



Indagor Service

- ST025 •

**ACQUISIZIONE PUNTUALE
CONSULTAZIONE REGISTRO IMPRESE
• RICERCA IMPRESA •**

Specifiche di Integrazione

INDICE

Modifiche al documento.....	1
Descrizione.....	2
Contenuto della Richiesta.....	3
Contenuto delle Eccezioni.....	7
Contenuto della Risposta.....	8
Esempi di Integrazione.....	9
API Java.....	9
API .Net (C#).....	11
API Javascript.....	13

MODIFICHE AL DOCUMENTO

Data Modifica	Versione modificata	Descrizione Modifica
29/05/2025	1.0	Nel “Contenuto della Richiesta” (Pag. 3) tolta barra finale in https://gateway1.steseiservizi.it/

ST025

Consultazione REGISTRO IMPRESE

RICERCA IMPRESA

DESCRIZIONE

Il servizio consente di acquisire in UNIONCAMERE - in sincrono - i dati inerenti al REGISTRO IMPRESE – RICERCA IMPRESA

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"**

Metodi:*Richiesta per Numero REA***RespST025 ST025_NumeroRea(string comunelstat, string numeroRea, string provincia)**

- **ST025:** Nome dell'Indagor Service richiamato
- **comunelstat:** Codice Istat del Comune fruitore dell'applicativo che inoltra la chiamata
 - Ambiente di test: Codice Istat **040007**
- **numeroRea:** Numero REA
 - Ambiente di test: Numero REA (Esempio **350128**)
- **provincia:** Sigla Provincia della Camera (Esempio **VI**)

1 Richiesta con Ricerca per Numero REA

1.1	Numero REA	Massimo 9 caratteri
1.2	Sigla Provincia	Obbligatorio (2 lettere)

*Richiesta per Codice Fiscale***RespST025 ST025_CodiceFiscale(string comunelstat, string codiceFiscale)**

- **ST025:** 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 della Ditta/Azienda/Società
 - Ambiente di test: Codice Fiscale (Esempio **03742480241**)

1 Richiesta con Ricerca per Codice Fiscale

1.1	Codice Fiscale	
-----	----------------	--

Richiesta per Ragione Sociale / Denominazione Impresa**RespST025 ST025_Denominazione(string comunelstat, string denominazione, string provincia)**

- **ST025:** Nome dell'Indagor Service richiamato
- **comunelstat:** Codice Istat del Comune fruitore dell'applicativo che inoltra la chiamata
 - Ambiente di test: Codice Istat **040007**
- **denominazione:** Ragione Sociale o Denominazione dell'Impresa (almeno 2 caratteri)
 - Ambiente di test: denominazione (Esempio **ASTER**)
- **provincia:** Sigla Provincia della Camera (Esempio **VI**)

1 Richiesta con Ricerca per Ragione Sociale/Denominazione Impresa

1.1	Ragione Sociale/Denominazione Impresa	Da 2 a 50 caratteri obbligatori
1.2	Sigla Provincia	Facoltativo (2 lettere)

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"
 - "Numero REA deve essere un numero da 1 a 9 cifre"
 - "Provincia non è valida"
 - "Denominazione deve avere lunghezza da 2 a 50 caratteri"
 - Se manca il criterio di ricerca
 - "Criteri di ricerca non validi"
- **"UCEException"**: errori e anomalie restituite da UnionCamere a fronte della ricerca effettuata

CONTENUTO DELLA RISPOSTA

Di seguito sono riportati i dati che compongono la risposta resa dal servizio ST025; se i valori della risposta sono pari a:

- null
- stringa vuota
- LocalDate.MIN

significa che tali dati non sono presenti in UnionCamere

1	CCIAA/Numero REA	CCIAA
		Numero REA
		Numero RD (Registro Ditte)
2	Denominazione	Denominazione
3	Forma Giuridica	Codice Natura Giuridica
		Descrizione Natura Giuridica
4	Codice Fiscale	Codice Fiscale
5	Stato Impresa	Stato Impresa
		Stato Ditta
6	Indirizzo, Sede Legale e PEC	Comune
		Sigla Provincia
		Toponimo (Via, Corso, Piazza, etc...)
		Via
		Numero Civico
		CAP
		PEC

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){
        TestST025();
    }
    public static void TestST025(){
        Client cl = new Client("http://84.240.190.236", "040007", "INIZIALE");
        RespST025 resp = null;
        Try{ /* scegliere uno dei tre criteri */
            resp = cl.ST025_CodiceFiscale("040007", "00149670242");
            resp = cl.ST025_NumeroRea("040007", "152100", "RO");
            resp = cl.ST025_Denominazione("040007", "IMP", "VI");
        }
        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 (UCEException euc){
    String msg = euc.getMessage(); //errori, anomalie restituiti da UnionCamere
}

