

---

# **Guida all'implementazione del SDK Mobile CIE id**

*Release version: latest*

**italia**

**01 set 2023**



<b>1</b>	<b>Flussi di autenticazione</b>	<b>3</b>
1.1	Flusso con reindirizzamento . . . . .	3
1.2	Flusso integrato . . . . .	6
<b>2</b>	<b>SDK Android</b>	<b>9</b>
2.1	Requisiti di integrazione . . . . .	9
2.2	Configurazione . . . . .	9
2.3	Modalità di integrazione . . . . .	10
<b>3</b>	<b>SDK iOS</b>	<b>13</b>
3.1	Requisiti tecnici . . . . .	13
3.2	Requisiti di integrazione . . . . .	13
3.3	Come si usa . . . . .	13
3.4	Flusso con reindirizzamento . . . . .	14
3.5	Flusso interno . . . . .	14
3.6	Importazione . . . . .	14
3.7	Configurazione URL Scheme . . . . .	14
3.8	Configurazione Service Provider URL . . . . .	14
3.9	Importazione del pulsante Entra con CIE . . . . .	14
3.10	Eeguire l'autenticazione . . . . .	15
3.11	Gestione eventi . . . . .	15
<b>4</b>	<b>Licenza</b>	<b>17</b>



I Service Provider che offrono parte o tutti i loro servizi tramite app hanno la possibilità di integrare lo schema di autenticazione «Entra con CIE» mediante l'SDK CieID descritta nel presente documento.



---

## Flussi di autenticazione

---

In questa Sezione sono descritti i principali flussi di autenticazione che è possibile integrare all'interno delle Applicazioni Mobile dei Fornitori di Servizi.

### **1.1 Flusso con reindirizzamento**

L'App del Service Provider, all'atto della richiesta di autenticazione dell'utente, reindirizza la richiesta all'App CieID che cura i dettagli la comunicazione e l'autenticazione con la CIE, mascherando tali aspetti all'applicazione chiamante.

