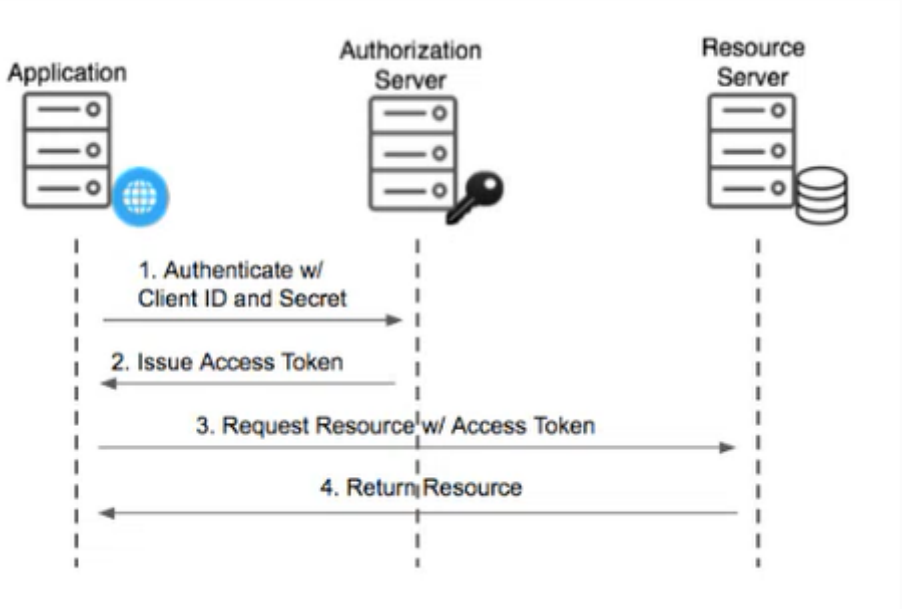
# 15.10 Applicazioni esterne

Questa funzione di vtenext, che si trova in **Impostazioni > Applicazioni esterne**, permette a servizi esterni di accedere ai dati del crm usando OAuth2 come protocollo di autorizzazione.


*Schermata di configurazione per le Applicazioni esterne*

Quindi come funziona esattamente? I servizi esterni, che in questo ambito chiamiamo *applicazioni esterne*, per collegarsi a vtenext tramite REST API, hanno prima bisogno di ottenere un **access_token**, che viene fornito tramite uno dei flussi messi a disposizione dal protocollo OAuth2. Vediamo una immagine esplicativa:




Questo flusso raffigurato, è denominato **Client Credentials**. Ci sono tre parti coinvolte:

1. Applicazione (per esempio un Gestionale)
2. Il Server di Autorizzazione (nel caso di vtenext, è lo stesso del punto 3)

3. Il Server che contiene di Dati (nel nostro caso vtenext)

In pratica, l'Applicazione (che potrebbe essere un Gestionale che deve leggere le Fatture su vtenext), chiede al Server di Autorizzazione un Access Token inviando la coppia di accesso (Client ID e Secret Key) per autenticarsi. Se le credenziali (Client ID e Secret Key) sono corrette, viene rilasciato un Access Token (stringa casuale) con il quale l'Applicazione può chiedere a vtenext le informazioni necessarie (es: Fatture) via REST API.