//risposta
if (resp != null)
{
    for (int i=0; i<resp.getImprese().length; i++)
    {
        String cciaa = resp.getImprese()[i].getCciaa();
        String numeroREA = resp.getImprese()[i].getNumeroREA();
        String numeroRD = resp.getImprese()[i].getNumeroRD();
        String denominazione = resp.getImprese()[i].getDenominazione();
        String naturaGiuridicaCodice = resp.getImprese()[i].getNaturaGiuridicaCodice();
        String naturaGiuridicaDescrizione = resp.getImprese()[i].getNaturaGiuridicaDescrizione();
        String codiceFiscale = resp.getImprese()[i].getCodiceFiscale();
        String statolImpresa = resp.getImprese()[i].getStatolImpresa();
        String statoDitta = resp.getImprese()[i].getStatoDitta();
        String comune = resp.getImprese()[i].getIndirizzoSedeLegaleComune();
        String provincia = resp.getImprese()[i].getIndirizzoSedeLegaleProvincia();
        String toponimo = resp.getImprese()[i].getIndirizzoSedeLegaleToponimo();
        String via = resp.getImprese()[i].getIndirizzoSedeLegaleVia();
        String numeroCivico = resp.getImprese()[i].getIndirizzoSedeLegaleNumeroCivico();
        String cap = resp.getImprese()[i].getIndirizzoSedeLegaleCap();
        String pec = resp.getImprese()[i].getPEC();
        String msgErrore= resp.getImprese()[i].getMessaggioErrore();
    }
}
}
```

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){
            TestST025();
        }
        static void TestST025(){
            Client cl = new Client("http://84.240.190.236", "040007", "INIZIALE");
            RespST025 resp = null;
            try{ /* scegliere uno dei tre criteri */
                resp = cl.ST025_CodiceFiscale("040007", "00149670242");
                resp = cl.ST025_NumeroRea("040007", "152100", "RO");
                resp = cl.ST025_Denominazione("040007", "IMP", "VI");
            }
            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 (UCEException euc){
                string msg = euc.Message; //errori, anomalie restituiti da UnionCamere
            }
        }
    }
}
```

```
//risposta
if (resp != null)
{
    for (int i=0; i<resp.imprese.Length; i++)
    {
        string cciaa = resp.imprese[i].Cciao();
        string numeroREA = resp.imprese[i].NumeroREA();
        string numeroRD = resp.imprese[i].NumeroRD();
        string denominazione = resp.imprese[i].Denominazione();
        string naturaGiuridicaCodice = resp.imprese[i].NaturaGiuridicaCodice();
        string naturaGiuridicaDescrizione = resp.imprese[i].NaturaGiuridicaDescrizione();
        string codiceFiscale = resp.imprese[i].CodiceFiscale();
        string statolImpresa = resp.imprese[i].StatolImpresa();
        string statoDitta = resp.imprese[i].StatoDitta();
        string comune = resp.imprese[i].IndirizzoSedeLegaleComune();
        string provincia = resp.imprese[i].IndirizzoSedeLegaleProvincia();
        string toponimo = resp.imprese[i].IndirizzoSedeLegaleToponimo();
        string via = resp.imprese[i].IndirizzoSedeLegaleVia();
        string numeroCivico = resp.imprese[i].IndirizzoSedeLegaleNumeroCivico();
        string cap = resp.imprese[i].IndirizzoSedeLegaleCap();
        string pec = resp.imprese[i].PEC();
        string msgErrore= resp.imprese[i].MessaggioErrore;
    }
}
}
```

API Javascript

```
let endpoint = "http://84.240.190.236/api/ST025";

var criteri = {      /* scegliere uno dei tre criteri */
    comunelstat: "040007", codiceFiscale: "00149670242"
    comunelstat: "040007", numeroRea: "152100", provincia: "RO"
    comunelstat: "040007", denominazione: "IMP", provincia: "VI"
};

var account = { utente: "040007", password: "INIZIALE" };
var richiesta = { credenzialiAccesso: account, 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": {
    "imprese": [
      {
        "Cciaa": "VI",
        "NumeroREA": "350128",
        "NumeroRD": "",
        "Denominazione": "IMP-INDUSTRIE MECCANICHE PEMA SRL",
        "NaturaGiuridicaCodice": "SR",
        "NaturaGiuridicaDescrizione": "SOCIETA' A RESPONSABILITA' LIMITATA",
        "CodiceFiscale": "03742480241",
        "StatoImpresa": "Registrata",
        "StatoDitta": "",
        "IndirizzoSedeLegaleComune": "DUEVILLE",
        "IndirizzoSedeLegaleProvincia": "VI",
        "IndirizzoSedeLegaleToponimo": "VIA",
        "IndirizzoSedeLegaleVia": "G. ROSSINI",
        "IndirizzoSedeLegaleNumeroCivico": "24",
        "IndirizzoSedeLegaleCap": "36031",
        "PEC": "PEMASRL@LEGALMAIL.IT"
      },
      {
        "Cciaa": "VI",
        "NumeroREA": "157181",
        "NumeroRD": "",
        "Denominazione": "I.M.P. IMBALLAGGI MATERIE PLASTICHE - S.P.A.",
        "NaturaGiuridicaCodice": "SP",
        "NaturaGiuridicaDescrizione": "SOCIETA' PER AZIONI",
        "CodiceFiscale": "01273630242",
        "StatoImpresa": "Cessata/Cancellata",
        "StatoDitta": "",
        "IndirizzoSedeLegaleComune": "ALTAVILLA VICENTINA",
        "IndirizzoSedeLegaleProvincia": "VI",
        "IndirizzoSedeLegaleToponimo": "VIA",
        "IndirizzoSedeLegaleVia": "QUATTRO NOVEMBRE",
        "IndirizzoSedeLegaleNumeroCivico": "8",
        "IndirizzoSedeLegaleCap": "36077",
        "PEC": "",
        "MessaggioErrore": ""
      }
    ]
  }
}
```


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"
 - "Provincia non è valida"
 - "Numero REA deve essere un numero da 1 a 9 cifre"
 - "Denominazione deve avere lunghezza da 2 a 50 caratteri"
 - "Se manca il criterio di ricerca"
 - "Criteri di ricerca non validi"
- **"UC Error"**: errori e anomalie restituite da UnionCamere a fronte della ricerca effettuata