

## Informační systém základních registrů



# ISZR Referenční agent .NET

Název dokumentu:	ISZR Referenční agent .NET
Projekt:	ISZR
Autor/Autoři:	Pavel Odstrčil

Verze:	1.04
Stádium:	Pracovní
Důvěrnost:	Pro partnery

Jméno souboru:	ISZR Referenční agent - NET v1.04.docx
Datum aktualizace:	2.10.2017 15:34:00

Počet stran:	12
Strana:	1

**Projekt Informační systém základních registrů (registrační číslo: CZ.1.06/1.1.00/03.05891) byl spolufinancován z prostředků Evropské unie, Evropského fondu pro regionální rozvoj prostřednictvím Integrovaného operačního programu.  
Šance pro Váš rozvoj.**

### Historie verzí

Datum	Verze	Popis	Zpracoval
15. 3. 2012	1.00	Vytvoření dokumentu	AutoCont CZ a.s.
29. 9. 2017	1.04	Podpora protokolu TLS 1.2	AutoCont CZ a.s.

### Použité zkratky

Zkratka	Popis
AIS	Agendový informační systém
ISoISVS	Informační systém o informačních systémech veřejné správy
ISZR	Informační systém základních registrů
RA	Referenční agent
ROB	Registr obyvatel
ROS	Registr osob
RUIAN	Registr územní identifikace, adres a nemovitostí
UI	User interface – uživatelské rozhraní
WS	Webová služba

**Obsah:**

<b>1.</b>	<b>Úvod</b> .....	<b>4</b>
<b>2.</b>	<b>Copyright a záruka</b> .....	<b>4</b>
<b>3.</b>	<b>Referenční agent pro prostředí .NET Framework</b> .....	<b>4</b>
3.1	Úvod .....	4
3.2	Uživatelské rozhraní .....	5
3.2.1	UI panel – Základní .....	5
3.2.2	UI panel – Systémová data .....	6
3.2.3	UI panel – ROB .....	6
3.2.4	UI panel – ROS .....	6
3.2.5	UI panel – ISZR .....	6
3.2.6	UI panel – Přístup k prostředí .....	7
3.2.7	UI panel – Volání .....	7
3.2.8	Proxy .....	7
3.3	Struktura projektu .....	9
3.3.1	Konfigurační soubor aplikace .....	9
3.3.2	Demo certifikát .....	9
3.4	Začlenění eGON webové služby .....	9
<b>4.</b>	<b>Shrnutí</b> .....	<b>12</b>

## 1. Úvod

Pro účely zjednodušení a urychlení implementace AIS v oblasti připojení na Informační systém základních registrů vznikl referenční agent, který demonstruje některé možné způsoby připojení. Konkrétně jde o připojení z prostředí Java a prostředí .NET.

Pro obě prostředí vznikly jednoduché aplikace, které demonstrují způsob volání webových služeb na eGON rozhraní. Aplikace si nekladou za cíl postihnout problematiku použití v libovolné konfiguraci, demonstrují základní principy a způsoby použití rozhraní.

Tento dokument popisuje část implementace referenčního agenta pro prostředí .NET.

Obecný popis referenčního agenta (použité služby, certifikáty, testovací data, a podobně) je v samostatném dokumentu – „ISZR Referenční agent.docx“.

Popis referenčního agenta pro prostředí Java je v samostatném dokumentu.

## 2. Copyright a záruka

Zdrojový kód aplikace i přeloženou aplikaci lze použít libovolným způsobem. Tvůrce aplikace a ostatních částí (aplikace, zdrojové kódy, dokumentace, certifikát, ostatní soubory a další) není v žádném ohledu odpovědný za jakýkoliv důsledek přímo nebo nepřímo vzniklý v souvislosti s libovolnou částí díla.

## 3. Referenční agent pro prostředí .NET Framework

Tato kapitola obsahuje popis referenčního klienta pro prostředí Microsoft .NET Framework. Referenční klient je *WinForms* aplikace, která umožňuje volání vybraných eGON služeb ISZR.

