



Indagor Service

• ST012 •

ACQUISIZIONE PUNTUALE DOMICILI DIGITALI DA INAD

Specifiche di Integrazione

INDICE

Descrizione.....	1
Contenuto della Richiesta.....	2
Contenuto delle Eccezioni.....	3
Contenuto della Risposta.....	4
Esempi di Integrazione.....	5
API Java.....	5
API .Net (C#).....	7
API Javascript.....	8

ST012

Dati di Domicili Digitali da INAD

DESCRIZIONE

Il servizio consente, dato un Codice Fiscale di un soggetto, di acquisire in INAD - in sincrono - i seguenti gruppi di dati:

- Domicili Digitali

Qui di seguito è riportato il contenuto

- Della Richiesta
- Delle Eccezioni
- Della Risposta

Segue un esempio di integrazione completo per:

- Ambiente di sviluppo JAVA
- Ambiente di sviluppo NET
- Javascript

CONTENUTO DELLA RICHIESTA

Classe Client:

Costruttore:

Client(String hostport, String utente, String password)

- **hostport:** radice dell'end point
 - Ambiente di test: <http://84.240.190.236>
 - Ambiente di produzione: <https://gateway1.steseiservizi.it>
- **utente:** Identificativo Gateway assegnato da Stesei all'applicativo per l'autenticazione delle chiamate effettuate
 - Ambiente di test: **040007**
- **password:** password Gateway associata all'identificativo: la password è gestita dal produttore dell'applicativo attraverso il portale INDAGOR (<http://www.steseiservizi.it/indagor-pdnd>)
 - Ambiente di test: **"INIZIALE"**

Metodo:

Richiesta per Codice Fiscale

RespST012 ST012(String idOperatore, int idUfficio, String comunelstat, String codiceFiscale, String motivoRichiesta)

- **ST012:** Nome dell'Indagor Service richiamato
- **comunelstat:** Codice Istat del Comune fruitore dell'applicativo che inoltra la chiamata
 - Ambiente di test: Codice Istat **040007**
- **codiceFiscale:** Codice Fiscale del cittadino
 - Ambiente di test: Codice Fiscale (Esempio **TRVVCN73H02L259I**)
- **motivoRichiesta:** Testo libero (ad esempio "Numero Pratica" a cui si riferisce l'acquisizione dati)

CONTENUTO DELLE ECCEZIONI

Eccezioni restituite da tutti i metodi del Gateway INDAGOR:

- **ConnectionException**: errore di connessione o di server non attivo
- **ServerException**: errori interni del server
- **AuthenticationException**: errore di autenticazione (utente e/o password errati)
- **PermissionDeniedException**:
 - operazione non permessa (es: comune non censito nel Gateway o Codice Istat del Comune vuoto)
- **RequestException**:
 - Se i parametri della ricerca inseriti non sono corretti:
 - "Il parametro Codice Istat ha un formato non valido"
 - "Il parametro Codice Fiscale ha un formato non valido"
 - Se manca il criterio di ricerca
 - "Criteri di ricerca non validi"
- **"INADException"**: errori e anomalie restituite da INAD a fronte della ricerca effettuata

CONTENUTO DELLA RISPOSTA

Di seguito sono riportati i dati che compongono la risposta resa dal servizio ST012; se i valori della risposta sono pari a:

- **null**
- **stringa vuota**
- **LocalDate.MIN**

significa che tali dati non sono presenti in INAD

Descrizione		Note
Codice Fiscale		
Data Estrazione		Formato aaaa-mm-gg
Domicilio Digitale 1		
Indirizzo Digitale 1	Indirizzo Digitale ID1	
	Professione ID1	
	Data Fine Validità ID1	Formato aaaa-mm-gg
	Motivo Cessazione ID1	
Domicilio Digitale 1		

Vedi Domicilio Digitale sopra

ESEMPI DI INTEGRAZIONE

API Java

Si aggiunge al progetto la libreria:

- **steseigateway-1.0.0-jar-with-dependencies.jar**

O in alternativa le seguenti librerie:

- **steseigateway-1.0.0.jar**
- **jackson-core-2.12.1.jar**
- **jackson-databind-2.12.1.jar**
- **jackson-annotations-2.1.2.jar**