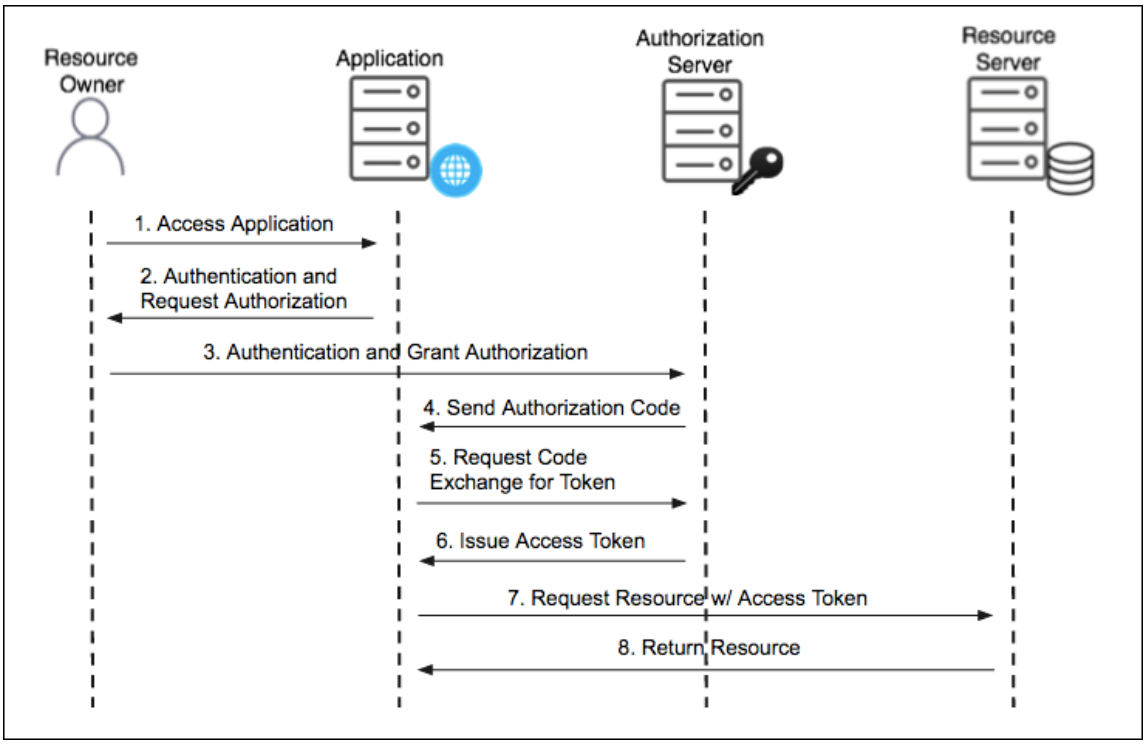
Un esempio di chiamata REST API passando l'access token, è il seguente:

```
curl -X POST '<VTE_URL>/restapi/v1/vtews/query' \
  -H 'Content-Type: application/json' \
  -H 'Authorization: Bearer <ACCESS_TOKEN>' \
  -d '{"query":"SELECT * FROM Invoice;"}'
```

Il vantaggio è che l'access Token ha una durata limitata, è revocabile lato server e quindi è possibile consentire l'accesso alle app in maniera controllata e, soprattutto, non vengono divulgate o condivise password di sistema. Naturalmente allo scadere del Token, sarà necessario ottenerne uno di nuovo ripetendo la procedura sopra menzionata.

Questo tipo di flusso è indicato per la comunicazione diretta tra 2 server, senza bisogno di intervento umano dopo aver configurato entrambe le parti.

Il secondo tipo di flusso, è denominato **Authorization Code**. Questo flusso coinvolge un elemento in più, che identificheremo come Utente.



