Position Services

Web Map Component Address

Dokumentace služby

verze 1.0

Copyright © Position s.r.o., Londýnská 665/45, 120 00 Praha 2

1 OBSAH

1 Obsah 2
2 Historie 4
3 Popis 5
4 Obecné struktury 6
4.1 Adresa 6
4.2 Souřadnice 6
5 JavaScript API 7
5.1 Inicializace 7
5.1.1 Funkce PositionServicesReady 7
5.1.2 Globální proměnná PositionServices 7
5.2 Autorizace 7
5.2.1 Login 7
5.2.2 Logout 8
5.2.3 IsLoggedIn 9
5.3 PositionServices.Address 9
5.3.1 SetAddress 9
5.3.2 GetAddress 9
5.3.3 ResetAddress 10
5.3.4 SetProjection 10
5.3.5 SetFields 11
5.3.6 SetRequiredFields 11
5.3.7 SetWarnEmptyFields 12
5.3.8 SetAutoValidate 12
5.3.9 SetAutoSelectBestMatch 12
5.3.10 ConfirmEdit 13
5.3.11 CancelEdit 13
5.3.12 GetEditResult 13
5.3.13 AddEvent 13
5.3.14 RemoveEvent 14
5.3.15 OnEditConfirm 14
5.3.16 OnEditCancel 15
5.4 Příklad použití JavaScript API 16
5.5 Integrace do nativní aplikace 16
5.6 Integrace do HTML aplikace 17

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

6 URL Parametry 19
6.1 Lokalizace 19
6.2 URL API 19
6.2.1 User 19
6.2.2 Password 19
6.2.3 Addr 20
6.2.4 Addr_<složka adresy> 20
6.2.5 AddrPos_<složka souřadnic> 20
6.2.6 Projection 20
6.2.7 Fields 20
6.2.8 RequiredFields 20
6.2.9 WarnEmptyFields 21
6.2.10 AutoValidate 21
6.2.11 AutoSelectBestMatch 21
6.2.12 URLOutput 21
7 AddressHash - ověřovací řetězec adresy 22
7.1 Způsob výpočtu AddressHash 22

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

2 HISTORIE

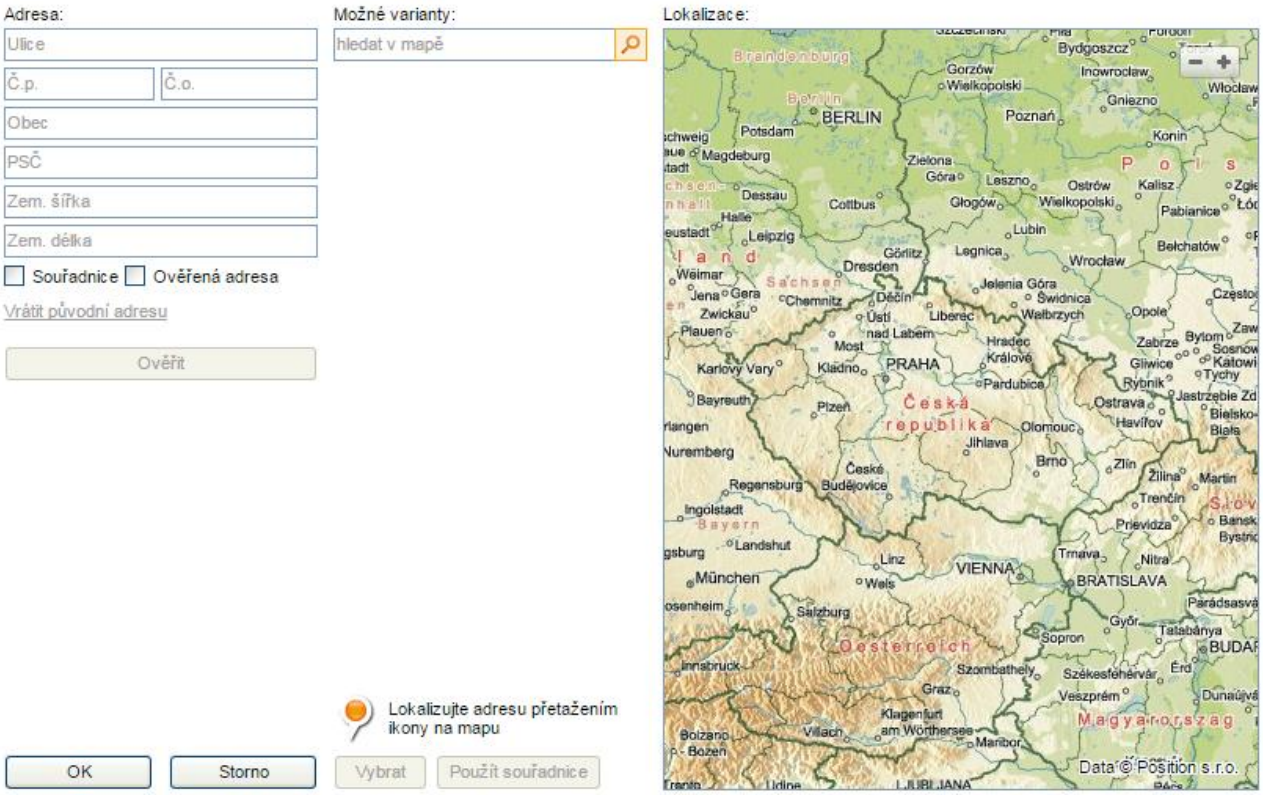
Verze	Publikováno	Autor	Popis
1.0	6.4.2016	J. Tulach	