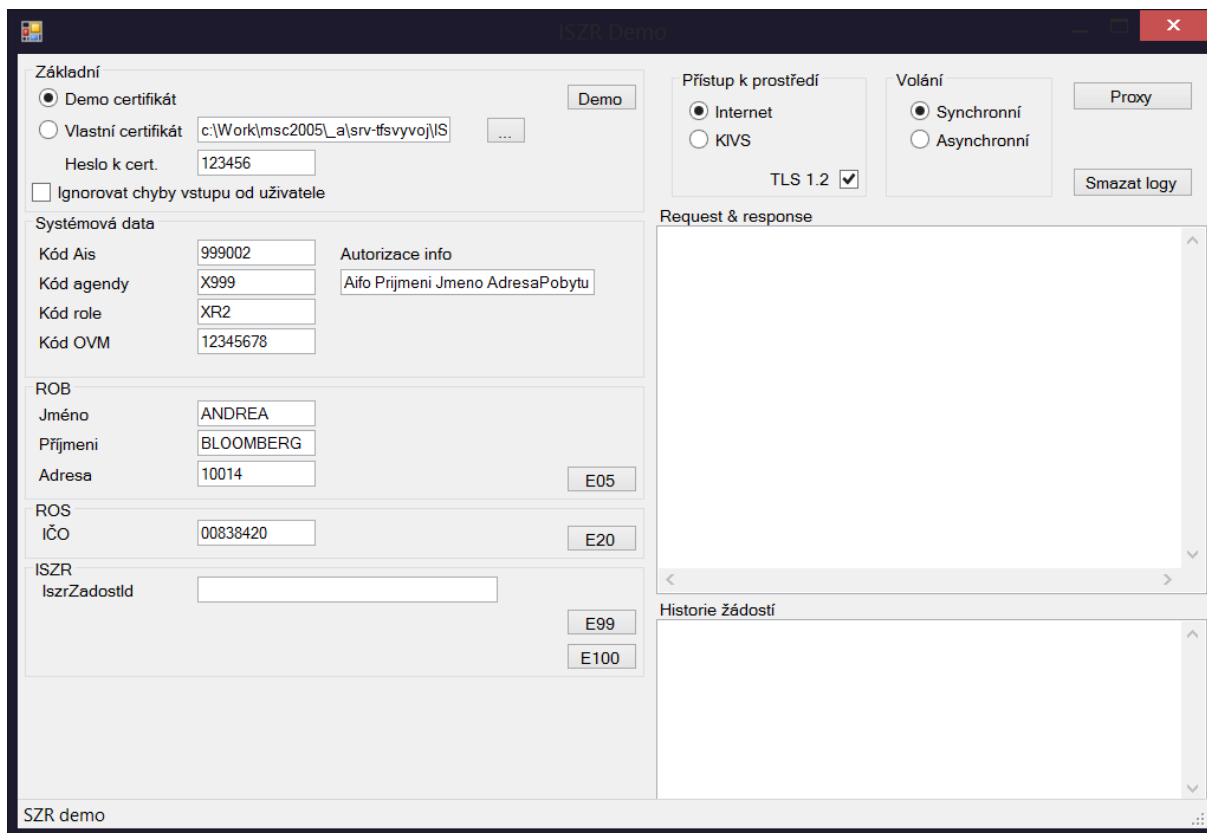
### 3.1 Úvod

RA pro prostředí .NET je připraven pro prostředí .NET Framework 4.0. Je připraven jako solution pro vývojové prostředí Microsoft Visual Studio 2013. Součástí připravené solution jsou:

- Komentované zdrojové kódy
- Testovací certifikát pro přístup k testovacímu prostředí publikačnímu

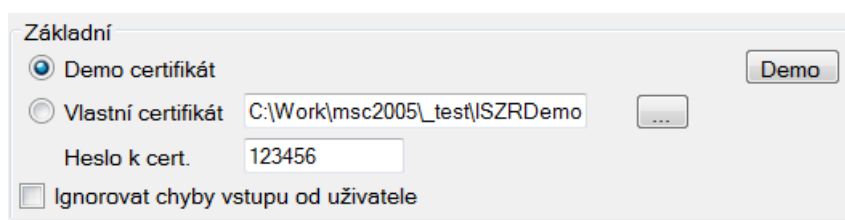
## 3.2 Uživatelské rozhraní

Na následujícím obrázku je znázorněno uživatelské rozhraní referenčního agenta v prostředí .NET.