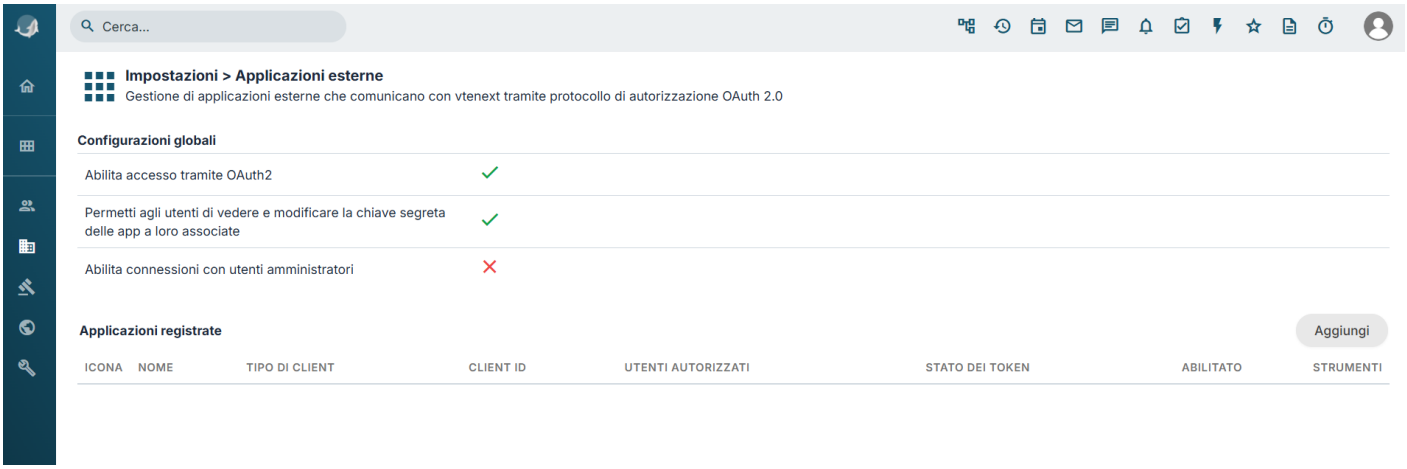
Nell'esempio, diremo che l'Applicazione che si vuole connettere a vtenext questa volta, è Facebook che necessita di leggere i Lead presenti in vtenext. Chi richiede i dati non è però Facebook direttamente, ma l'Utente che lo usa, che deve fornire a Facebook i Leads che l'Utente possiede all'interno del crm. L'obiettivo è quindi che l'utente dia il permesso a Facebook di leggere i Lead su vtenext, quindi sarà fatta un'autenticazione da parte dell'utente su vte, il quale risponderà a Facebook con un Authorization Code. Facebook userà quell'Auth Code per richiedere all'Authorization Server l'Access Token (che può essere temporaneo, da usare una tantum, oppure può essere rinnovato in autonomia) e leggere finalmente i dati nel Server.

Quindi, nonostante Facebook abbia avuto accesso ai Lead di vtenext, non ha memorizzato le credenziali dell'utente, e dopo circa un'ora (durata standard dell'Access Token) scadranno e quindi l'accesso ai dati sensibili sarà bloccato fino a nuova autorizzazione.

Questo flusso è indicato quando una applicazione esterna deve accedere ai dati di un utente per successive elaborazioni.