Si richiama nel seguente modo:

```
package main;
import stesei.gateway.*;
import java.io.IOException;
import java.time.LocalDate;
public class Test{
    public static void main(String[] args){
        TestST012();
    }
    public static void TestST012(){
        Client cl = new Client("http://84.240.190.236", "040007", "INIZIALE");
        RespST012 resp=null;
        try{
            resp = cl.ST012("040007", "TRVVCN73H02L259I", "Pratica N. 987654321");
        }

        catch (ConnectionException econn){
            String msg=econn.getMessage(); //errore di connessione al Gateway
            System.out.println(msg);
        }
        catch (ServerException eserver){
            String msg=eserver.getMessage(); //errore interno del server
            System.out.println(msg);
        }
        catch (AuthenticationException eauth){
            String msg=eauth.getMessage(); //utente e/o password errati
            System.out.println(msg);
        }
    }
}
```

```
catch (PermissionDeniedException edenied){
    String msg=edenied.getMessage(); //operazione non permessa (es: comune non disponibile)
    System.out.println(msg);
}
catch (RequestException erequest){
    String msg=erequest.getMessage(); //dati mancanti nella richiesta o richiesta vuota
    System.out.println(msg);
}

catch (INADException einad){
    String msg=einad.getMessage(); //errori, anomalie restituiti da INAD
    System.out.println(msg);
}

// risposta
if (resp != null) {
    String codiceFiscale = resp.getCodiceFiscale();
    String dataEstrazione = resp.getDataEstrazione();
    LocalDateTime dataEstrazioneData = resp.getDataEstrazioneAsDate();
    for (int i = 0; i < resp.getIndirizziDigitali().length; i++) {
        String indirizzoDigitale = resp.getIndirizziDigitali()[i].getIndirizzoDigitale();
        String professione = resp.getIndirizziDigitali()[i].getProfessione();
        String dataFineValidita=resp.getIndirizziDigitali()[i].getDataFineValidita();
        LocalDateTime dataFineValiditaData=resp.getIndirizziDigitali()[i].getDataFineValiditaAsDate();
        String motivazioneCessazione = resp.getIndirizziDigitali()[i].getMotivazioneCessazione();
    }
}
}
```

API .Net (C#)

Si aggiunge al progetto la libreria:

- **SteseiGateway.dll**

Si richiama nel seguente modo:

```
namespace Test {
    class Program {
        static void Main(string[] args) {
            TestST012();
        }
        static void TestST012() {
            Client cl = new Client("http://84.240.190.236", "040007", "INIZIALE");
            RespST012 resp = null;
            try {
                resp = cl.ST012("040007", "TRVVCN73H02L259I", "Pratica N. 987654321");
            }
            catch (ConnectionException econn) {
                string msg = econn.Message; //errore di connessione
            }
            catch (ServerException eserver) {
                string msg = eserver.Message; //errore interno del server
            }
            catch (AuthenticationException eauth) {
                string msg = eauth.Message; //utente e/o password errati
            }
            catch (PermissionDeniedException edenied) {
                string msg = edenied.Message; //operazione non permessa (es: comune non disponibile)
            }
            catch (RequestException erequest) {
                string msg = erequest.Message; //dati mancanti nella richiesta o richiesta vuota
            }
            catch (INADException einad) {
                string msg = einad.Message; //errori, anomalie restituiti da INAD
            }

            //risposta
            if (resp != null) {
                string codiceFiscale = resp.codiceFiscale;
                string dataEstrazione = resp.dataEstrazione;
                DateTime dataEstrazioneData = resp.dataEstrazioneAsDate();
                for (int i = 0; i < resp.indirizziDigitali.Length; i++) {
                    string indirizzoDigitale = resp.indirizziDigitali[i].indirizzoDigitale;
                    string professione = resp.indirizziDigitali[i].professione;
                    string dataFineValidita = resp.indirizziDigitali[i].dataFineValidita;
                    DateTime dataFineValiditaData = resp.indirizziDigitali[i].dataFineValiditaAsDate();
                    string motivazioneCessazione = resp.indirizziDigitali[i].motivazioneCessazione;
                }
            }
        }
    }
}
```

API Javascript

```
let endpoint= "http://84.240.190.236/api/ST012";
var criteri={comuneIstat: "040007", codiceFiscale: "TRVVCN73H02L259I"};
var account={utente: "040007", password: "INIZIALE"};
var restrizioni={ motivoRichiesta:"test pratica"};
var richiesta={credenzialiAccesso: account, restrizioniAccesso: restrizioni, criteriRicerca: criteri};
```

```
$.ajax({
  type: "POST",
  url: endpoint,
  data: JSON.stringify(richiesta),
  contentType: "application/json; charset=utf-8",
  dataType: "json",
  success: function (risposta){
    showJson(risposta);
  },
  error: function (qXHR, textStatus, errorThrown){
    alert("ERR");
  }
});
```

```
let risposta={
  "errore": {
    "errore": false, "tipoErrore": "", "messaggio": ""
  },
  "response": {
    "codiceFiscale": "TRVVCN73H02L259I",
    "dataEstrazione": "2023-12-15T14:38:09Z",
    "indirizziDigitali": [{
      "indirizzoDigitale": "test@pec.it",
      "professione": "NON AVVOCATO",
      "motivazioneCessazione": "",
      "dataFineValidita": ""
    },
    {
      "indirizzoDigitale": "test@pec.it",
      "professione": "MAI AVVOCATO",
      "motivazioneCessazione": "",
      "dataFineValidita": ""
    },
    {
      "indirizzoDigitale": "test@pec.it",
      "professione": "",
      "motivazioneCessazione": "",
      "dataFineValidita": ""
    }
  ]
}
```

Valori tipoErrore

- **"Server Error"**: errori interni del server
- **"Authentication Error"**: errore di autenticazione (utente e/o password errati)
- **"Permission Error"**: operazione non permessa (esempio: Comune non censito nel Gateway o Codice ISTAT del Comune vuoto)
- **"Request Error"**:
 - Se i parametri della ricerca inseriti non sono corretti:
 - "Il parametro Codice Istat ha un formato non valido"
 - "Il parametro Codice Fiscale ha un formato non valido"
 - Se manca il criterio di ricerca
 - "Criteri di ricerca non validi"
- **"INAD Error"**: errori e anomalie restituite da INAD a fronte della ricerca effettuata