Uživatelské rozhraní je rozděleno na několik základních panelů, které jsou popsány v následujících kapitolách.

### 3.2.1 UI panel – Základní



V tomto panelu:

- Je možné zadat certifikát, který bude použit pro přístup k eGON rozhraní. Je možné použít vestavěný certifikát (zkompilovaný do aplikace – označený jako „Demo“), nebo vybrat certifikát umístěný na disku.
- Je možné nastavit výchozí hodnoty ve všech polích formuláře, pomocí tlačítka *Demo*.
- Je možné zvolit ignorování chyb na vstupu od uživatele. RA před voláním eGON služeb kontroluje nastavení systémových dat volání a v případě chyb toto volání neprovede a na chybu upozorní. Při nastavení této možnosti je možné provést volání i s nevyplněnými hodnotami systémové hlavičky.

### 3.2.2 UI panel – Systémová data

Systémová data		Autorizace info	
Kód Ais	<input type="text" value="99001"/>	Aifo Příjmeni Jmeno AdresaPobytu	
Kód agendy	<input type="text" value="A999"/>	Obsah pro element AutorizaceInfo bude specifikován v dokumentaci rozhraní. Uvedené hodnoty jsou předběžné.	
Kód role	<input type="text" value="CTENAR"/>		
Kód OVM	<input type="text" value="12345678"/>		

V tomto panelu se zadávají data pro systémovou hlavičku odesílanou přes webovou službu. Jde o tyto údaje:

- *Kód AIS* – je přidělen při vydávání certifikátu pro přístup k eGON rozhraní (obvykle ID AIS v ISOISVS).
- *Kód agendy* – kód agendy, pro jejíž výkon je volána WS na eGON rozhraní ISZR.
- *Kód role* - kód role, pro jejíž výkon je volána WS na eGON rozhraní ISZR.
- *Kód OVM* – OVM vykonávající činnost, pro jejíž účely je volána WS
- *Autorizace info* – informace, které určují požadavek na výdej údajů a způsob zpracování služby v ISZR. Seznam povolených hodnot je uveden v dokumentaci eGON rozhraní na webu SZR.

### 3.2.3 UI panel – ROB

ROB	
Jméno	<input type="text" value="ANDREA"/>
Příjmeni	<input type="text" value="BLOOMBERG"/>
Adresa	<input type="text" value="10014"/>
<input type="button" value="E05"/>	

V tomto panelu se zadávají vybraná data pro volání služby *E05 – robCtiPodleUdaju*. Termínem vybraná data se rozumí, že služba E05 podporuje různé způsoby vyhledávání, pro účely RA bylo jako příklad zvoleno vyhledání dle jména, příjmení a adresy (referenční odkaz do RUAIN). Po stlačení tlačítka *E05* je služba vyvolána.

### 3.2.4 UI panel – ROS

ROS	
IČO	<input type="text" value="00838420"/>
<input type="button" value="E20"/>	

V tomto panelu je možné volat službu ROS – *E20 – rosCtilco*. Pro volání je třeba zadat IČO, po stlačení tlačítka *E20* je služba vyvolána.

### 3.2.5 UI panel – ISZR

ISZR	
IszrZadostId	<input type="text"/>
<input type="button" value="E99"/>	
<input type="button" value="E100"/>	

V tomto panelu je možné volat služby ISZR:

- E99 – IszrAsyncVypisFronty
- E100 – IszrAsyncOdpovedZFronty

Služba *E99* vypisuje obsah fronty. V referenčním agentovi je implementována bez parametrů, vypíše tedy definovaný (interně v ISZR) počet záznamů z fronty. Ve výsledku služby je uvedeno, zda byla fronta vypsána celá, nebo zda výpis pokračuje. Další dávku z fronty lze získat, pokud by v parametrech volání byl uveden identifikátor počátku výpisu (v RA toto implementováno není).

Služba *E100* vyzvedává z fronty výsledek služby specifikované pomocí *IszrZadostId*.

