Indagor Service
• ST028 •

ACQUISIZIONE PUNTUALE
INDICATORE ISEE
RESIDENTI

Specifiche di Integrazione

INDICE

Descrizione 1
Contenuto della Richiesta 2
Contenuto delle Eccezioni 4
Contenuto della Risposta 5
Esempi di Integrazione 6
API Java 6
API .Net (C#) 8
API Javascript 10
Tabelle 12

ST028
INDICATORE ISEE Residenti

DESCRIZIONE

Il servizio consente di acquisire in INPS - in sincrono - i dati inerenti l'Indicatore ISEE.

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"

Nelle pagine che seguono descriviamo il metodo da utilizzare per l'inoltro della richiesta.
Tuttavia segnaliamo che non è disponibile un ambiente di test.
Nel caso, andremo a predisporre l'ambiente di test specifico per l'ente fruitore del connettore INDAGOR GATEWAY

Metodo:

Richiesta per Codice Fiscale

RespST028 ST028_CodiceFiscale(string comuneIstat, string codiceFiscale, string dataValidita, string prestazioneDaErogare, string protocolloDomandaEnteErogatore, string statoDomandaPrestazione, string tipoIndicatore, string idOperatore, string idUfficio)

- **ST028**: Nome dell'Indagor Service richiamato
- **comuneIstat**: Codice Istat del Comune fruitore dell'applicativo che inoltra la chiamata
- **codiceFiscale**: Codice Fiscale del cittadino
- **dataValidita**: Data di Validità
- **prestazioneDaErogare**: Vedi Tabella "Prestazione da Erogare"
- **protocolloDomandaEnteErogatore**
- **statoDomandaPrestazione**: Vedi Tabella "Stato Domanda Prestazione"
- **tipoIndicatore**: Vedi Tabella "Tipo Indicatore"
- **idOperatore**: Codice ID assegnato dall'applicativo all'operatore del Comune che inoltra la richiesta
- **idUfficio**: Codice Ufficio di appartenenza dell'operatore che inoltra la Richiesta; valore compreso tra 1 e 21 (vedi Tabella "Uffici Comunali")

1	Richiesta con Ricerca per Codice Fiscale	
1.1	Tipo Indicatore	Vedi Tabella "Tipo Indicatore"
1.2	Ricerca per Codice Fiscale	
1.2.1	Codice Fiscale	
1.2.2	Data Validità	aaaa-mm-gg
1.2.3	Prestazione da Erogare	Vedi Tabella "Prestazione da Erogare"
1.2.4	Protocollo Domanda Ente Erogatore	Massimo 256 caratteri
1.2.5	Stato Domanda Prestazione	Vedi Tabella "Stato Domanda Prestazione "

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"
- **"INPSException:"**: errori e anomalie restituite da INPS a fronte della ricerca effettuata

CONTENUTO DELLA RISPOSTA

Di seguito sono riportati i dati che compongono la risposta resa dal servizio ST028; se i valori della risposta sono pari a:

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

significa che tali dati non sono presenti in INPS

1	Tipo Indicatore	
2	ISE	
3	Scala Equivalenza	
4	ISEE	
5	ISR	
6	ISP	
7	Protocollo DSU	Formato regex INPS-ISEE-[0-9]{4}-[0-9]{8}[A-Z]-[0-9]{2} Esempio: INPS-ISEE-2024-01234567A-01
8	Data Presentazione	aaaa-mm-gg

ESEMPI DI INTEGRAZIONE

Tali esempi rivestono una funzione puramente indicativa in quanto non è disponibile l'ambiente di test.