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

3 POPIS

Služba **Position Services – Web Map Component Address (WMC Address)** poskytuje vizuální mapovou komponentu pro validaci a geocoding adres. Komponenta je realizována formou HTML aplikace (SPA – single page application) a jejím primárním použitím je integrace do dalších aplikací (HTML nebo nativních).

Komponenta poskytuje uživatelské rozhraní, ve kterém je možné na základě vstupu dohledat odpovídající záznamy v databázi adres (ulic, obcí, …) a/nebo lokalizovat místo na základě umístění bodu do mapy.



Bezpečnost je zajištěna pomocí komunikace přes zabezpečený kanál SSL s použitím certifikátu THAWTE.

Zabezpečení přístupu je řešeno pomocí uživatelského jména a hesla.

Služba je dostupná na adrese: **https://services.position.cz/wmc-address/**

Pro integraci lze využít následující možnosti:

JavaScript API	Komunikace je realizována voláním JavaScript funkcí.
URL API	Komunikace je realizována změnou parametrů URL.
Windows DLL (32-bit)	Komunikace je realizována voláním DLL funkcí knihovny. Dostupnost a podmínky - na dotaz.

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

4 OBECNÉ STRUKTURY

4.1 ADRESA

{	
HouseNo	Číslo domu.
OrientationNo	Orientační číslo domu.
Street	Název ulice.
City	Název obce.
CityPart	Název části obce.
PostCode	Poštovní směrovací číslo (PSČ).
County	Název okresu.
State	Název kraje.
Country	Název země.
Pos	Souřadnice (viz 4.2 Souřadnice)
StrPos	Souřadnice zapsané ve formě řetězce.
AddressHash	Ověřovací řetězec adresy (viz 7. AddressHash)
AddressHashTS	Časové razítko vytvoření ověřovacího řetězce adresy.
ID	Unikátní identifikátor adresy.
}	

4.2 SOUŘADNICE

{	
proj	vždy uveden, název projekce
lat	uveden v závislosti na projekci, zeměpisná šířka ve WGS-84
lon	uveden v závislosti na projekci, zeměpisná délka ve WGS-84
x	uveden v závislosti na projekci, x-ová souřadnice
y	uveden v závislosti na projekci, y-nová souřadnice
e	uveden v závislosti na projekci, souřadnice e UTM projekce
n	uveden v závislosti na projekci, souřadnice n UTM projekce
z	uveden v závislosti na projekci, zóna UTM
}	

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

5 JAVASCRIPT API

5.1 INICIALIZACE

5.1.1 FUNKCE POSITIONSERVICESREADY

JavaScript API není k dispozici ihned po načtení obsahu z URL. Důvodem je, že po načtení obsahu ještě nějakou dobu probíhá inicializace (např. načítání dalších zdrojů ze serveru).

