lekérdezés patikák
számára

API dokumentáció

verzió: 2.01
2013.03.26

Tartalomjegyzék

1	BEVEZETÉS	3
1.1	A fejlesztés célja.....	3
2	API ELÉRÉS ÉS MŰKÖDÉS	3
2.1	Web szolgáltatás autentikáció	5
2.1.1	WSS.....	5
2.1.2	TLS.....	6
2.1.3	PKI.....	6
3	WEB SZOLGÁLTATÁSON KERESZTÜL ELÉRHETŐ FUNKCIÓK	7
3.1.1	Kliens készítése	7
3.2	Betegéletút lekérdezés indítása (házi orvosok számára)	7
3.3	Betegéletút eredményének lekérdezése.....	10
3.4	Vénytörténet lekérdezés indítása (gyógyszertárak számára).....	12
3.5	Vénytörténet eredményének lekérdezése	14
3.6	Üzemelés tesztelése (ping)	16
4	TESZTRENDSZER.....	18
4.1	Teszt felület	18
4.2	Éles felület	26

1 BEVEZETÉS

1.1 A FEJLESZTÉS CÉLJA

Az egészségügyi és a hozzájuk kapcsolódó személyes adatok kezeléséről és védelméről szóló 1997. évi XLVII. törvény (továbbiakban Eüak.) 2011. április 1-től hatályos rendelkezései lehetővé teszik, hogy a háziorvosok az Országos Egészségbiztosítási Pénztár (továbbiakban OEP) informatikai rendszeréből lekérdezzék betegük mikor és milyen jellegű, a társadalombiztosítás által támogatott egészségügyi szolgáltatást vett igénybe.

A háziorvosok részére a saját betegeikre vonatkozóan a lekérdezés az OJOTE rendszer webes felületéről 2011 áprilisa óta lehetséges.

Igény jelentkezett API szolgáltatás fejlesztésére, amely kapcsán lehetővé kell tenni a háziorvosok számára, hogy orvosi alkalmazásaikból elérhető legyen a betegéletút lekérdezés webszolgáltatásként.

Ugyanezt a lehetőséget, némileg korlátozott adattartalommal 2013-ban a törvény a gyógyszertárak számára is engedélyezte, és 2013.04.01-től kezdve a patikák is használhatják a betegéletút API bizonyos webszolgáltatásait.

A megfelelő biztonság elérése érdekében a szolgáltatások csak HTTPS protokollon keresztül érhetők el. A felhasználó azonosítására a SOAP üzenet fejlécében, a WS-Security szabványnak megfelelően elhelyezett felhasználói név és jelszó szolgál.

A webszolgáltatás használatának további feltétele az is, hogy a fejlesztő (vagy fejlesztő cég) elfogadja a regisztrációra vonatkozó általános szerződési feltételeket, és a programját az OEP erre szolgáló felületén beregisztrálja.

2 API ELÉRÉS ÉS MŰKÖDÉS

Az OJOTE betegéletút és vénytörténet lekérdezés funkcióit az arra jogosult háttérrendszerek szabványos, webszolgáltatás felületen keresztül a

<https://jogviszony.oep.hu/eletut/eletut>

címen érhetik el.

A **W3C** definíciója szerint a Web szolgáltatás egy olyan szoftver rendszer, amely támogatja a gép-gép közötti, hálózaton keresztüli kommunikációt. A szolgáltatás kliens oldalról nézve gyakran csak alkalmazás programozói felület (**API**), amely hálózaton keresztül (például internet) érhető el, és a végrehajtás egy távoli gépen történik, amely a szolgáltatást nyújtja.

FONTOS!

A Betegéletút és vénytörténet lekérdezés igen erőforrás igényes, ezért az API funkció használatát nagyon körültekintően kell végezni. A túl gyakori, nagytömegű lekérdezés a rendszer túlterheléséhez, legrosszabb esetben működésképtelenné válásához vezethet.

A funkcióhívásokhoz időbeli és gyakorisághoz kötődő megszorításokat építettünk be.

A felhasználói programok fejlesztőinek figyelembe kell venniük a környezet korlátait, és a funkcióhívásokat nem automatikusan, hanem csak kifejezett igény esetén, egyedi lekérdezésre szabad engedniük.

Az egyes webszolgáltatások meghívása a következő módon végezhető el:

- Meg kell hívni a kívánt paraméterekkel
 - háziorvosi felhasználó esetén a **betegéletut** funkciót
 - gyógyszerértékesítési felhasználó esetén a **venytortenet** funkciót
- Ha a bejelentkezési adatok és program azonosító helyes, akkor a hívó vissza kap egy **jegy** és egy **ido** értéket, amelyre hivatkozva a későbbiekben megszerezheti a lekérdezés eredményét. A lekérdezések feldolgozására nem születik azonnal válasz, a rendszer a háttérben kezdi el őket feldolgozni. A lekérdezést a későbbiekben a **jegy** segítségével lehet azonosítani. A visszaadott **ido** értéke másodpercben értendő és azt határozza meg, hogy mennyi idő múlva kérdezhető le leg hamarabb az eredmény. A visszaadott idő egy durva becslésen alapul.
- A visszakapott **ido** másodperc elteltével meg lehet hívni az **eredmeny** funkciót, amely helyes adatok megadása esetén kétféle eredményt adhat annak függvényében, hogy a lekérdezés eredménye elkészült vagy sem.
- Ha a lekérdezés eredménye még nem készült el, akkor egy üres struktúrát kapunk vissza, amelyben csak az **ido** paraméter értéke van kitöltve, amelynek értelmezése teljesen megegyezik

a betegéletut funkció meghívásakor visszakapott ido érték értelmezésével (ennyi másodperc múlva kell újra az eredmény után érdeklődnünk).

- Ha az eredmény elkészült, akkor az **ido** paraméterben **0** értéket kapunk vissza és a funkció részletes leírásánál található struktúrában megkapjuk az adatokat.

A W3C által kiadott definíció több, különböző típusú rendszert magába foglal, de az általánosan elterjedt használati mód a SOAP formátumú XML üzenetekkel történő kommunikációt, a felület leírására pedig a WSDL formátumot alkalmazza.