API Java

Si aggiunge al progetto la libreria:

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

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){
		TestST028();
	}
	public static void TestST028(){
		Client cl = new Client("http://84.240.190.236", "040007", "INIZIALE");
		RespST028 resp = null;
		Try{
			String idOperatore = "user00001";
			String idUfficio = "1234";
			resp = cl.ST028_CodiceFiscale("017029", "RSSMRA70A01H501S", "2020-12-31", "A1.01", "ABCD-1234-ZQ2",
			"Da Erogare", "Ordinario", idOperatore, idUfficio);
		}
		catch (ConnectionException econn){
			String msg = econn.getMessage();   //errore di connessione
			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 (INPSException einps){
                String msg = einps.getMessage();  //errori, anomalie restituiti da INPS
            }



            //risposta
            if (resp != null){
                String tipoIndicatore = resp.getTipoIndicatore();
                String ise = resp.getISE();
                double iSENum = resp.getISEAsNumber();
                String scalaEquivalenza = resp.getScalaEquivalenza();
                double scalaEquivalenzaNum = resp.getScalaEquivalenzaAsNumber();
                String isee = resp.getISEE();
                double iSEENum = resp.getISEEAsNumber();
                String isr = resp.getISR();
                double iSRNum = resp.getISRAsNumber();
                String isp = resp.getISP();
                double iSPNum = resp.getISPAsNumber();
                String ProtocolloDSU = resp.getProtocolloDSU();
                String dataPresentazione = resp.getDataPresentazione();
                LocalDate dataPresentazioneData = resp.getDataPresentazioneAsDate();
                String[] codiciOmissioneDifformita = resp.getCodiciOmissioneDifformita();
            }
        }
}
```

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){
            TestST028();
        }
        static void TestST028(){
            Client cl = new Client("http://84.240.190.236", "040007", "INIZIALE");
            RespST028 resp = null;
            try{
                string idOperatore = "user00001";
                string idUfficio = "1234";
                resp = cl.ST028_CodiceFiscale("017029", "RSSMRA70A01H501S", "2020-12-31", "A1.01",
                "ABCD-1234-ZQ2", "Da Erogare", "Ordinario", idOperatore, idUfficio);
            }
            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 (INPSException einps){
                string msg = einps.Message;  //errori, anomalie restituiti da INPS
            }
```

```
                //risposta
                if (resp != null){
                        string tipoIndicatore = resp.TipoIndicatore;
                        string ise = resp.ISE;
                        double iSENum = resp.ISEAsNumber();
                        string scalaEquivalenza = resp.ScalaEquivalenza;
                        double scalaEquivalenzaNum = resp.ScalaEquivalenzaAsNumber();
                        string isee = resp.ISEE;
                        double iSEENum = resp.ISEEAsNumber();
                        string isr = resp.ISR;
                        double iSRNum = resp.ISRAsNumber();
                        string isp = resp.ISP;
                        double iSPNum = resp.ISPAsNumber();
                        string ProtocolloDSU = resp.ProtocolloDSU;
                        string dataPresentazione = resp.DataPresentazione;
                        DateTime dataPresentazioneData = resp.DataPresentazioneAsDate();
                        string[] codiciOmissioneDifformita = resp.CodiciOmissioneDifformita;
                }
        }
    }
}
```

API Javascript

```
let endpoint = "http://84.240.190.236/api/ST028";

var criteri = {
	comuneIstat: "017029", tipoIndicatore: "Ordinario", ricercaCF: {
		codiceFiscale: "RSSMRA70A01H501S",
		dataValidita: "2020-12-31",
		prestazioneDaErogare: "A1.01",
		protocolloDomandaEnteErogatore: "ABCD-1234-ZQ2",
		statodomandaPrestazione: "Da Erogare"
	}
};
var account = { utente: "040007", password: "INIZIALE" };
var restrizioni = { idOperatore: "user00001", idUfficio: "1234" };
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": {
		"TipoIndicatore": "Ordinario",
		"ISE": "1951.20",
		"ScalaEquivalenza": "1.50",
		"ISEE": "1300.80",
		"ISR": "0.00",
		"ISP": "9756.00",
		"ProtocolloDSU": "INPS-ISEE-2024-33445566Z-00",
		"DataPresentazione": "2024-01-11",
		CodiciOmissioneDifformita: []
	}
}
```