Připravenost JavaScript API je indikována vyvoláním JavaScript funkce `PositionServicesReady()`.

Tuto funkci implementuje integrátor komponenty. Funkce může být definována buď přímo v rámci kontextu stránky s komponentou nebo v nadřazeném IFRAME.

Syntaxe:

```
void PositionServicesReady(object api)
```

Parametry:

api - reference na JavaScript API Position Services

Návratová hodnota:

Není.

5.1.2 GLOBÁLNÍ PROMĚNNÁ POSITIONSERVICES

V rámci kontextu stránky s komponentou je po inicializaci JavaScript API dostupná reference na toto API přes globální proměnnou `PositionServices`.

Syntaxe:

```
var PositionServices
```

5.2 AUTORIZACE

Zabezpečení přístupu je řešeno pomocí uživatelského jména a hesla.

5.2.1 LOGIN

Přihlášení uživatele.

Syntaxe:

```
void PositionServices.Login(string username,string password [, callback
onsuccess=null, callback onerror=null])
```

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

Parametry:

username - uživatelské jméno
password - heslo
onsuccess - callback funkce, která je vyvolána při úspěšném přihlášení (volitelné)
onerror - callback funkce, která je vyvolána při chybném přihlášení (volitelné)

Návratová hodnota:

Není.

Callbacky:

Syntaxe:

```
void onsuccess(object api)
void onerror(object api)
```

Parametry:

api - reference na JavaScript API Position Services

Návratová hodnota:

Není.

Příklad použití:

```
function PositionServicesReady(api)
{
  api.Login('someuser','somepassword',
    function(api) {
      api.Address.SetAddress({Street:'Londynska',HouseNo:'665'});
    }
  );
}
```

5.2.2 LOGOUT

Odhlášení uživatele.

Syntaxe:

```
void PositionServices.Logout([callback onerror=null])
```

Parametry:

onerror - callback funkce, která je vyvolána při chybném odhlášení (volitelné)

Návratová hodnota:

Není.

Callbacky:

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

Syntaxe:

```
void onerror(object api)
```

Parametry:

api - reference na JavaScript API Position Services

Návratová hodnota:

Není.

5.2.3 ISLOGGEDIN

Zjištění, zda je uživatel přihlášen.

Syntaxe:

```
bool PositionServices.IsLoggedIn()
```

Návratová hodnota:

TRUE, pokud je uživatel úspěšně přihlášen.

5.3 POSITIONSERVICES.ADDRESS

Část JavaScript API `PositionServices.Address`, týkající se přímo validace adres, je přístupná až po úspěšné autorizaci (viz kapitola Autorizace).

5.3.1 SETADDRESS

Nastavení adresy.

Syntaxe:

```
void PositionServices.Address.SetAddress(object addr)
```

Parametry:

addr - objekt s jednotlivými složkami adresy (viz 4.1 Adresa)

Návratová hodnota:

Není.

5.3.2 GETADDRESS

Získání adresy.

Syntaxe:

```
object PositionServices.Address.GetAddress()
```

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

Návratová hodnota:
Objekt s jednotlivými složkami adresy (viz 4.1 Adresa). Obsah odpovídá aktuálnímu stavu zadání adresy v okamžiku volání funkce.

5.3.3 RESETADDRESS

Vymazání adresy.

Syntaxe:
```
void PositionServices.Address.ResetAddress()
```

Návratová hodnota:
Není.

5.3.4 SETPROJECTION

Nastavení souřadného systému. Výchozí souřadný systém je WGS-84. Metodu `SetProjection` je nutné volat pouze a jen při inicializaci mapové komponenty (v callbacku onsuccess metody `Login`). Pozdější volání metody vyvolá výjimku s oznámením, že mapová komponenta byla již inicializovaná.

Syntaxe:
```
void PositionServices.Address.SetProjection(string proj)
```

Parametry:
proj - souřadný systém (viz níže)

Návratová hodnota:
Není.