---

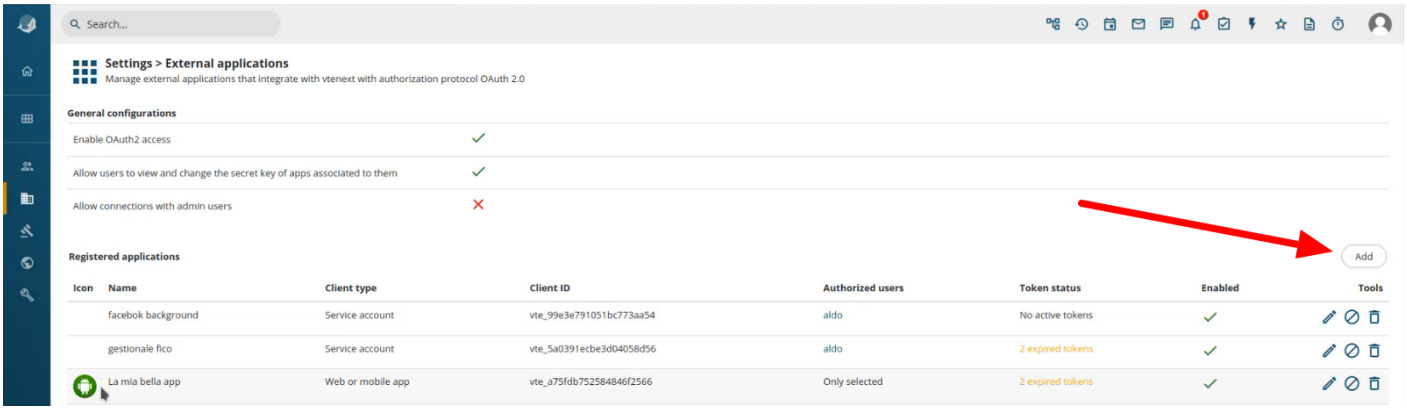
### Configurazione Generale



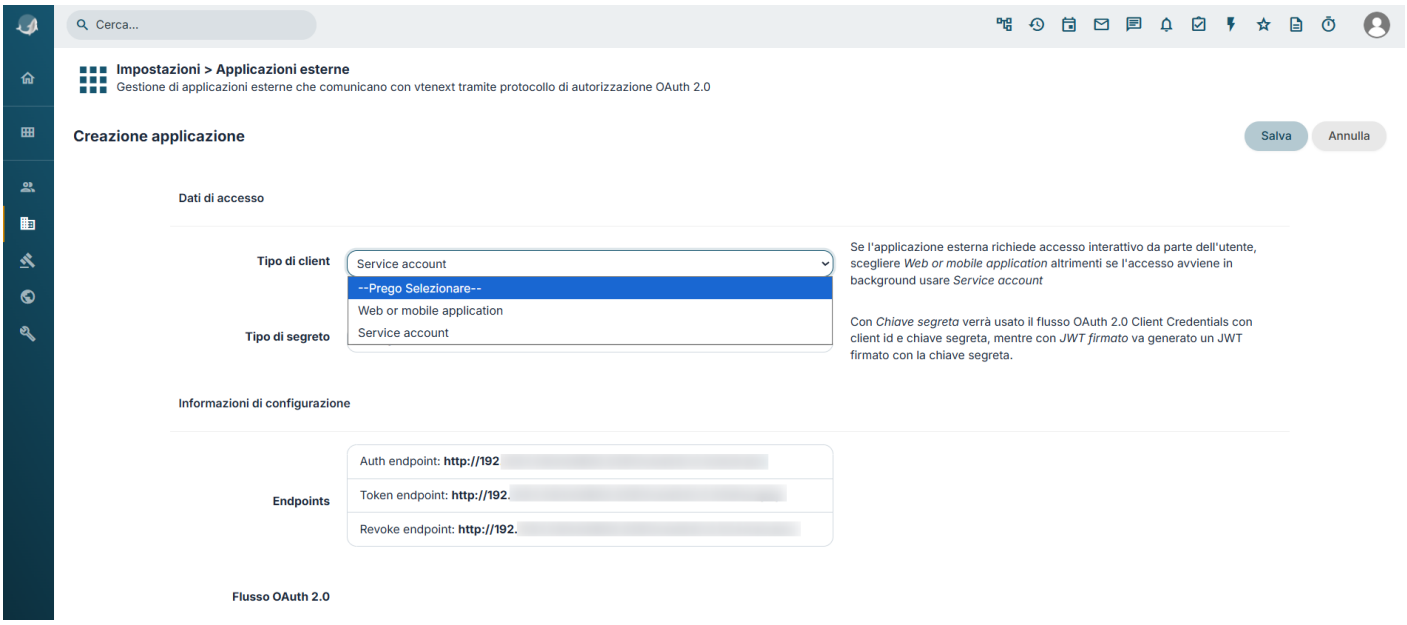

In testa alla pagina sono presenti alcune opzioni generali:

- Abilita accesso tramite OAuth2: attiva o disattiva globalmente l'utilizzo di OAuth2 come protocollo di autorizzazione. Se disattivato, nessuna app esterna può autenticarsi tramite protocollo OAuth2
- Permetti agli utenti di vedere e modificare la chiave segreta delle app a loro associate: se attiva, gli utenti non amministratori possono vedere e modificare la chiave segreta delle applicazioni a loro associate dal loro profilo utente
- Abilita connessioni con utenti amministratori: se attiva, le app esterne possono collegarsi tramite un account di tipo amministratore.