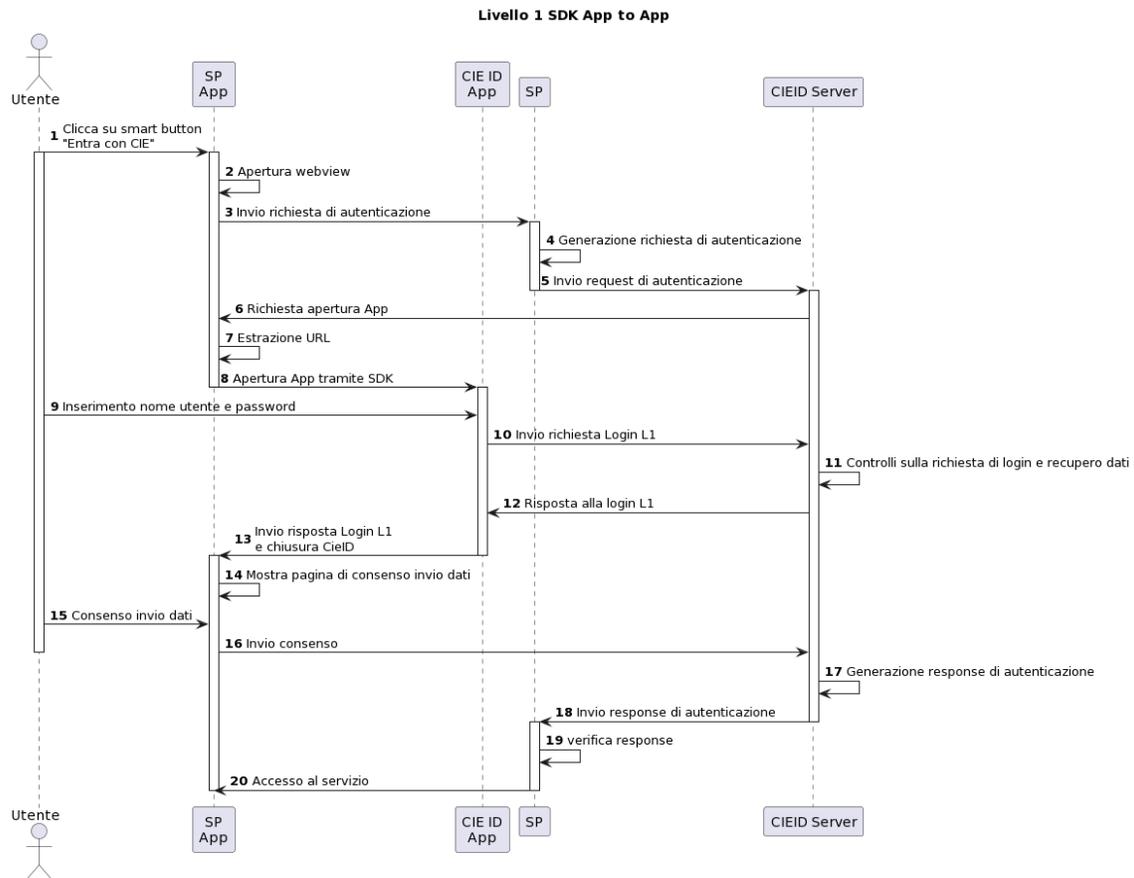


Fig. 1.1: Autenticazione di livello 1 - Flusso con reindirizzamento

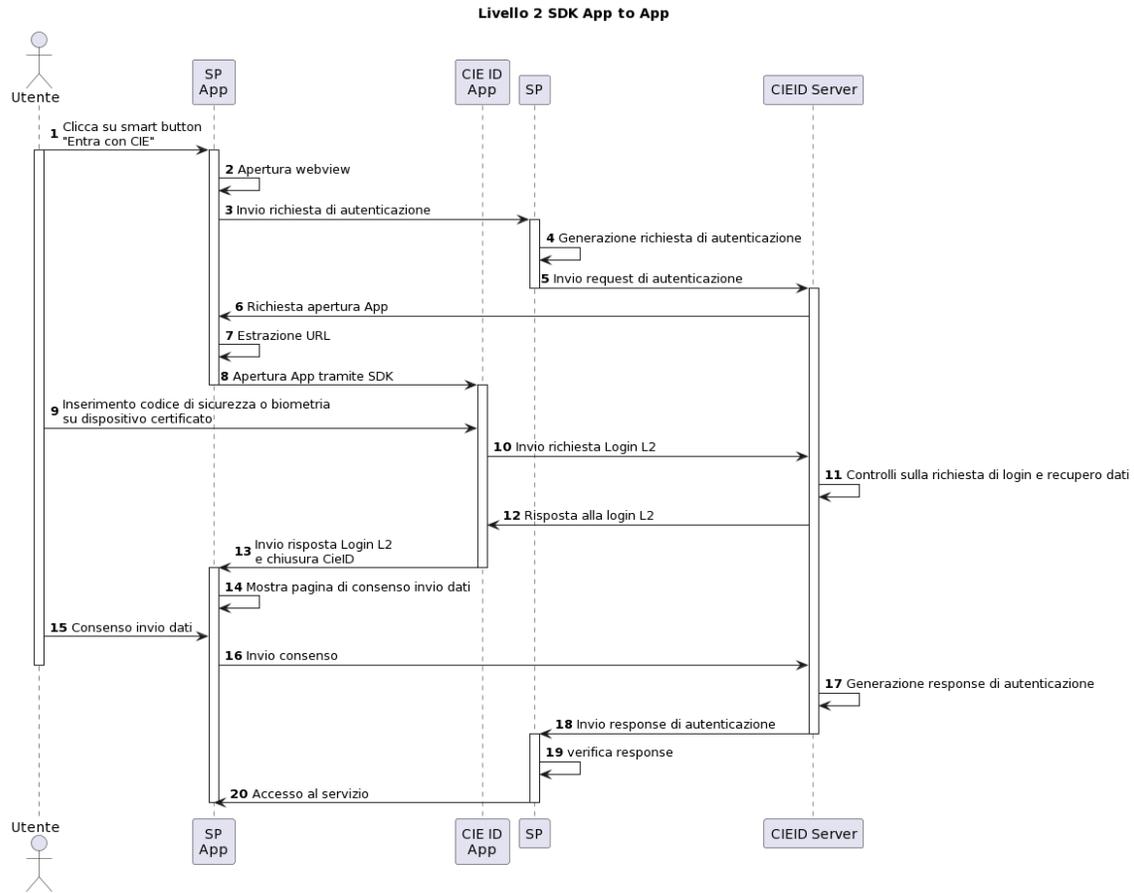


Fig. 1.2: Autenticazione di livello 2 - Flusso con reindirizzamento