Podporované souřadné systémy:
'WGS-84' - souřadný systém WGS-84
'UTM' - souřadný systém UTM
'UTMZone<číslozóny>' - souřadný systém UTM v definované zóně, např. UTMZone33
'WebM' - souřadný systém Web Mercator
'JTSK' - souřadný systém JTSK (Česká Republika)
a další (na dotaz)

Příklad použití:

```
function PositionServicesReady(api)
{
  api.Login('someuser','somepassword',
    function(api) {
      api.Address.SetProjection('UTMZone33');
    }
  );
```

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

```
}
```

5.3.5 SETFIELDS

Nastavení viditelnosti editačních polí mapové komponenty. Metodu `SetFields` je nutné volat pouze a jen při inicializaci mapové komponenty (v callbacku onsuccess metody `Login`). Pozdější volání metody vyvolá výjimku s oznámením, že mapová komponenta byla již inicializovaná.

Syntaxe:

```
void PositionServices.Address.SetFields(array fields)
```

Parametry:

fields - pole se seznamem polí (viz níže)

Návratová hodnota:

Není.

Podporované identifikátory polí:

‘houseno’ - číslo domu
‘orientationno’ - orientační číslo domu
‘street’ - název ulice
‘city’ - název obce
‘citypart’ - název části obce
‘postcode’ - poštovní směrovací číslo (PSČ)
‘county’ - název okresu
‘state’ - název kraje
‘country’ - název země
‘pos’ - souřadnice

Příklad použití:

```
function PositionServicesReady(api)
{
  api.Login('someuser','somepassword',
    function(api) {
      api.Address.SetFields(['houseno','orientationno','street','city','postcode
','county','state','country','pos']);
    }
  );
}
```

5.3.6 SETREQUIREDFIELDS

Nastavení editačních polí mapové komponenty, které musí být vyplněny v okamžiku kliknutí na tlačítko OK.

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

Syntaxe:

```
void PositionServices.Address.SetRequiredFields(array fields)
```

Parametry:

fields - pole se seznamem polí (viz popis metody `SetFields`)

Návratová hodnota:

Není.

5.3.7 SETWARNEMPTYFIELDS

Nastavení editačních polí mapové komponenty, na které je uživatel upozorněn v okamžiku kliknutí na tlačítko OK v případě, že jsou nevyplněna.

Syntaxe:

```
void PositionServices.Address.SetWarnEmptyFields(array fields)
```

Parametry:

fields - pole se seznamem polí (viz popis metody `SetFields`)

Návratová hodnota:

Není.

5.3.8 SETAUTOVALIDATE

Nastavení automatické validace po nastavení adresy (v případě, že nejde o ověřenou adresu). Výchozí hodnota je TRUE.

Syntaxe:

```
void PositionServices.Address.SetAutoValidate(bool state)
```

Parametry:

state - TRUE, pokud se má provádět automatická validace adresy

Návratová hodnota:

Není.

5.3.9 SETAUTOSELECTBESTMATCH

Nastavení automatického výběru v případě, že nalezená adresa jednoznačně odpovídá zadání. Výchozí hodnota je TRUE.

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

Syntaxe:

```
void PositionServices.Address.SetAutoSelectBestMatch(bool state)
```

Parametry:

state - TRUE, pokud se má provádět automatický výběr

Návratová hodnota:

Není.

5.3.10 CONFIRMEDIT

Programové potvrzení editace. Odpovídá kliknutí na tlačítko OK.

Syntaxe:

```
void PositionServices.Address.ConfirmEdit()
```

Návratová hodnota:

Není.

5.3.11 CANCELEDIT

Programové zrušení editace. Odpovídá kliknutí na tlačítko Storno.

Syntaxe:

```
void PositionServices.Address.CancelEdit()
```

Návratová hodnota:

Není.

5.3.12 GETEDITRESULT

Zjištění stavu editace.

Syntaxe:

```
int PositionServices.Address.GetEditResult()
```

Návratová hodnota:

1=editace potvrzena (uživatel kliknul na tlačítko OK), 0=editace zrušena (uživatel kliknul na tlačítko Storno), undefined=editace probíhá

5.3.13 ADDEVENT

Registrace funkce k události.