### 3.2.6 UI panel – Přístup k prostředí

V tomto panelu je možné nastavit, ve kterém prostředí je volání služby prováděno. RA předpokládá volání publikačního testovacího prostředí, a to jak ve variantě řízeného internetu, tak i ve variantě KIVS.

Dále se v tomto panelu nastavuje podpora komunikace s použitím protokolu TLS 1.2. Starší, dříve podporované protokoly TLS 1.1 a 1.0 byly v testovacím prostředí ISZR vypnuty 1. 8. 2017. Pokud není protokol na straně klienta povolen, nenaváže klient komunikaci s ISZR.

Implementace je provedena v původním kódu RA pro .NET 4.0, kdy ještě nebyl protokol TLS 1.2 standardně podporován. Platí:

- .NET 4.6 a vyšší – podporováno, default TLS 1.2.
- .NET 4.5 – podporováno, TLS 1.2 není default
- .NET 4.0 – standardně nepodporováno, lze realizovat za předpokladu že podporuje OS a je nainstalován .NET 4.5 nebo vyšší.

### 3.2.7 UI panel – Volání

V tomto panelu je možné nastavit mód volání služby – synchronní nebo asynchronní. Toto nastavení má vliv logicky pouze na služby, u kterých je to definováno v katalogu služeb (tedy v RA jsou to služby E05 a E20).

V případě, že je služba zavolána jako asynchronní, vrátí se jako výsledek přidělený identifikátor *IszrZadostId*. Pomocí tohoto identifikátoru lze pak výsledek služby vyzvednout.

### 3.2.8 Proxy

Tlačítko *Proxy* slouží pro nastavení přístupu přes proxy.

Aplikace umí přistupovat přímo nebo použít proxy server nastavený v systému Windows. Při přístupu přes proxy server je možné zvolit buď autentizaci dle aktuálně přihlášeného uživatele (*DefaultNetworkCredentials*), nebo je možné přihlášení na proxy zadat ručně.

Hodnoty v tomto dialogu mohou být přednastaveny v konfiguračním souboru aplikace.

Tlačítko *Smazat logy* slouží pro smazání textových polí *Request & response* a *Historie žádostí*.

V textovém poli *Request & response* se zobrazují XML zprávy odesílané na eGON rozhraní a přijímané jako odpovědi z eGON rozhraní.

V textovém poli *Historie žádostí* se zobrazuje informace volané službě (kód služby), výsledek služby (OK / CHYBA / VAROVANI) a identifikátor, který byl volání služby přidělen (*IszrZadostId*).

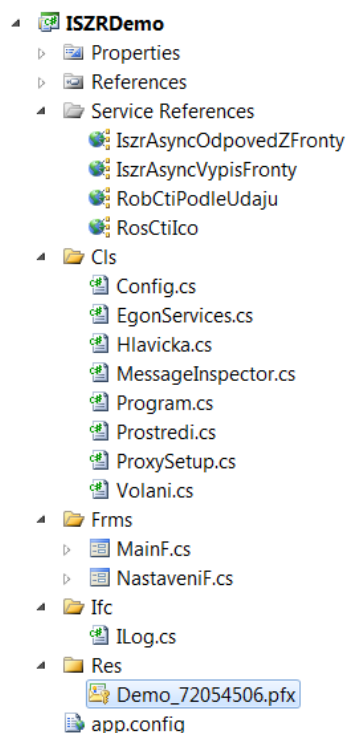




### 3.3 Struktura projektu

Obsah solution *ISZRDemo.sln* je popsán v následujícím odstavci.

- *Service References* – importované rozhraní webových služeb
- *Cls*
  - o *Config.cs* – čtení konfiguračního souboru aplikace
  - o *EgonServices.cs* – volání eGON služeb a pomocné metody pro volání
  - o *Hlavicka.cs* – data pro systémovou část zprávy jako AIS, agenda, agendová role, ...
  - o *MessageInspector.cs* – pomocná třída pro logování komunikace
  - o *Program.cs* – startovací třída aplikace
  - o *Prostredi.cs* – třída pro definici způsobu volání, prostředí, proti kterému se volá, konstrukce URL
  - o *ProxySetup.cs* – třída pro definici přístupu při připojení přes proxy
  - o *Volani.cs* – třída pro předávání parametrů mezi UI a business logikou
- *Frms*
  - o *MainF.cs* – hlavní formulář aplikace
  - o *NastaveniF.cs* – formulář nastavení (proxy, ...)
- *Ifc*
  - o *ILog.cs* – interface logování
- *Res*
  - o *Demo\_72054506.pfx* – testovací referenční certifikát
- *app.config* – konfigurační soubor aplikace