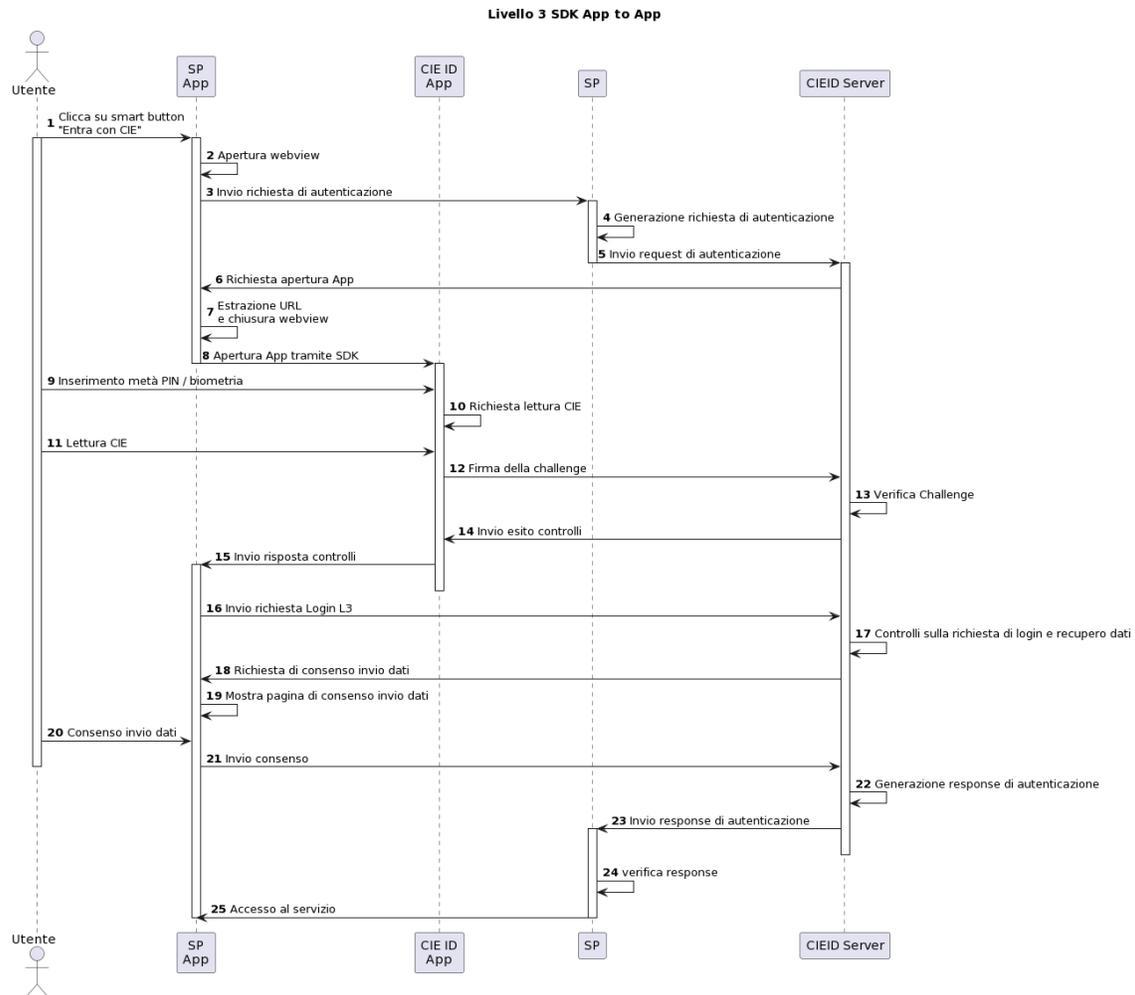


Fig. 1.3: Autenticazione di livello 3 - Flusso con reindirizzamento

## 1.2 Flusso integrato

Il processo di autenticazione viene effettuato direttamente all'interno dell'app del Service Provider, il quale integra la comunicazione con la CIE mediante una libreria SDK, disponibile sia per sistemi operativi Android che iOS.