Syntaxe:

```
void PositionServices.Address.AddEvent(string event,function callback)
```

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816 Důvěrné

Parametry:

event - název události

callback - funkce, která se má k události zaregistrovat

Návratová hodnota:

Není.

5.3.14 REMOVEEVENT

Odregistrace funkce k události.

Syntaxe:

```
void PositionServices.Address.RemoveEvent(string event,function callback)
```

Parametry:

event - název události

callback - funkce, která se má z události odregistrovat

Návratová hodnota:

Není.

5.3.15 ONEDITCONFIRM

Událost, která se vyvolá po potvrzení editace (uživatel kliknul na tlačítko OK).

Syntaxe:

```
void PositionServices.Address.AddEvent('OnEditConfirm',function oneditconfirm)
```

Parametry:

oneditconfirm - funkce, která se má vyvolat po potvrzení editace

Návratová hodnota:

Není.

Callbacky:

Syntaxe:

```
void oneditconfirm(object addrapi)
```

Parametry:

addrapi - reference na JavaScript API PositionServices.Address

Návratová hodnota:

Není.

Příklad použití:

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

```
function PositionServicesReady(api)
{
  api.Login('someuser','somepassword',
    function(api) {
      api.Address.AddEvent('OnEditConfirm',
        function(addrapi) {
          var addr=addrapi.GetAddress();
          // do something
        }
      );
    }
  );
}
```

5.3.16 ONEDITCANCEL

Událost, která se vyvolá po zrušení editace (uživatel kliknul na tlačítko Storno).

Syntaxe:

```
void PositionServices.Address.AddEvent('OnEditCancel',function oneditcancel)
```

Parametry:

oneditcancel - funkce, která se má vyvolat po zrušení editace

Návratová hodnota:

Není.

Callbacky:

Syntaxe:

```
void oneditcancel(object addrapi)
```

Parametry:

addrapi - reference na JavaScript API PositionServices.Address

Návratová hodnota:

Není.

Příklad použití:

```
function PositionServicesReady(api)
{
  api.Login('someuser','somepassword',
    function(api) {
      api.Address.AddEvent('OnEditCancel',
        function(addrapi) {
          // do something
        }
      );
    }
```

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

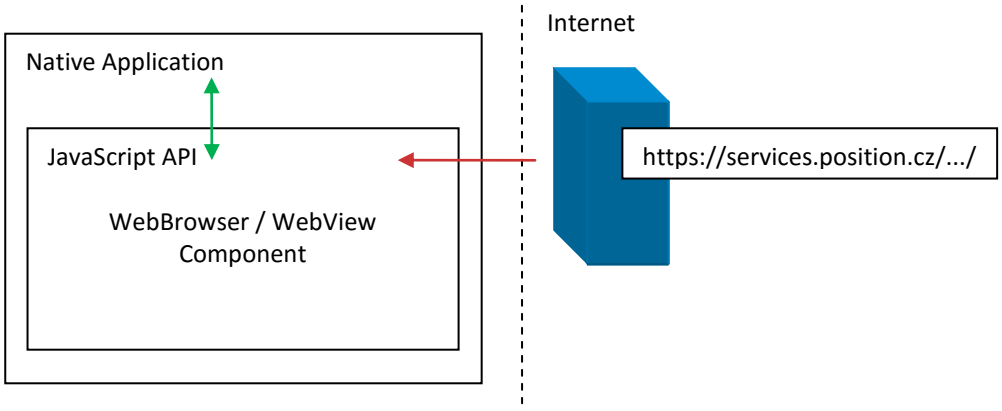
```
  );
}
```

5.4 PŘÍKLAD POUŽITÍ JAVASCRIPT API

```
function PositionServicesReady(api)
{
  api.Login('someuser','somepassword',
    function(api) {

      api.Address.SetAddress({Street:'Londynska',HouseNo:'665'});

       api.Address.AddEvent('OnEditConfirm',
         function(addrapi) {
          var addr=addrapi.GetAddress();
          alert('Edit Success');
          addrapi.ResetAddress();
         }
       );

       api.Address.AddEvent('OnEditCancel',
         function(addrapi) {
          alert('Edit Cancel');
          addrapi.ResetAddress();
         }
       );
    }
  );
}
```

5.5 INTEGRACE DO NATIVNÍ APLIKACE

Nativní aplikace zpravidla umožňují vložit do svého uživatelského rozhraní komponentu s integrovaným webovým prohlížečem (WebBrowser, WebView, …). Kromě načtení a zobrazení obsahu URL je možné z nativní aplikace volat JavaScript funkce a/nebo vytvářet nové JavaScript funkce, které zase zpětně vyvolají funkce nativní aplikace. Způsoby, jak toto nativní aplikace dělají, jsou různé a závisí od použité technologie.



Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

5.6 INTEGRACE DO HTML APLIKACE

5.6.1 INTEGRACE DO HTML POMOCÍ TAGU <SCRIPT>

Mapová komponenta se snadno integruje do jakéhokoliv HTML kódu pomocí tagu `<script>`. Tento tag zajistí načtení JavaScriptových kódů mapové komponenty:

```
<script src="https://services.position.cz/wmc-address/?apploader=1"></script>
```

Mapová komponenta se vytvoří do DIVu s identifikátorem "`PositionServicesAddress`" a přizpůsobí se jeho velikosti:

```
<div id="PositionServicesAddress" style="position: absolute;width:800px;height:600px;"></div>
```

Po úspěšném načtení mapová komponenta provolá inicializační funkci `PositonServicesReady()`:

```
function PositionServicesReady(api)
{
}
```

Po vyvoální inicializační funkce je možno začít používat JavaScript API mapové komponenty.

Mapová komponenta je navržena pro encoding UTF-8:

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

Příklad index.html

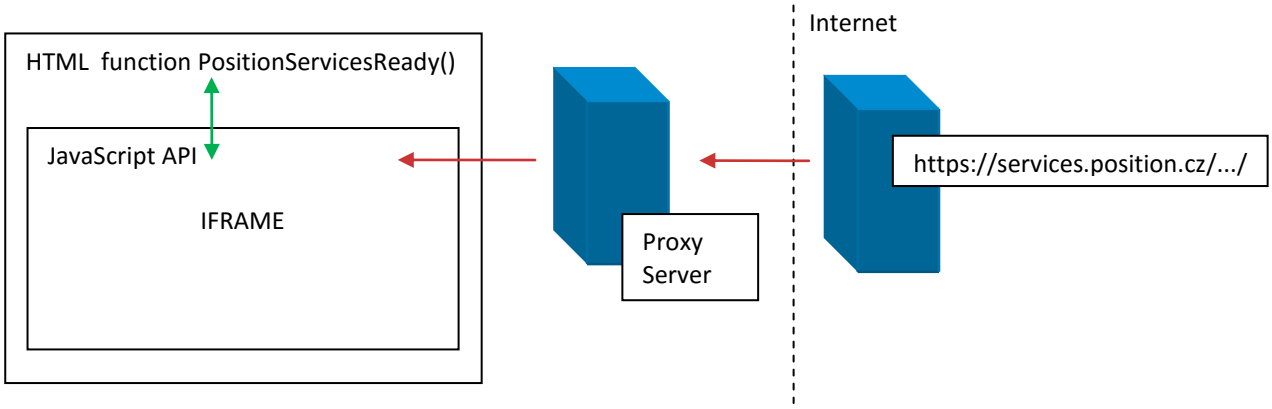
```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <script src="https://services.position.cz/wmc-address/?apploader=1"></script>

    <script type="text/javascript">
      function PositionServicesReady(api)
      {
        api.Login('someuser','somepassword',
          function(api) {

            api.Address.SetAddress({Street:'Londynska',HouseNo:'665'});

            api.Address.AddEvent('OnEditConfirm',
              function(addrapi) {
                var addr=addrapi.GetAddress();
                alert('Edit Success');
                addrapi.ResetAddress();
              }
            );

            api.Address.AddEvent('OnEditCancel',
              function(addrapi) {
                alert('Edit Cancel');
                addrapi.ResetAddress();
              }
            );
          }
        );
```

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

```
      }
    </script>
  </head>
  <body>
    <div id="PositionServicesAddress" style="position: absolute;width:800px;height:600px;"></div>
  </body>
</html>
```