#### 3.3.1 Konfigurační soubor aplikace

V konfiguračním souboru aplikace lze v elementu *appSettings* nastavit hodnoty uvedené v tabulce dále. Tyto hodnoty představují výchozí nastavení, které lze nastavit přímo v UI referenčního agenta.

Key	Typ	Popis
Proxy.UseSystem	Bool	Použít proxy nastavenou v systému
Proxy.UseDefaultNetworkCredentials	Bool	Při použití proxy použít credentials aktuálního bezpečnostního kontextu
Proxy.UserName	String	Uživatelské jméno pro ověření na proxy
Proxy.Password	String	Heslo pro ověření na proxy
Proxy.Domain	String	Doména pro ověření na proxy

#### 3.3.2 Demo certifikát

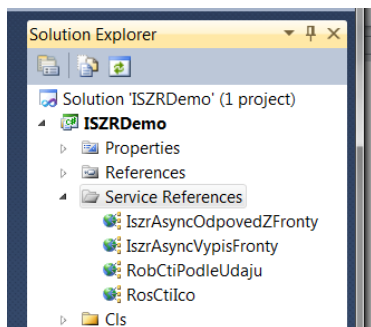
Demo certifikát je obsažen v řešení ve složce *Res* - soubor *Demo\_72054506.pfx*. Má nastavenou *Build Action* = *Embedded Resource*, a *Copy To Output*. Certifikát obsažený v řešení je pouze vzorový, jeho platnost vypršela a nemůže být použit pro komunikaci s ISZR. V řešení jej lze nahradit vlastním certifikátem nebo referenčním certifikátem, který lze získat postupem uvedeným na stránkách SZR.

### 3.4 Začlenění eGON webové služby

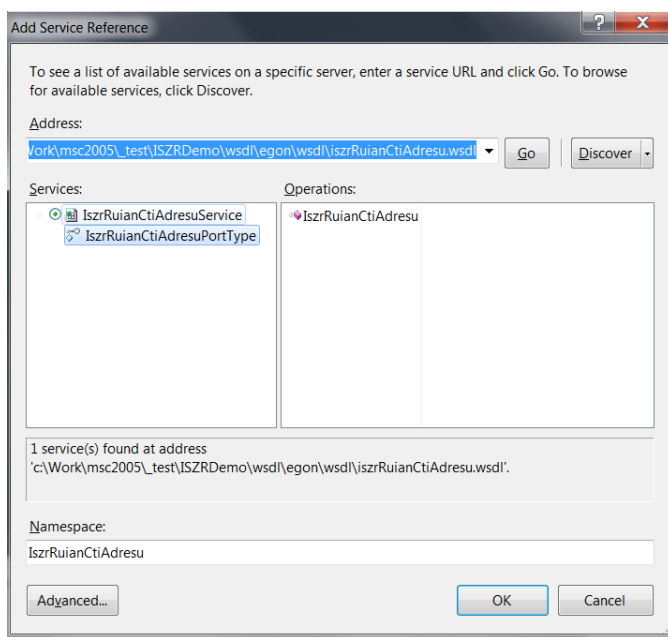
V této kapitole je popsán příklad začlenění další služby ZR. V příkladu je začleněna služba *E36 – ruainCtiAdresu*.

## 1. Načtení WS

V prvním kroku se do projektu načte definice WS.

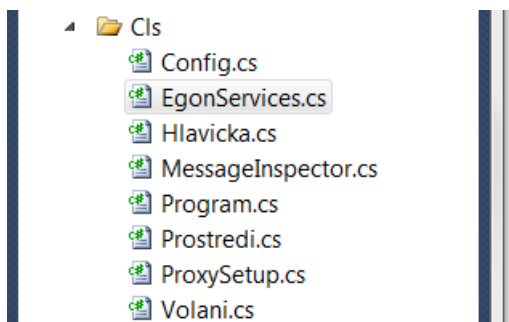


V Solution exploreru nad větví Service References se zvolí Add Service Reference.