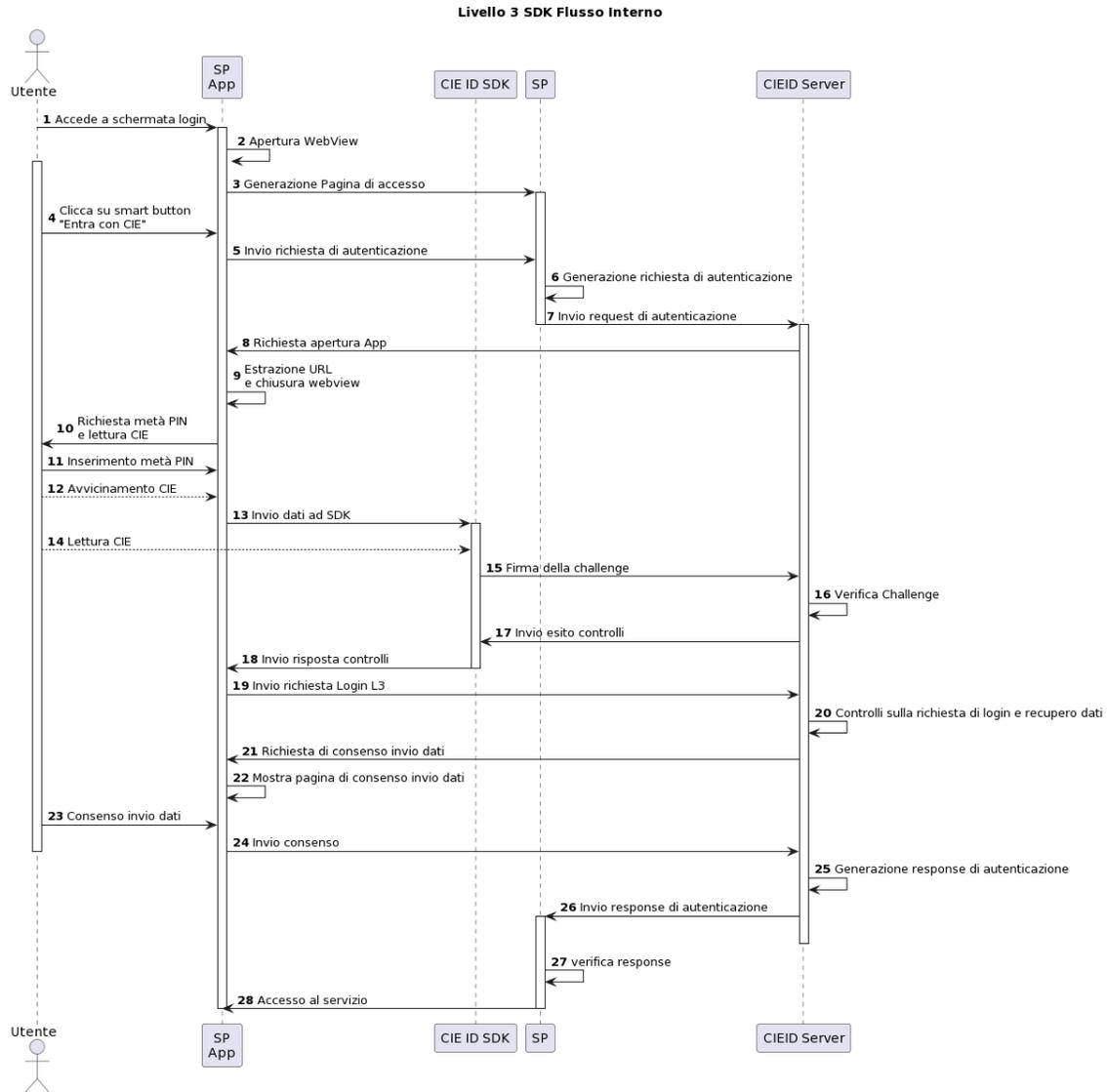


Fig. 1.4: Autenticazione di livello 3 - Flusso interno



La versione del SDK per Android, **CieID-android-sdk**, è disponibile a [questo link](#)<sup>1</sup>. É costituita da una libreria software, realizzata in codice nativo Android Kotlin, e integra una app di esempio che descrive le diverse modalità di integrazione dello schema «Entra con CIE»:

## 2.1 Requisiti di integrazione

L'utilizzo dell'SDK presuppone che il Service Provider sia correttamente federato con l'Identity Provider e che abbia implementato almeno uno tra i protocolli previsti dallo schema di autenticazione «Entra con CIE».

Inoltre, è necessario che i seguenti requisiti siano soddisfatti:

- versione Android 6.0 (API level 23) o successive;
- utilizzo di un dispositivo mobile dotato di tecnologia NFC;
- disponibilità di una connessione internet

## 2.2 Configurazione

Dell'IdP sono messi a disposizione degli erogatori di servizi due ambienti: uno di **preproduzione**, per gli sviluppi applicativi e i test di federazione e l'altro di **produzione**, per il deploy in esercizio. Difatti, per l'impiego di un'app terza con uno dei flussi disponibili è necessaria una fase iniziale di configurazione, che dipende dal tipo di flusso adottato.

Entrambi i flussi vengono avviati tramite l'utilizzo di una *WebView*: é necessario caricare la URL del Service Provider che integra il pulsante «Entra con CIE» come mostrato nell'esempio:

```
*/inserire url service provider*  
webView.loadUrl("URL del Service Provider")
```

---

<sup>1</sup> <https://github.com/italia/cieid-android-sdk>

### Flusso con reindirizzamento

Nel caso di *flusso con reindirizzamento*, per far proseguire correttamente il flusso, è necessario selezionare l'applicazione «CieID» a cui indirizzare le richieste di autenticazione. Ciò può essere fatto modificando i commenti dalle righe di interesse, come mostrato di seguito.

```
val appPackageName = "it.ipzs.cieid"  
*//COLLAUDO*  
*//val appPackageName = "it.ipzs.cieid.collaudo"*
```

### Flusso integrato

Per quanto riguarda il *flusso integrato*, invece, la fase di autenticazione viene gestita dalla libreria software. In questo caso è necessario integrare il modulo «CieIDSdk».

L'SDK utilizza *Gradle* con strumento di build automatico. Per configurare correttamente il flusso, è necessario selezionare l'ambiente server dell'Identity Provider a cui indirizzare le richieste di autenticazione. Ciò può essere fatto modificando il file *build.gradle* modificando i commenti dalle righe di interesse, come mostrato di seguito:

## 2.3 Modalità di integrazione

L'SDK fornisce una app di esempio, con 2 activity, una per flusso, per facilitare al Service Provider l'integrazione all'interno della propria App. La gestione degli errori è demandata all'app integrante.