A WSDL egy XML alapú, Web szolgáltatás leíró nyelv. Segítségével a hálózaton keresztül kiejánlott szerver oldali szolgáltatásokkal történő kommunikáció módját lehet leírni. Interneten keresztüli Web szolgáltatáshoz a WSDL, gyakran a SOAP és az XML Schema szabványokkal együtt kerül alkalmazásra. A kliens program, amely egy Web szolgáltatáshoz csatlakozik, a WSDL állományokon keresztül tudhatja meg, hogy milyen funkciók érhetők el a szerveren. A szolgáltatásokhoz tartozó speciális adattípusok, a WSDL állományba ágyazhatók XML Schema formátumban. Az OJOTE szolgáltatás visszaterési értékének típusa speciális adattípus, így a WSDL állományban meg kell adni a felépítését. Az XML Schema validációs szándékkal lett létrehozva. Tartalmára nézve olyan szabályok halmaza, amelynek ha egy XML dokumentum megfelel, akkor az a dokumentum valid arra a sémára nézve.

A WSDL állománynak tartalmaznia kell a szolgáltatáshoz tartozó üzenetek formátumát. A megadott formátum meghatározza, hogy az adott üzenet milyen részeket tartalmaz, és azoknak mi a típusa.

2.1 WEB SZOLGÁLTATÁS AUTHENTIKÁCIÓ

A Web szolgáltatás nyújtásakor történő autentikációt az alkalmazás szerver végzi, az **OASIS WS-Security 1.0** biztonsági szabványának (WSS) megfelelően. Az autentikáció kliens oldali teendőit a jól megválasztott SOAP környezet elintézi. Első körben az autentikáció a SOAP üzenet fejlécében elhelyezett felhasználói név és jelszó alapján történik, a későbbiekben X509 tanúsítvány segítségével.

2.1.1 WSS

A WS-Security egy kommunikációs protokoll, amely a Web szolgáltatások számára nyújt biztonsági megoldást. A protokoll specifikációkat tartalmaz arról, hogyan lehet az üzenetek integritását és titkosságát megtartani az üzenetváltás alatt (például tanúsítványok alkalmazásával). A Web szolgáltatás üzenetek integritásának és titkosságának kényszerítése történhet biztonságos szállítási réteg (TLS) alkalmazásával is (például HTTPS protokollon keresztüli üzenetküldéssel). Ez utóbbi jelentősen csökken-

ti a többletterhelést. A **WSS** a biztonsági szolgáltatásokat a **SOAP** üzenet fejlécén keresztül nyújtja, az alkalmazás rétegen, így biztosítja a végpontok közötti biztonságot.

2.1.2 TLS

Biztonságos szállítási réteg (**Transport Layer Security - TLS**) és az őse (**SSL**) titkosító protokollok, amelyek biztonságos adat továbbítást nyújtanak interneten keresztül. A TLS protokoll lehetőséget nyújt az alkalmazások számára, hogy a hálózaton keresztül kommunikáljanak, megghusítva a hallgatóságokat (**eavesdropping**), babrálásokat (**tampering**) és üzenet hamisítást (**message forgery**). A TLS végpont autentikációt és titkosságot nyújt az interneten keresztüli kommunikáció számára titkosítás alkalmazásával. Tipikusan csak a szerver autentikált (azonossága garantált), a kliens nem kerül azonosításra. Ez azt jelenti, hogy a felhasználó (amely lehet személy, vagy egy alkalmazás, mint például egy Web böngésző) biztos lehet abban, hogy kivel kommunikál. A biztonság következő fokozata, amikor a kommunikáció mindkét fele biztos lehet a másik fél identitásában, ezt kölcsönös azonosításnak hívják. A kölcsönös azonosítás publikus kulcs infrastruktúrát (**PKI**) igényel, amelyet a kliensre telepíteni kell.

A TLS három alap fázist tartalmaz:

1. Egyenrangú egyeztetés a támogatott algoritmusokról
2. Nyilvános kulcsú titkosítás, amely kulcs cserén és tanúsítvány alapú hitelesítésen alapszik
3. Szimmetrikus kód, amely forgalomtitkosításon alapszik

A TLS az alkalmazás protokollok alatt (például **HTTP**, **FTP**), de a szállítási protokollok (például **TCP**, **UDP**) felett helyezkedik el, képes biztonságossá tenni tetszőleges protokollt, leggyakrabban a **HTTP** protokollal használják (ez a **HTTPS**).

2.1.3 PKI

A kriptográfiában a nyilvános kulcsú infrastruktúra (**PKI**) egy megbízható harmadik fél által lektorált és a felhasználó azonosságára kezességet vállaló intézményen alapszik. Lehetőséget biztosít a nyilvános kulcsok felhasználóhoz történő rendelésére. A nyilvános kulcsok általában tanúsítványokban vannak. A **PKI** lehetőséget biztosít arra, hogy felhasználók azonosítsák egymást, és a tanúsítványban tárolt információk használatával titkosítsák, illetve dekódolják az üzeneteiket. Általánosan a **PKI** kliens és szerver szoftverből, valamilyen hardver elemből (például smart card) állhat.

3 WEB SZOLGÁLTATÁSON KERESZTÜL ELÉRHETŐ FUNKCIÓK

A betegéletút lekérdezés kettő funkciót tartalmaz, egyet a lekérdezés indítására, egyet pedig az eredmény letöltésére. Ezek a szolgáltatások a WSDL állományban kerültek leírásra, amely alapján kliens készíthető.

3.1.1 Kliens készítése

A legtöbb programozási környezet támogatja a WSDL állomány alapján történő SOAP kliens generálást. A kliens generálása programozási környezettől függően történhet fordítási, illetve futtatási időben.

A funkciókhoz tartozó WSDL állomány a mellékletben található.

Mind a futtatási időben, mind pedig a fordítási időben generált kliens végpont címe eltérhet a szükségstől, ezt ellenőrizni kell, de még jobb, ha a végpont címét explicit felül definiáljuk a megfelelőre. A teszt rendszer esetén a megfelelő végpont cím <http://tesztjogviszony.oep.hu/eletut/eletut>, az éles rendszer esetén <https://jogviszony.oep.hu/eletut/eletut>.