## 2. Vytvoření služby

Nyní je třeba vytvořit kód pro volání služby.



V souboru *EgonServices.cs* využijeme stávající kód, zkopírujeme a upravíme pro službu E36. Vznikne následující kód:

```

//-----
/// <summary>
/// E36 - ruianCtiAdresu
/// </summary>
    
```

```

/// <param name="volani"></param>
/// <param name="kod"></param>
internal static void E36(Volani volani, int kod)
{
    String svc = "E36";
    Binding bind = null;
    EndpointAddress adr = GetEgonAddr(volani, svc, true, out bind);
    //
    IszrRuianCtiAdresu.IszrRuianCtiAdresuPortTypeClient c1 =
        new IszrRuianCtiAdresu.IszrRuianCtiAdresuPortTypeClient(bind, adr);
    //
    c1.ClientCredentials.ClientCertificate.Certificate = volani.Certifikat;
    //
    IszrRuianCtiAdresu.RuianCtiAdresuType req = new IszrRuianCtiAdresu.RuianCtiAdresuType();
    //
    req.ZadostInfo = new IszrRuianCtiAdresu.ZadostInfoIszrType();
    FillSystemPart(volani, req.ZadostInfo);
    // při čtení ROB je vyžadováno - zde čteme RUIAN, není třeba vyplňovat
    // req.ZadostInfo.Subjekt = "Subjekt";
    // req.ZadostInfo.Uzivatel = "Uzivatel1156";
    // req.ZadostInfo.DuvodUcel = "Duvod a ucel";
    //
    req.AutorizaceInfo = new IszrRuianCtiAdresu.AutorizaceType();
    req.AutorizaceInfo.SeznamUdaju = volani.Hlavicka.AutorizaceInfo;
    //
    req.Zadost = new IszrRuianCtiAdresu.RuianCtiAdresuTypeZadost();
    req.Zadost.RuianCtiAdresuData = new IszrRuianCtiAdresu.CtiAdresuDotazType();
    req.Zadost.RuianCtiAdresuData.Kod = kod;
    req.Zadost.RuianCtiAdresuData.FormatAdresy = IszrRuianCtiAdresu.FormatAdresyType.RADKOVY;
    //
    c1.Endpoint.Behaviors.Add(new MessageLoggerBehavior(volani.Log));
    IszrRuianCtiAdresu.RuianCtiAdresuResponseType resp = c1.IszrRuianCtiAdresu(req);
    LogHistory(svc, volani, resp);
}

```

### 3. Začlenění do UI

Upravíme UI – formulář *MainF* pro volání služby E36. Přidáme tlačítko, pole pro vstup a kód pro obsluhu události při stlačení tlačítka.

The screenshot shows a window titled 'ISZR'. It contains two main sections. The first section is labeled 'lszrZadostId' and has a text input field followed by two buttons labeled 'E99' and 'E100'. The second section is labeled 'RUIAN' and has a text input field labeled 'Kód adresy' followed by a button labeled 'E36'.

Kód pro obsluhu události tlačítka bude obdobný jako u tlačítek pro volání služeb ROB, ROS a ISZR.

```

//-----
/// <summary>
/// tlačitko E36
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void btRuain_Click(object sender, EventArgs e)
{
    try
    {
        if (!CheckNotNull(tbAdr, typeof(int))) return;
        Volani volani = new Volani(this, CtiProstredi(), CtiCertifikat(), CtiSystem());
        EgonServices.E36(volani, Convert.ToInt32(tbAdr.Text.Trim()));
    }
}

```

```
    catch (Exception ex)
    {
        Log(ex.Message);
    }
}
```

## 4. Shrnutí

V dokumentu byl popsán referenční agent jako vzor pro volání eGON služeb ZR. Dále byl příklad implementace volání eGON služby. Cílem referenčního agenta bylo poskytnout příklad přístupu k uvedené problematice. Možných přístupů samozřejmě existuje celá řada, závisí na kontextu, ve kterém bude implementace prováděna.