### Configurazione Service Account



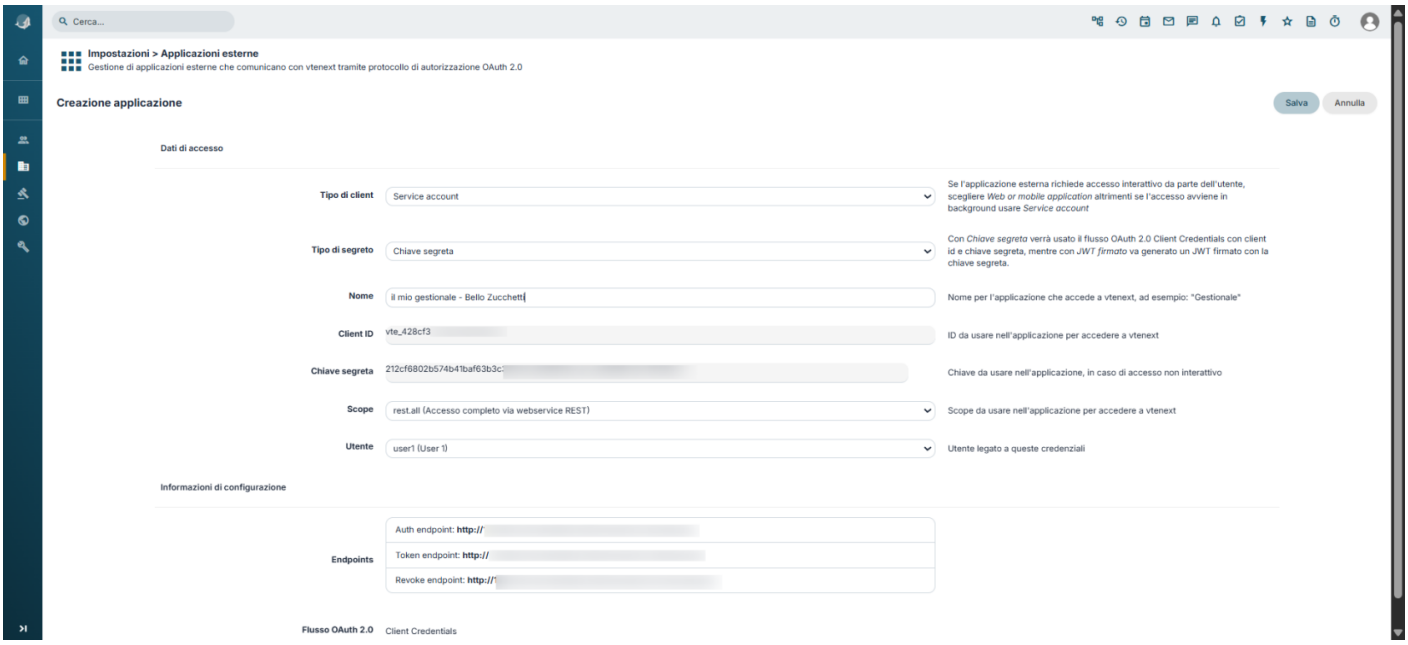

*Cliccare sul pulsante AGGIUNGI sulla destra*




Si accede ad una schermata con un Wizard che ci aiuta a determinare il flusso che vogliamo configurare. Come prima cosa impostare il Tipo di Client e si potrà scegliere tra:

- **Web or Mobile application** se abbiamo bisogno di un accesso tramite login con utente (flusso **Authorization Code**)
- **Service account** se l'accesso avviene in background tra due server (flusso **Client Credentials**)

Volendo dunque configurare un flusso Client Credentials, come **Tipo di Client** scegliamo **Service Account** e come **Tipo di Segreto =  Chiave segreta**. Compariranno dunque altri campi da configurare che vediamo di seguito:

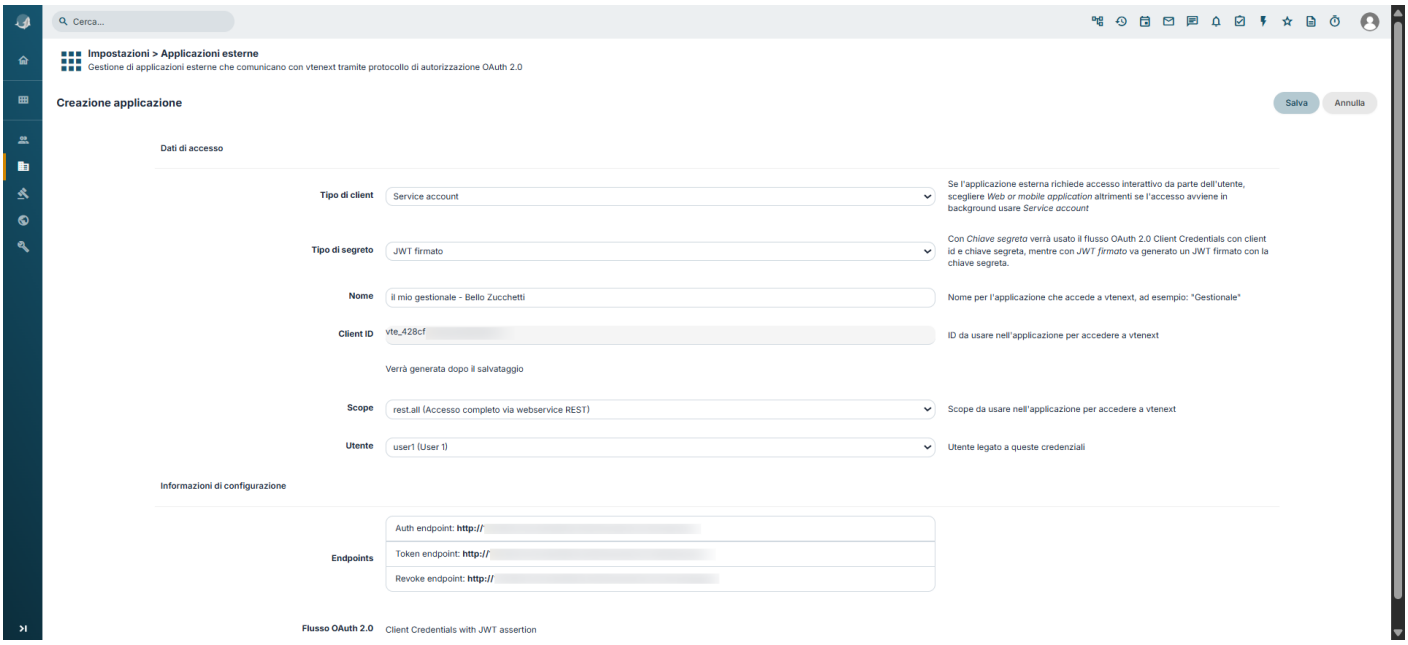



A questo punto si dovrà dare un **Nome** a questa applicazione, che tipicamente identifica anche quello che fa (nell'esempio fa connettere ad un gestionale). Vengono già messe in evidenza il **Client ID e la Chiave Segreta** autogenerati. Infine si deve determinare lo **Scope**, ovvero quali operazioni sono ammesse dal sistema (sola lettura, sola scrittura, oppure in lettura/scrittura) e l' **Utente** con il quale effettuare le operazioni collegandosi via REST API con Access Token. Se lo scope è rest.all.read, sarà possibile chiamare solamente REST API che leggono dati. Se è invece rest.all.write, sarà possibile chiamare solamente REST API che scrivono dati. Con rest.all è possibile chiamare qualsiasi API.