5.6.2 INTEGRACE DO HTML POMOCÍ IFRAME

Dalším možným způsobem integrace mapové komponenty do HTML aplikace je integrace pomocí IFRAME. JavaScript API je na tuto variantu připraveno a tak funkci `PositionServicesReady()`, určenou pro inicializaci mapové komponenty, hledá i ve svých parent oknech.
Jedinou nevýhodou integrace pomocí IFRAME jsou bezpečnostní politiky prohlížečů, které brání volání JavaScript funkcí v případě, že dochází ke křížení domén (cross-site scripting). Toto je možné řešit pomocí proxy serveru, který zajistí, že aplikace i mapová komponenta jsou servírovány ze stejné domény.



6 URL PARAMETRY

6.1 LOKALIZACE

Mapová komponenta podporuje lokalizaci do různých jazyků. Ve výchozím stavu je jazyk detekován na základě nastavení prohlížeče a dostupných lokalizací.

Lokalizaci je možné vynutit i pomocí URL parametru `lang`.

Příklad:
https://services.position.cz/.../?lang=en

Podporované lokalizace:
cz - čeština
en - angličtina
a další (na dotaz)

6.2 URL API

Místo JavaScript API je možné s mapovou komponentou komunikovat i přes parametry v URL. Tyto parametry jsou uvedeny v URL za # a oddělují se pomocí znaku @.
Změna hodnot parametrů se (až na výjimky) okamžitě projeví v mapové komponentě.

Kódování parametrů je UTF-8.

Příklad URL:
https://services.position.cz/wmc-address/#Addr_Street=Londynska@Add_HouseNo=665@Addr_OrientationNo=45@Addr_City=Praha@User=test@Password=test

6.2.1 USER

Název parametru: User
Hodnota: Uživatelské jméno.

Použije se při inicializaci komponenty pro autorizaci. Nereaguje na změnu.

6.2.2 PASSWORD

Název parametru: Password
Hodnota: Heslo.

Použije se při inicializaci komponenty pro autorizaci. Nereaguje na změnu.

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

6.2.3 ADDR

Název parametru: Addr
Hodnota: Adresa ve formátu JSON (viz 4.1 Adresa).

Nastavení validované adresy.

6.2.4 ADDR_<SLOŽKA ADRESY>

Název parametru: Addr_<složka adresy>
Hodnota: Dle typu složky.

Složky adresy: viz 4.1 Adresa

6.2.5 ADDRPOS_<SLOŽKA SOUŘADNIC>

Název parametru: AddrPos_<složka souřadnic>
Hodnota: Dle typu složky.

Složky souřadnic: Dle typu souřadného systému - viz 4.2 Souřadnice.

6.2.6 PROJECTION

Název parametru: Projection
Hodnota: Souřadný systém.

Nastavení souřadného systému. Výchozí souřadný systém je WGS-84. Nereaguje na změnu.
Seznam podporovaných souřadných systémů – viz metoda `SetProjection` v JavaScript API.

6.2.7 FIELDS

Název parametru: Fields
Hodnota: Seznam polí oddělených středníkem

Nastavení viditelnosti editačních polí mapové komponenty. Nereaguje na změnu. Seznam polí – viz metoda `SetFields` v JavaScript API.

6.2.8 REQUIREDFIELDS

Název parametru: RequiredFields
Hodnota: Seznam polí oddělených středníkem

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

Nastavení editačních polí mapové komponenty, které musí být vyplněny v okamžiku kliknutí na tlačítko OK. Seznam polí – viz metoda `SetFields` v JavaScript API.

6.2.9 WARNEMPTYFIELDS

Název parametru: WarnEmptyFields
Hodnota: Seznam polí oddělených středníkem

Nastavení editačních polí mapové komponenty, na které je uživatel upozorněn v okamžiku kliknutí na tlačítko OK v případě, že jsou nevyplněna. Seznam polí – viz metoda `SetFields` v JavaScript API.

6.2.10 AUTOVALIDATE

Název parametru: AutoValidate
Hodnota: 0 – vypnuto, 1 – zapnuto (výchozí)

Nastavení automatické validace po nastavení adresy (v případě, že nejde o ověřenou adresu).