3.2 BETEGÉLETÚT LEKÉRDEZÉS INDÍTÁSA (HÁZIORVOSOK SZÁMÁRA)

Használt webszolgáltatás: betegéletut

A betegéletút lekérdezés indításához használt funkció. Ellenőrzi, hogy a WS-Security fejlécben meghatározott OJOTE felhasználó jogosult-e a megadott TAJ-hoz tartozó betegéletút információkat lekérdezni, és megfelelő jogosultság esetén elindítja a lekérdezési folyamatot.

Egy felhasználónak egy időben 1 betegéletút lekérdezése lehet.

Az eljárás paramétere a TAJ szám és a lekérdezendő időszakot meghatározó kezdő és befejező dátum. A két dátum között legfeljebb 365 naptári nap lehet.



betegletut webszolgáltatás paraméterei		
Név	Típus	Leírás
program	szöveg	Program azonosító
taj	szöveg	A lekérdezendő betegletúthoz tartozó TAJ szám
tol	szöveg	Időszak kezdete paraméter (éééé.hh.nn formában)
ig	szöveg	Időszak vége paraméter (éééé.hh.nn formában)

A visszatérési érték egy Jegy nevű adatstruktúra, mely egy **jegy** és egy időt tartalmaz. Az idő megmondja, hogy a jegyet felhasználva hány másodperc múlva lehet meghívni az eredmény webszolgáltatást az betegletút lekérdezés eredményének megszerzéséért.

A jegy 1 órán belül használható fel.

a Jegy nevű adatstruktúra attribútumai		
Név	Típus	Leírás
jegy	szöveg	Lekérdezéshez tartozó azonosító
ido	szám	Az eredmény lekérdezéséhez ennyi másodpercet kell várni


Lehetséges hibák

Kód	Leírás
8	Hibás név vagy jelszó
9	A bejelentkezés nem érhető el
10	A felhasználó letiltásra került.
101	Kérjük lépjen be az OJOTE rendszerbe és változtassa meg a jelszavát.
102	A felhasználó nem található a rendszerben vagy nincs megfelelő jogosultsága. (pl. státusza nem érvényes, nincs eJelentés jogosultsága, stb.)
103	Az Ön által használt felhasználó státusza még nem érvényes.
104	Az Ön által használt felhasználó státusza már nem érvényes.
105	Az Ön által használt felhasználó jogosultsága nem megfelelő a funkció végrehajtásához.
3000	Belső hiba, kérem próbálja meg a lekérdezést később
3001	Hibás vagy üres TAJ parameter
3002	Hibás vagy üres tol parameter
3003	Hibás vagy üres ig parameter
3004	A kezdő és befejező datum között nem lehet több 365 napnál
3005	A megadott TAJ nem kérdezhető le (pl. nyilatkozat)
3006	A felhasználó nem futtathat több párhuzamos lekérdezést kérem próbálja meg a lekérdezést később
3007	A tol parameter értéke nem lehet nagyobb, mint az ig parameter értéke
3008	Érvénytelen program azonosító
3009	A felhasználó túllépte a napi lekérdezési keretet.

3.3 BETEGÉLETÚT EREDMÉNYÉNEK LEKÉRDEZÉSE

Használt szolgáltatás neve: **eredmeny**

A felhasználók a betegéletút lekérdezés indításakor vissza kapott **jegy** segítségével le lekérdezhetik az általuk indított lekérdezés aktuális állapotát. Ha a lekérdezés sikeresen lefutott és az eredménye rendelkezésre áll, akkor a visszatérési érték az eredményt is tartalmazza.



eredmeny webservice paraméterei		
Név	Típus	Leírás
jegy	szöveg	A lekérdezés indításakor vissza kapott jegy

A visszatérési érték egy **eredmeny** struktúra.

eredmeny attribútumok		
Név	Típus	Leírás
ellatasok	Ellatas tömb	Ellátási adatok tömbje
venyek	Veny tömb	Vény kiváltási adatok tömbje
penzbeli	Penz tömb	Pénzbeli ellátási adatok tömbje
ido	szám	A következő lekérdezéshez ennyi másodpercet kell várni. Értéke általában 1 és 200 másodperc közötti

Az eredmény struktúra ellátási, vény kiváltási és pénzbeli ellátási adatok tömbjeit tartalmazza.

Ellatas attribútumok

Név	Típus	Leírás
szolg_nev	szöveg	Szolgáltató neve
szolg_cim	szöveg	Szolgáltató címe
kezdete	szöveg	Szolgáltatás kezdete
vege	szöveg	Szolgáltatás vége
ellatas_nev	szöveg	Ellátás neve
tamogatas	szöveg	Támogatás összege forintban
orv_kod	szöveg	Orvos kód
orv_nev	szöveg	Orvos név

Veny attribútumok

Név	Típus	Leírás
pat_nev	szöveg	Vénykiváltás helye
pat_cim	szöveg	Vénykiváltás helyének címe
kivaltas	szöveg	Kiváltás ideje
termek	szöveg	Kiváltott termék vagy szolgáltatás
tamogatas	szöveg	Támogatás összege forintban
ter_dij	szöveg	Beteg térítési díj forintban
orv_kod	szöveg	Orvos kód
orv_nev	szöveg	Orvos név

Penzbeli attribútumok

Név	Típus	Leírás
jellege	szöveg	A pénzbeli ellátás jellege
kezdete	szöveg	Ellátás kezdete
vege	szöveg	Ellátás vége

A pénzbeli ellátásoknál az összeget nem adjuk át.

Lehetséges hibák	
Kód	Leírás
8	Hibás név vagy jelszó
9	A bejelentkezés nem érhető el
10	A felhasználó letiltásra került.
101	Kérjük lépjen be az OJOTE rendszerbe és változtassa meg a jelszavát.
102	A felhasználó nem található a rendszerben vagy nincs megfelelő jogosultsága. (pl. státusza nem érvényes, nincs eJelentés jogosultsága, stb.)
103	Az Ön által használt felhasználó státusza még nem érvényes.
104	Az Ön által használt felhasználó státusza már nem érvényes.
105	Az Ön által használt felhasználó jogosultsága nem megfelelő a funkció végrehajtásához.
106	Az Ön által használt kulcs nem létezik, elavult, vagy erre a funkcióra nem használható. Jelentkezzen be újra, hogy folytatni tudja a munkáját!
3000	Belső hiba, kérem próbálja meg a lekérdezést később
3101	A jegy megadása kötelező
3102	Érvénytelen jegy
3103	Túl gyakori lekérdezés
3104	A lekérdezés hibával ért véget

3.4 VÉNYTÖRTÉNET LEKÉRDEZÉS INDÍTÁSA (GYÓGYSZERTÁRAK SZÁMÁRA)

Használt webszolgáltatás: venytoritenet

Vénytoritenet lekérdezés indításához használt funkció. Ellenőrzi, hogy a WS-Security fejlécben meghatározott e-Jelentés jogú felhasználó jogosult-e a vénytoriteneti információkat lekérdezni, és a megfelelő jogosultság esetén elindítja a lekérdezési folyamatot.