### Integrazione del flusso con reindirizzamento

Per integrare nativamente le funzionalità dell'SDK è necessario, per prima cosa, intercettare la URL contenente il valore «/OpenApp» ed avviare l'App CieID integrando il codice seguente:

```
val intent = Intent()  
**try** {  
  
    intent.setClassName(appPackageName, className)  
    *//mettere la url caricata dalla webview su /OpenApp*  
    intent.data = Uri.parse(url)  
    intent.action = Intent.ACTION_VIEW  
    startActivityForResult(intent, 0)  
} **catch** (a : ActivityNotFoundException) {  
    startActivity(  
  
        Intent(  
  
            Intent.ACTION_VIEW,  
  
            Uri.parse("https://play.google.com/store/apps/details?id=  
↪$appPackageName")  
  
        )  
  
    )  
  
    )  
  
)  
  
**return true**
```

Una volta avviata correttamente l'App CieID, avviene l'autenticazione tramite la CIE, e al termine viene restituita una nuova URL da ricarica nella WebView precedente, come mostrato nell'esempio seguente:

```
override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {  
  
    **super**.onActivityResult(requestCode, resultCode, data)  
    val url = data?.getStringExtra(URL)  
  
    webView.loadUrl(url)  
  
}
```

### Integrazione del flusso integrato

Per integrare le funzionalità dell'SDK si utilizzano i seguenti metodi:

```
*/Configurazione iniziale*  
CieIDSdk.start(activity, callback)  
  
*/Avvio utilizzo NFC*  
CieIDSdk.startNFCListening(activity)  
  
*/Abilitare o disabilitare i log, da disattivare in produzione*  
CieIDSdk.enableLog = **true**  
  
*/Bisogna settare la url caricata dalla pagina web dell' SP dalla  
webview su /OpenApp*  
CieIDSdk.setUrl(url.toString())  
  
*/inserire il pin della CIE*  
CieIDSdk.pin = input.text.toString()  
  
*/Avviare NFC*  
startNFC()
```

É necessario, inoltre, realizzare le interfacce di Callback implementando i seguenti metodi:

```
override fun onEvent(event: Event) {  
  
*/evento*  
}  
  
override fun onError(e: Throwable) {  
  
*/caso di errore*  
}  
  
override fun onSuccess(url: String) {  
  
*/caso di successo con url della pagina da caricare*  
}
```



CieID-iOS-sdk è un SDK per smartphone iOS sviluppato in linguaggio "Swift" che include le funzionalità di autenticazione di

«Entra con CIE». Utilizzando questo kit, gli sviluppatori di applicazioni terze iOS possono integrare nella propria app l'autenticazione mediante la Carta d'identità elettronica.

### 3.1 Requisiti tecnici

CieID-iOS-sdk richiede versione iOS 13.0 o successive, inoltre è necessario uno smartphone iOS con tecnologia NFC (iPhone 7 o successivo, non è compatibile con iPhone SE di prima generazione - mod 2016).

### 3.2 Requisiti di integrazione

L'utilizzo dell'SDK presuppone che il Service Provider sia correttamente federato con l'Identity Provider e che abbia implementato almeno uno tra i protocolli previsti dallo schema di autenticazione «Entra con CIE».

### 3.3 Come si usa

Il kit prevede il solo flusso di autenticazione con reindirizzamento di seguito descritto. L'integrazione richiede pochi semplici passaggi:

- Importazione del kit all'interno del progetto
- Configurazione dell'URL Scheme
- Configurazione dell'URL di un Service Provider valido all'interno del file Info.plist
- Configurazione dello smart button Entra con CIE all'interno dello storyboard
- Inizializzazione e presentazione della webView di autenticazione

- Gestione dei delegati

## 3.4 Flusso con reindirizzamento

Il flusso di autenticazione con reindirizzamento permette ad un Service Provider accreditato di integrare l'autenticazione Entra con CIE nella propria app iOS, demandando le operazioni di autenticazione all'app CieID. Questo flusso di autenticazione richiede che l'utente abbia l'app CieID installata sul proprio smartphone in **versione 1.2.1 o successiva**.