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"
    - "Il parametro Data Validità ha un formato non corretto"
    - "Il parametro Prestazione Da Erogare è assente"
    - "Il parametro Prestazione Da Erogare non è un valore valido"
    - "Il parametro Protocollo Domanda Ente Erogatore è assente"
    - "Il parametro Stato Domanda Prestazione è assente"
    - "Il parametro Stato Domanda Prestazione non è un valore ammesso"
    - "Il parametro Tipo Indicatore è assente"
    - "Il parametro Tipo Indicatore non è un valore ammesso
  - "Se manca il criterio di ricerca
    - "Criteri di ricerca non validi"
- **"INPS Error"**: errori e anomalie restituite da INPS a fronte della ricerca effettuata

TABELLE

Tabella TIPO INDICATORE	
1	Ordinario
2	Minorenne
3	Universitario
4	SocioSanitario
5	Dottorato
6	Residenziale

Tabella STATO DOMANDA PRESTAZIONE	
1	Da Erogare
2	Erogata

Tabella Codici Omissione Difformita	
PATRED	Patrimoniali e/o Reddituali
NUCLEO	Discordanze nel Nucleo Familiare

Tabella PRESTAZIONI DA EROGARE	
A1.01	Assegno per il nucleo familiare erogati dai comuni
A1.02	Assegno Maternità erogato dai Comuni
A1.03	Carta acquisti
A1.04	Contributi economici a integrazione del reddito Familiare
A1.05	Contributi economici per alloggio
A1.06	Buoni spesa o buoni pasto
A1.07	Contributi e integrazioni a rette per asili nido
A1.08	Contributi e integrazioni a rette per servizi integrativi o innovativi per la prima infanzia
A1.09	Contributi economici per i servizi scolastici
A1.10	Contributi economici per cure o prestazioni sociali a rilevanza sanitaria
A1.11	Assegnazioni economiche per il sostegno della domiciliarità e dell'autonomia personale
A1.12	Contributi e integrazioni a rette per accesso a centri Diurni
A1.13	Contributi e integrazioni a rette per accesso ai servizi semi-residenziali
A1.14	Contributi e integrazioni a rette per accesso a servizi Residenziali
A1.15	Contributi per servizi alla persona
A1.16	Contributi economici per servizio trasporto e mobilità
A1.17	Contributi economici erogati a titolo di prestito/prestiti d'onore
A1.18	Contributi economici per l'inserimento lavorativo
A1.19	Borse di studio
A1.20	Buono vacanze
A2.01	Mensa sociale
A2.02	Sostegno Socio-Educativo territoriale o domiciliare
A2.03	Prestazioni del diritto allo studio universitario
A2.04	Agevolazioni per tasse universitarie
A2.05	Agevolazioni per i servizi di pubblica utilità (telefono, luce, gas)
A2.06	Agevolazioni tributarie comunali (nettezza urbana, ecc.)
A2.07	Assistenza domiciliare Socio-Assistenziale
A2.08	A.D.I.- Assistenza domiciliare integrata con servizi Sanitari
A2.09	Supporto all'inserimento lavorativo
A2.10	Servizi integrativi per la prima infanzia
A2.11	Sostegno Socio-Educativo scolastico
A2.12	Mensa scolastica
A3.01	Strutture semiresidenziali
A3.02	Strutture residenziali
A3.03	Asilo Nido
Z9.99	Altro

Tabella UFFICI COMUNALI	
1	Scuola e Pubblica Istruzione
2	Sociali/Socio-Sanitari – Welfare
3	Biblioteche
4	Governo del Territorio – Sportelli Edilizia
5	Tributi
6	Cimiteri
7	Ambiente
8	Segreteria Generale – Organi Istituzionali - Circoscrizioni, Decentramento, Quartieri, Istituti di partecipazione
9	Avvocatura e Uffici competenti per Ordinanze – Ingiunzioni
10	Notifiche e Spedizioni
11	Personale – Risorse Umane
12	Patrimonio mobiliare ed immobiliare
13	Espropri
14	Igiene e Sanità Pubblica – Animali d'affezione - Inquinamento
16	Stazione Unica Appaltante
17	Manutenzioni Lavori Pubbliche, Strade, Infrastrutture, Reperibilità
18	Ragioneria e Servizi Finanziari
19	Economato
20	Polizia Locale – Attività di Polizia Giudiziaria e Servizi della Centrale Operativa
21	Polizia Locale – Attività Amministrative