Éles környezet esetén a vénytoritenet lekérdezésre 2013.04.01-től van lehetőség. Ezt megelőzően a vénytoritenet szolgáltatás megfelelő jogosultság és programazonosító használata esetén is a 3008-as hibakódot adja vissza.

A hatályos törvények értelmében a vénytoritenetből bizonyos BNO kódú gyógyszerek kiszűrésre kerülnek. (Lásd: kormányrendelet.)

Egy felhasználónak egy időben egy vénytoritenet lekérdezése lehet!

Az eljárás paramétere a TAJ szám és annak a vénynek a vényazonosítója, amely alapján a gyógyszer-tár a szolgáltatás használatára jogosult lehet.

Fontos! A szoftver fejlesztőjének a vénytörténet által visszaadott tranzakciós kódhoz személyre pontosan el kell tárolnia, hogy mely patika, mely személye kezdeményezte a vénytörténet lekérdezést, hogy egy esetleges visszaélés esetén jogi felelősségre lehessen vonni a kezdeményezőt.



venytortenet webszolgáltatás paraméterei		
Név	Típus	Leírás
program	szöveg	Program azonosító
taj	szöveg	A lekérdezendő betegéletúhoz tartozó TAJ szám
venyazonosito	szöveg	A szolgáltatás igénybevételének alapjául szolgáló vény vényazonosítója
szemely_id	szöveg	A patikai szoftver, ha akarja, átadhatja az OEP számára a webszolgáltatást meghívó személy adatait (pl. pénztár azonosító, felhasználó azonosító, vagy bármi olyan adat, amely egy esetleges visszaélés a lekérdezést kezdeményező személy beazonosítására szolgálhat.) Ezeket az adatokat ugyanakkor a patikai alkalmazásnak is mindenképpen tárolnia kell. A mezőt nem kötelező kitölteni. Maximális hossza: 100 karakter.

A visszatérési érték egy VenytortenetJegy nevű adatstruktúra, mely egy **jegyet**, a folyamat egyértelmű azonosítására szolgáló **tranzakciós kódot** és egy **időt** tartalmaz. Az idő megmondja, hogy a jegyet felhasználva hány másodperc múlva lehet meghívni az eredményVenyortortenet webszolgáltatást a vénytörténet megszerzéséért.

A jegy 1 órán belül használható fel.

a VenytortenetJegy nevű adatstruktúra attribútumai		
Név	Típus	Leírás
jegy	szöveg	Lekérdezéshez tartozó azonosító
tranz_kod	szöveg	Tranzakciós kód, mely a vénytörténet hívás egyértelmű azonosítására szolgál. Az OEP hivatalos megkeresése esetén az API fejlesztőnek a kapott tranzakciós kódról meg kell tudnia mondania, hogy mely patika, mely alkalmazottja használta.
ido	szám	Az eredmény lekérdezéséhez ennyi másodpercet kell várni. Értéke általában 1 és 200 másodperc közötti


Lehetséges hibák	
Kód	Leírás
8	Hibás név vagy jelszó
9	A bejelentkezés nem érhető el
10	A felhasználó letiltásra került.
101	A felhasználó nem található a rendszerben.
102	Kérjük lépjen be az e-Jelentés rendszerbe és változtassa meg a jelszavát.
103	Az Ön által használt felhasználó státusza még nem érvényes.
104	Az Ön által használt felhasználó státusza már nem érvényes.
105	Az Ön által használt felhasználó jogosultsága nem megfelelő a funkció végrehajtásához.
3000	Belső hiba, kérem próbálja meg a lekérdezést később
3006	A felhasználó nem futtathat több párhuzamos lekérdezést kérem próbálja meg a lekérdezést később
3008	Érvénytelen program azonosító
3201	Hibás vagy üres TAJ paraméter
3202	Hiányzó vényazonosító paraméter
3203	A személy_id mező hossza nagyobb, mint 100 karakter.
3204	A megadott TAJ nem kérdezhető le (a TAJ érvénytelen, a TAJ nem létezik vagy a felhasználó letiltotta a lekérdezhetőségét).

3.5 VÉNYTÖRTÉNET EREDMÉNYÉNEK LEKÉRDEZÉSE

Használt webszolgáltatás: `venytortenetEredmeny`

A felhasználók a betegletút/venytörténet lekérdezés indításakor vissza kapott **jegy** segítségével lekérdezhetik az általuk indított lekérdezés aktuális állapotát. Ha a lekérdezés sikeresen lefutott és az eredménye rendelkezésre áll, akkor a visszatérési érték az eredményt is tartalmazza.

Fontos! A hatályos törvények értelmében a vénytörténetből bizonyos BNO kódú gyógyszerek kiszűrésre kerülnek. (Lásd: kormányrendelet.)