## 3.5 Flusso interno

Non disponibile.

## 3.6 Importazione

Trascinare il folder CieIDsdk all'interno del progetto xCode

## 3.7 Configurazione URL Scheme

Nel flusso di autenticazione con reindirizzamento l'applicazione CieID avrà bisogno aprire l'app chiamante per poterli notificare l'avvenuta autenticazione. A tal fine è necessario configurare un URL Scheme nel progetto Xcode come segue:

Selezionare il progetto **Target**, aprire il pannello **Info** ed aprire poi il pannello **URL Types**. Compilare i campi **Identifier** e **URL Scheme** inserendo il **Bundle Identifier** dell'app, impostare poi su **none** il campo **Role**.

Il parametro appena inserito nel campo **URL Scheme** dovrà essere riportato nel file **Info.plist**, aggiungendo un parametro chiamato **SP\_URL\_SCHEME** di tipo **String**, come mostrato nell'esempio:

A seguito dell'apertura dell'app la webView dovrà ricevere un nuovo URL e proseguire la navigazione. Di seguito si riporta il metodo **openUrlContext** da importare nello **SceneDelegate** che implementa tale logica:

## 3.8 Configurazione Service Provider URL

Entrambi i flussi vengono avviati tramite l'utilizzo di una WebView, per questo motivo è necessario caricare la URL dell'ambiente di produzione della pagina web del Service Provider che integra il pulsante «Entra con CIE» all'interno del file **Info.plist**, aggiungendo un parametro chiamato **SP\_URL** di tipo **String**, come mostrato nell'esempio:

## 3.9 Importazione del pulsante Entra con CIE

Aggiungere nello storyboard di progetto un oggetto di tipo **UIButton** ed inserire nella voce **Class** del menù **Identity inspector** la classe che lo gestisce: **CieIDButton**. L'oggetto grafico verrà automaticamente renderizzato con il pulsante ufficiale "Entra con CIE".

## 3.10 Eseguire l'autenticazione

Di seguito un esempio di gestione dell'evento **TouchUpInside** per eseguire il codice necessario per inizializzare e presentare la WebView di autenticazione.

La classe chiamante dovrà essere conforme al protocollo **CieIdDelegate** come mostrato nell'esempio.

```
**class ExampleViewController**: UIViewController, CieIdDelegate {
...
}
```

L'utente potrà navigare nella webView mostrata che lo indirizzerà sull'app CieID dove potrà eseguire l'autenticazione con la Carta di Identità Elettronica, al termine verrà nuovamente reindirizzato sull'app chiamante in cui potrà dare il consenso alla condivisione delle informazioni personali e portare al termine l'autenticazione.

Al termine dell'autenticazione verrà chiamato il delegato **CieIDAuthenticationClosedWithSuccess**. La chiamata di questo delegato avviene nella classe **CieIDWKWebViewController**. Potrebbe rendersi necessario posticipare la chiamata di questo delegato in base alla logica di autenticazione del Service Provider.

## 3.11 Gestione eventi

Il protocollo impone la gestione dei seguenti eventi mediante delegati

```
**func** CieIDAuthenticationClosedWithSuccess() {
    print("Authentication closed with SUCCESS")
}
```

```
**func** CieIDAuthenticationCanceled() {
    print("L'utente ha annullato l'operazione")
}
```

```
**func** CieIDAuthenticationClosedWithError(errorMessage: String) {
    print("ERROR MESSAGE: *\>(*\ errorMessage\ *)")
}
```



## CAPITOLO 4

---

### Licenza

---

Il codice sorgente è rilasciato sotto licenza BSD (codice SPDX: BSD-3-Clause).