6.2.11 AUTOSELECTBESTMATCH

Název parametru: AutoSelectBestMatch
Hodnota: 0 – vypnuto, 1 – zapnuto (výchozí)

Nastavení automatického výběru v případě, že nalezená adresa jednoznačně odpovídá zadání.

6.2.12 URLOUTPUT

Název parametru: URLOutput
Hodnota:
json – adresa se uloží do parametru Addr
urlparams – adresa se uloží do parametrů Addr_*, AddrPos_*
none – vypnuto (výchozí)

Nastavuje uložení adresy do URL parametrů po ukončení editace. Kromě adresy se do URL parametrů uloží i parametr EditResult s hodnotou:
1 – uživatel kliknul na tlačítko OK,
0 – uživatel kliknul na tlačítko Cancel

Poznámka: V ramci integrace je možné monitorovat změny URL parametrů a v případě, že se v parametrech objeví parametr EditResult, tak integrace ví, že došlo k ukončení editace a dle nastavení si z parametrů načte editovanou adresu.

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

7 ADDRESSHASH - OVĚŘOVACÍ ŘETĚZEC ADRESY

Mapová komponenta pracuje s termínem ověřená adresa.

Za ověřenou adresu se považuje taková adresa, která byla získána na základě dohledání v databázi adres, ulic, …

Za ověřenou adresu se naopak nepovažuje adresa, kterou uživatel jakkoliv ručně upraví nebo byla dohledána na základě ručního umístění bodu v mapě. Důvodem je, že souřadnice takto umístěného bodu nemusí odpovídat souřadnicím v databázi adres, ulic, …

Při validaci adresy mapová komponenta ví, odkud byla adresa získána a zda byla uživatelem upravována. Může proto snadno zobrazit, zda jde nebo nejde o ověřenou adresu. V okamžiku, kdy si validovanou adresu převezme např. informační systém a uloží ji do svých databází, se informace o ověřenosti adresy ztrácí. Navíc, pokud informační systém umožňuje nezávislou editaci jednotlivých složek adresy, může dojít k tomu, že např. při změně čísla popisného zůstanou u adresy původní souřadnice. Jednoduše tak vznikne adresa, která má nesprávné souřadnice, popř. vůbec neexistuje.

Mapová komponenta pro validaci adres tento problém řeší pomocí tzv. ověřovacího řetězce adresy – AddressHash. V případě, že jde o ověřenou adresu, je tento ověřovací řetězec součástí výstupu mapové komponenty. Pokud si ho informační systém uloží do své databáze společně s ostatními složkami adresy, tak při pozdějším zobrazení adresy v mapové komponentě je možné určit, zda jde o ověřenou adresu a že nedošlo k její jakékoliv modifikaci.

7.1 ZPŮSOB VÝPOČTU ADDRESSHASH

Zda jde o ověřenou adresu, která nebyla jakkoliv modifikována, si může kdokoliv ověřit. Udělá to tak, že si sám spočítá ověřovací řetězec a porovná ho s hodnotou vrácenou z mapové komponenty.

Pro výpočet ověřovacího výpočtu se používají následující složky adresy (seřazeny jsou podle abecedy):
City, CityPart, HouseNo, OrientationNo, PostCode, Street, Pos

Jednotlivé hodnoty složek adresy se seskládají do řetězce a oddělí svislítkem |. Do složky souřadnic (Pos) se vyskládají složky souřadnic (bez souřadného systému) oddělené středníkem. Výsledný řetězec se obalí XML tagem <ValidAddr> a z výsledku se spočítá MD5 hash.

Příklad:
Adresa:

```
{
  HouseNo: 665,
  OrientationNo: 45,
  Street: 'Londýnská',
  City: 'Praha 2',
```

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816

```
  PostCode: '120 00',
  Pos: {
    proj: 'WGS-84'
    lat: 50.073785,
    lon: 14.433811
  }
}
```

AddressHash:
MD5("<ValidAddr>Praha 2||665|45|120 00|Londýnská|50.073785;14.433811</ValidAddr>")

Copyright © Position s.r.o., IČO: 26422816, DIČ: CZ26422816