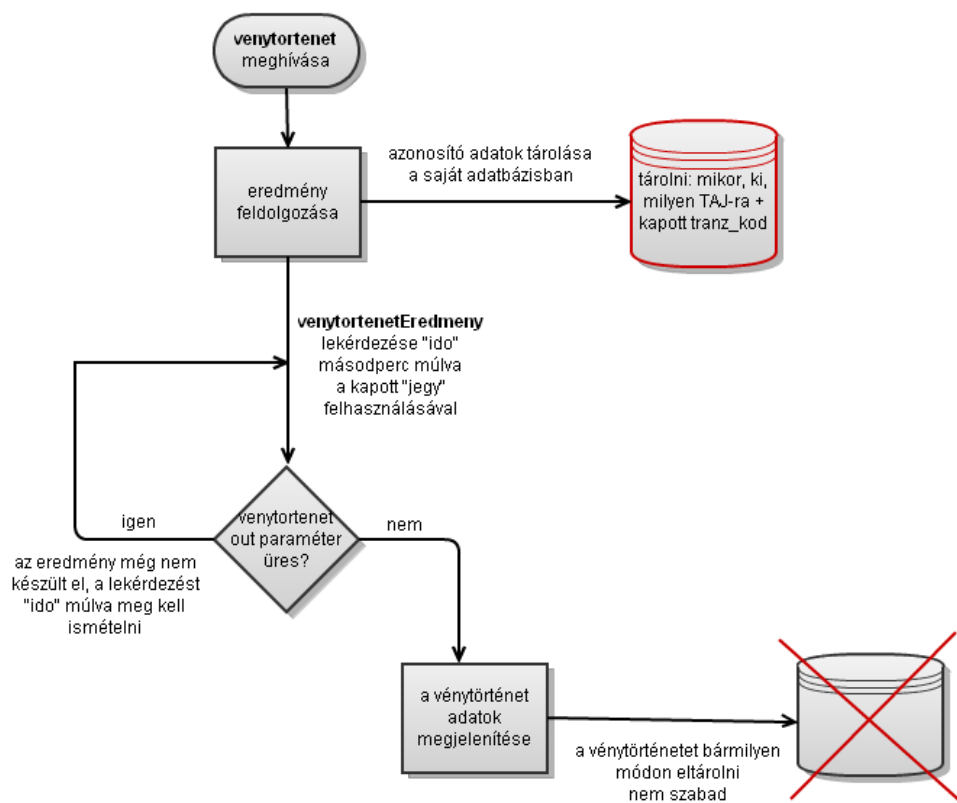
venytortenetEredmeny webszolgáltatás paraméterei		
Név	Típus	Leírás
<code>jegy</code>	szöveg	A lekérdezés indításakor vissza kapott jegy

A visszatérési érték egy **venytortenetAdatok** struktúra.

venytortenetAdatok attribútumok		
Név	Típus	Leírás
venytortenet	VenyrtortenetAdat tömb	Vényrtörténet eredmény adatok tömbje
ido	szám	Az eredmény lekérdezéséhez ennyi másodpercet kell várni

VenyrtortenetAdat attribútumok		
Név	Típus	Leírás
ttt_kod	szöveg (20)	A korábban finanszírozott gyógyszer TTT kódja
gyogyszer_nev	szöveg (150)	A gyógyszer neve
elsz_mennyiseg	szám	Elszámolt mennyiség (-1 esetén a mennyiség nem meghatározható*)
kiv_datuma	szöveg (10)	A kiváltás dátuma <i>éééé.hh.nn</i> formátumban.

Lehetséges hibák	
Kód	Leírás
8	Hibás név vagy jelszó
9	A bejelentkezés nem érhető el
10	A felhasználó letiltásra került.
101	Kérjük lépjen be az OJOTE rendszerbe és változtassa meg a jelszavát.
102	A felhasználó nem található a rendszerben vagy nincs megfelelő jogosultsága. (pl. státusza nem érvényes, nincs eJelentés jogosultsága, stb.)
103	Az Ön által használt felhasználó státusza még nem érvényes.
104	Az Ön által használt felhasználó státusza már nem érvényes.
105	Az Ön által használt felhasználó jogosultsága nem megfelelő a funkció végrehajtásához.
106	Az Ön által használt kulcs nem létezik, elavult, vagy erre a funkcióra nem használható. Jelentkezzen be újra, hogy folytatni tudja a munkáját!
3000	Belső hiba, kérem próbálja meg a lekérdezést később
3101	A jegy megadása kötelező
3102	Érvénytelen jegy
3103	Túl gyakori lekérdezés
3104	A lekérdezés hibával ért véget

A vénytörténet lekérdezésének menete:**3.6 ÜZEMELÉS TESZTELÉSE (PING)**

Bizonyos időnként előfordulhat, hogy a Betegéletút webszolgáltatásokat az OEP karbantartás, rendszerfrissítés, hibajavítás, stb. okból lekapcsolja.

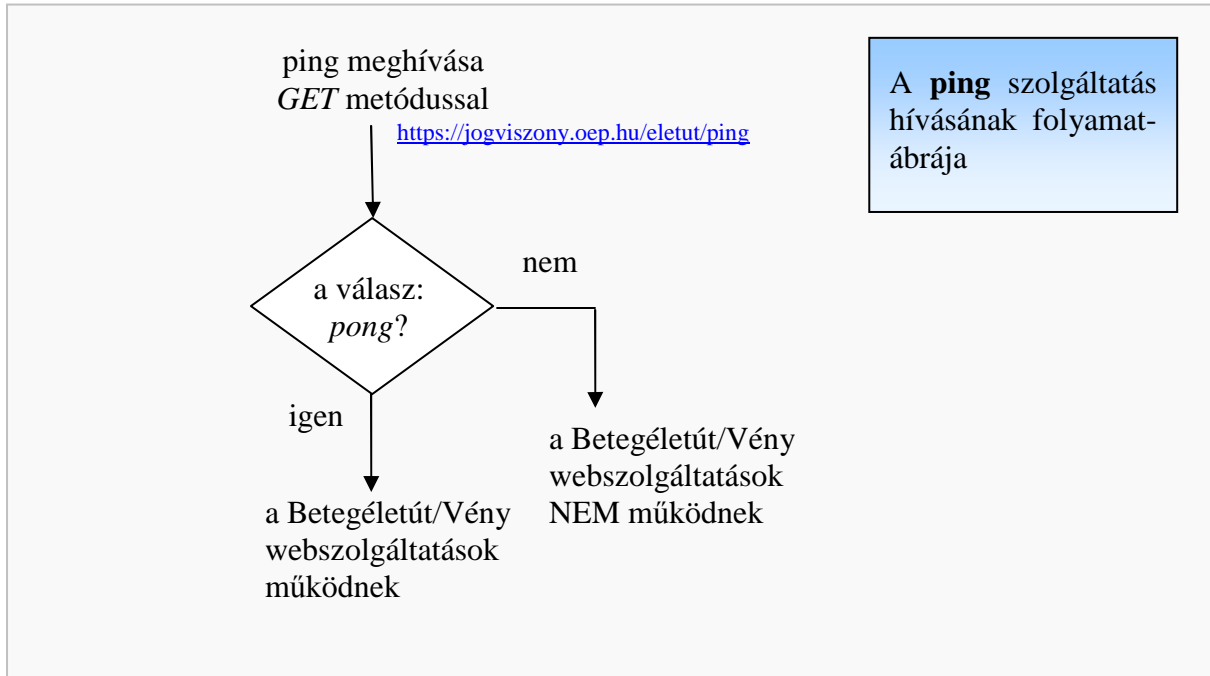
Ha szeretné, hogy programja tisztában legyen vele, hogy a webszolgáltatások adott pillanatban elérhetőek-e vagy sem, kérjük, használja az erre a célra kialakított ping szolgáltatást.

Fontos! A ping nem egy webszolgáltatás, hanem egy egyszerű HTTP/GET metódussal hívható formátlan HTML szöveg. Amennyiben a hívása során a ping által visszatartalom a "pong" szót tartalmazza, a Betegéletút webszolgáltatások elérhetőek, bármilyen egyéb esetben nem.

A ping szolgáltatás címe...

- teszt környezet esetén: <http://tesztjogviszony.oep.hu/eletut/ping>
- éles környezet esetén: <https://jogviszony.oep.hu/eletut/ping>

Kérjük, amennyiben úgy érzi, ennek ping szolgáltatás meghívására Önnek szüksége van, a szolgáltatást legkevesebb 5 másodperces (de inkább 30-120 másodpercenkénti) hívásgyakorisággal hívja meg.



4 TESZTRENSZER

A betegéletút és vénytörténet lekérdezés teszrendszer esetén a <http://tesztjogviszony.oep.hu/eletut/eletut> címen érhető el. A webszolgáltatás kliensek által használandó WSDL állománya a mellékletben található. A teszrendszeren a bejelentkezési nevet és jelszót (amelyet a regisztráció során a kapott) a SOAP üzenet fejlécébe kell helyezni, a WSSE (Web Service Security) leírásában található szöveges név és jelszó megadással.

A teszteléshez szükséges teszt felhasználót az OEP biztosítja a regisztrációt elvégző külső fejlesztők számára.

A betegéletút tesztelésre felhasználható még a Web alkalmazás felülete is, amely a <http://tesztjogviszony.oep.hu/ojote/> címen érhető el. Ez a felület azonban a vénytörténet tesztelésére nem használható.

4.1 TESZT FELÜLET

A <http://tesztjogviszony.oep.hu/eletut/eletut> végponthoz tartozó WSDL állomány

```
<definitions
  name="eletut"
  targetNamespace="http://oep.hu/"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:tns="http://oep.hu/"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
>
<types>
  <schema xmlns="http://www.w3.org/2001/XMLSchema" targetNamespace="http://oep.hu/"
    elementFormDefault="qualified"
      xmlns:tns="http://oep.hu/"      xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xmlns:soap11-enc="http://schemas.xmlsoap.org/soap/encoding/">
<complexType name="BetegeletutException">
  <sequence>
    <element name="kod" type="int" nillable="false"/>
  </sequence>
</complexType>
<element name="betegeletutElement">
  <complexType>
    <sequence>
      <element name="program" type="string" nillable="true"/>
      <element name="taj" type="string" nillable="true"/>
      <element name="tol" type="string" nillable="true"/>
      <element name="ig" type="string" nillable="true"/>
    </sequence>
  </complexType>
</element>
<element name="betegeletutResponseElement">
  <complexType>
    <sequence>
      <element name="result" type="tns:Jegy" nillable="true"/>
    </sequence>
  </complexType>
</element>
<complexType name="Jegy">
  <sequence>
    <element name="ido" type="long"/>
    <element name="jegy" type="string" nillable="true"/>
  </sequence>
</complexType>
<element name="BetegeletutExceptionElement" type="tns:BetegeletutException"/>
<element name="eredmenyElement">
  <complexType>
    <sequence>
      <element name="jegy" type="string" nillable="true"/>
    </sequence>
  </complexType>
</element>
<element name="eredmenyResponseElement">
  <complexType>
```

```
<sequence>
  <element name="result" type="tns:Eredmeny" nillable="true"/>
</sequence>
</complexType>
</element>
<complexType name="Eredmeny">
  <sequence>
    <element name="venyek" type="tns:Veny" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <element name="ido" type="long"/>
    <element name="ellatasok" type="tns:Ellatas" nillable="true"
minOccurs="0" maxOccurs="unbounded"/>
    <element name="penzbeli" type="tns:Penz" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </sequence>
</complexType>
<complexType name="Veny">
  <sequence>
    <element name="termek" type="string" nillable="true"/>
    <element name="kivaltas" type="string" nillable="true"/>
    <element name="pat_nev" type="string" nillable="true"/>
    <element name="ter_dij" type="string" nillable="true"/>
    <element name="tamogatas" type="string" nillable="true"/>
    <element name="orv_kod" type="string" nillable="true"/>
    <element name="pat_cim" type="string" nillable="true"/>
    <element name="orv_nev" type="string" nillable="true"/>
  </sequence>
</complexType>
<complexType name="Ellatas">
  <sequence>
    <element name="szolg_nev" type="string" nillable="true"/>
    <element name="ellatas_neve" type="string" nillable="true"/>
    <element name="tamogatas" type="string" nillable="true"/>
    <element name="orv_kod" type="string" nillable="true"/>
    <element name="kezdete" type="string" nillable="true"/>
    <element name="szolg_cim" type="string" nillable="true"/>
    <element name="orv_nev" type="string" nillable="true"/>
    <element name="vege" type="string" nillable="true"/>
  </sequence>
</complexType>
```

```
<complexType name="Penz">
  <sequence>
    <element name="kezdetek" type="string" nillable="true"/>
    <element name="vege" type="string" nillable="true"/>
    <element name="jellege" type="string" nillable="true"/>
  </sequence>
</complexType>
<element name="venytortenetElement">
  <complexType>
    <sequence>
      <element name="program" type="string" nillable="true"/>
      <element name="taj" type="string" nillable="true"/>
      <element name="venyazonosito" type="string" nillable="true"/>
      <element name="szemely_id" type="string" nillable="true"/>
    </sequence>
  </complexType>
</element>
<element name="venytortenetResponseElement">
  <complexType>
    <sequence>
      <element name="result" type="tns:VenyortenetJegy"
nillable="true"/>
    </sequence>
  </complexType>
</element>
<complexType name="VenyortenetJegy">
  <sequence>
    <element name="ido" type="long"/>
    <element name="tranz_kod" type="string" nillable="true"/>
    <element name="jegy" type="string" nillable="true"/>
  </sequence>
</complexType>
<element name="venytortenetEredmenyElement">
  <complexType>
    <sequence>
      <element name="jegy" type="string" nillable="true"/>
    </sequence>
  </complexType>
</element>
<element name="venytortenetEredmenyResponseElement">
```

```
<complexType>
  <sequence>
    <element name="result" type="tns:VenytortenetAdatok"
nillable="true"/>
  </sequence>
</complexType>
</element>
<complexType name="VenytortenetAdatok">
  <sequence>
    <element name="ido" type="long"/>
    <element name="venytortenet" type="tns:VenytortenetAdat"
nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </sequence>
</complexType>
<complexType name="VenytortenetAdat">
  <sequence>
    <element name="ttt_kod" type="string" nillable="true"/>
    <element name="kiv_datuma" type="string" nillable="true"/>
    <element name="gyogyszer_nev" type="string" nillable="true"/>
    <element name="elsz_mennyisege" type="long"/>
  </sequence>
</complexType>
</schema>
</types>
<message name="Eletut_betegeletut">
  <part name="parameters" element="tns:betegeletutElement"/>
</message>
<message name="Eletut_betegeletutResponse">
  <part name="parameters" element="tns:betegeletutResponseElement"/>
</message>
<message name="BetegeletutException">
  <part name="parameters" element="tns:BetegeletutExceptionElement"
name="BetegeletutExceptionElement"
element="tns:BetegeletutExceptionElement"/>
</message>
<message name="Eletut_eredmeny">
  <part name="parameters" element="tns:eredmenyElement"/>
</message>
<message name="Eletut_eredmenyResponse">
  <part name="parameters" element="tns:eredmenyResponseElement"/>
</message>
```

```
</message>
<message name="Eletut_venytortenet">
  <part name="parameters" element="tns:venytortenetElement"/>
</message>
<message name="Eletut_venytortenetResponse">
  <part name="parameters" element="tns:venytortenetResponseElement"/>
</message>
<message name="Eletut_venytortenetEredmeny">
  <part name="parameters" element="tns:venytortenetEredmenyElement"/>
</message>
<message name="Eletut_venytortenetEredmenyResponse">
  <part name="parameters" element="tns:venytortenetEredmenyResponseElement"/>
</message>
<portType name="eletut">
  <operation name="betegeletut">
    <input message="tns:Eletut_betegeletut"/>
    <output message="tns:Eletut_betegeletutResponse"/>
    <fault name="BetegeletutException" message="tns:BetegeletutException"/>
  </operation>
  <operation name="eredmeny">
    <input message="tns:Eletut_eredmeny"/>
    <output message="tns:Eletut_eredmenyResponse"/>
    <fault name="BetegeletutException" message="tns:BetegeletutException"/>
  </operation>
  <operation name="venytortenet">
    <input message="tns:Eletut_venytortenet"/>
    <output message="tns:Eletut_venytortenetResponse"/>
    <fault name="BetegeletutException" message="tns:BetegeletutException"/>
  </operation>
  <operation name="venytortenetEredmeny">
    <input message="tns:Eletut_venytortenetEredmeny"/>
    <output message="tns:Eletut_venytortenetEredmenyResponse"/>
    <fault name="BetegeletutException" message="tns:BetegeletutException"/>
  </operation>
</portType>
<binding name="eletutSoapHttp" type="tns:eletut">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="betegeletut">
    <soap:operation soapAction="http://oep.hu//betegeletut"/>
```

```
<input>
  <soap:body use="literal"/>
</input>
<output>
  <soap:body use="literal"/>
</output>
<fault name="BetegletutException">
  <soap:fault name="BetegletutException" use="literal" encodingStyle=""/>
</fault>
</operation>
<operation name="eredmeny">
  <soap:operation soapAction="http://oep.hu//eredmeny"/>
  <input>
    <soap:body use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="BetegletutException">
    <soap:fault name="BetegletutException" use="literal" encodingStyle=""/>
  </fault>
</operation>
<operation name="venytortenet">
  <soap:operation soapAction="http://oep.hu//venytortenet"/>
  <input>
    <soap:body use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="BetegletutException">
    <soap:fault name="BetegletutException" use="literal" encodingStyle=""/>
  </fault>
</operation>
<operation name="venytortenetEredmeny">
  <soap:operation soapAction="http://oep.hu//venytortenetEredmeny"/>
  <input>
    <soap:body use="literal"/>
  </input>
```



```
<output>
  <soap:body use="literal"/>
</output>
<fault name="BetegeletutException">
  <soap:fault name="BetegeletutException" use="literal" encodingStyle=""/>
</fault>
</operation>
</binding>
</binding>
<service name="eletut">
  <port name="eletutSoapHttpPort" binding="tns:eletutSoapHttp">
    <soap:address location="http://tesztjogviszony.oep.hu/eletut/eletut"/>
  </port>
</service>
</definitions>
```

4.2 ÉLES FELÜLET

A <https://jogviszony.oep.hu/ojote/jogviszony> végponthoz tartozó WSDL állomány

```
<definitions
  name="eletut"
  targetNamespace="http://oep.hu/"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:tns="http://oep.hu/"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
>
<types>
  <schema xmlns="http://www.w3.org/2001/XMLSchema" targetNamespace="http://oep.hu/"
    elementFormDefault="qualified"
    xmlns:tns="http://oep.hu/" xmlns:w3="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:soap11-enc="http://schemas.xmlsoap.org/soap/encoding/"
    <complexType name="BetegeletutException">
      <sequence>
        <element name="kod" type="int" nillable="false"/>
      </sequence>
    </complexType>
    <element name="betegeletutElement">
      <complexType>
        <sequence>
          <element name="program" type="string" nillable="true"/>
          <element name="taj" type="string" nillable="true"/>
          <element name="tol" type="string" nillable="true"/>
          <element name="ig" type="string" nillable="true"/>
        </sequence>
      </complexType>
    </element>
    <element name="betegeletutResponseElement">
      <complexType>
```

```
<sequence>
  <element name="result" type="tns:Jegy" nillable="true"/>
</sequence>
</complexType>
</element>
<complexType name="Jegy">
  <sequence>
    <element name="ido" type="long"/>
    <element name="jegy" type="string" nillable="true"/>
  </sequence>
</complexType>
<element name="BetegletutExceptionElement" type="tns:BetegletutException"/>
<element name="eredmenyElement">
  <complexType>
    <sequence>
      <element name="jegy" type="string" nillable="true"/>
    </sequence>
  </complexType>
</element>
<element name="eredmenyResponseElement">
  <complexType>
    <sequence>
      <element name="result" type="tns:Eredmeny" nillable="true"/>
    </sequence>
  </complexType>
</element>
<complexType name="Eredmeny">
  <sequence>
    <element name="venyek" type="tns:Veny" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <element name="ido" type="long"/>
    <element name="ellatasok" type="tns:Ellatas" nillable="true"
minOccurs="0" maxOccurs="unbounded"/>
    <element name="penzbeli" type="tns:Penz" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </sequence>
</complexType>
<complexType name="Veny">
  <sequence>
    <element name="termek" type="string" nillable="true"/>
  </sequence>
</complexType>
```

```
<element name="kivaltas" type="string" nillable="true"/>
<element name="pat_nev" type="string" nillable="true"/>
<element name="ter_dij" type="string" nillable="true"/>
<element name="tamogatas" type="string" nillable="true"/>
<element name="orv_kod" type="string" nillable="true"/>
<element name="pat_cim" type="string" nillable="true"/>
<element name="orv_nev" type="string" nillable="true"/>
</sequence>
</complexType>
<complexType name="Ellatas">
  <sequence>
    <element name="szolg_nev" type="string" nillable="true"/>
    <element name="ellatas_neve" type="string" nillable="true"/>
    <element name="tamogatas" type="string" nillable="true"/>
    <element name="orv_kod" type="string" nillable="true"/>
    <element name="kezdete" type="string" nillable="true"/>
    <element name="szolg_cim" type="string" nillable="true"/>
    <element name="orv_nev" type="string" nillable="true"/>
    <element name="vege" type="string" nillable="true"/>
  </sequence>
</complexType>
<complexType name="Penz">
  <sequence>
    <element name="kezdete" type="string" nillable="true"/>
    <element name="vege" type="string" nillable="true"/>
    <element name="jellege" type="string" nillable="true"/>
  </sequence>
</complexType>
<element name="venytortenetElement">
  <complexType>
    <sequence>
      <element name="program" type="string" nillable="true"/>
      <element name="taj" type="string" nillable="true"/>
      <element name="venyazonosito" type="string" nillable="true"/>
      <element name="szemely_id" type="string" nillable="true"/>
    </sequence>
  </complexType>
</element>
<element name="venytortenetResponseElement">
```

```
<complexType>
  <sequence>
    <element name="result" type="tns:VenyrtortenetJegy"
nillable="true"/>
  </sequence>
</complexType>
</element>
<complexType name="VenyrtortenetJegy">
  <sequence>
    <element name="ido" type="long"/>
    <element name="tranz_kod" type="string" nillable="true"/>
    <element name="jegy" type="string" nillable="true"/>
  </sequence>
</complexType>
<element name="venyrtortenetEredmenyElement">
  <complexType>
    <sequence>
      <element name="jegy" type="string" nillable="true"/>
    </sequence>
  </complexType>
</element>
<element name="venyrtortenetEredmenyResponseElement">
  <complexType>
    <sequence>
      <element name="result" type="tns:VenyrtortenetAdatok"
nillable="true"/>
    </sequence>
  </complexType>
</element>
<complexType name="VenyrtortenetAdatok">
  <sequence>
    <element name="ido" type="long"/>
    <element name="venyrtortenet" type="tns:VenyrtortenetAdat "
nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </sequence>
</complexType>
<complexType name="VenyrtortenetAdat">
  <sequence>
    <element name="ttt_kod" type="string" nillable="true"/>
```



```
</operation>
<operation name="eredmeny">
  <input message="tns:Eletut_eredmeny"/>
  <output message="tns:Eletut_eredmenyResponse"/>
  <fault name="BetegeletutException" message="tns:BetegeletutException"/>
</operation>
<operation name="venytortenet">
  <input message="tns:Eletut_venytortenet"/>
  <output message="tns:Eletut_venytortenetResponse"/>
  <fault name="BetegeletutException" message="tns:BetegeletutException"/>
</operation>
<operation name="venytortenetEredmeny">
  <input message="tns:Eletut_venytortenetEredmeny"/>
  <output message="tns:Eletut_venytortenetEredmenyResponse"/>
  <fault name="BetegeletutException" message="tns:BetegeletutException"/>
</operation>
</portType>
<binding name="eletutSoapHttp" type="tns:eletut">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="betegeletut">
    <soap:operation soapAction="http://oep.hu//betegeletut"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
    <fault name="BetegeletutException">
      <soap:fault name="BetegeletutException" use="literal" encodingStyle=""/>
    </fault>
  </operation>
  <operation name="eredmeny">
    <soap:operation soapAction="http://oep.hu//eredmeny"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>
</service>
```

```
<fault name="BetegeletutException">
  <soap:fault name="BetegeletutException" use="literal" encodingStyle=""/>
</fault>
</operation>
<operation name="venytortenet">
  <soap:operation soapAction="http://oep.hu//venytortenet"/>
  <input>
    <soap:body use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="BetegeletutException">
    <soap:fault name="BetegeletutException" use="literal" encodingStyle=""/>
  </fault>
</operation>
<operation name="venytortenetEredmeny">
  <soap:operation soapAction="http://oep.hu//venytortenetEredmeny"/>
  <input>
    <soap:body use="literal"/>
  </input>
  <output>
    <soap:body use="literal"/>
  </output>
  <fault name="BetegeletutException">
    <soap:fault name="BetegeletutException" use="literal" encodingStyle=""/>
  </fault>
</operation>
</binding>
<service name="eletut">
  <port name="eletutSoapHttpPort" binding="tns:eletutSoapHttp">
    <soap:address location="https://jogviszony.oep.hu/eletut/eletut"/>
  </port>
</service>
</definitions>
```