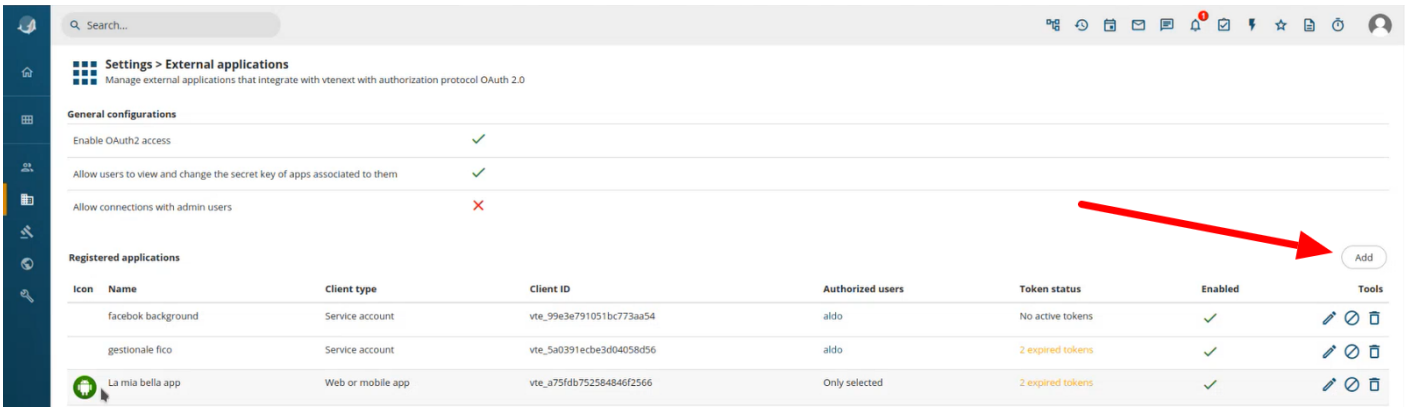
Ovviamente, le operazioni effettuabili e i dati restituiti sono comunque filtrati secondo la visibilità dell'**Utente** scelto, a seconda di ruoli e profili. Se si desidera pieno controllo senza limitazioni, dovrà essere selezionato un utente Admin.

Fornendo dunque gli Endpoints, il Client ID e la Chiave Segreta all'ipotetico tecnico del gestionale al quale ci si vuole connettere, sarà possibile finalizzare il collegamento.

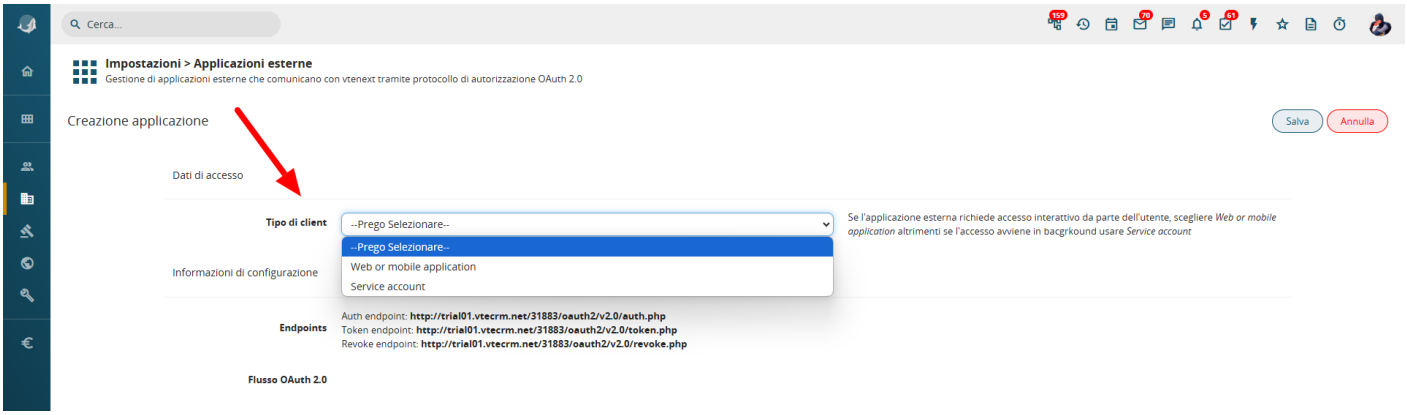
Tornando alla configurazione iniziale, se come **Tipo di Segreto** si sceglie invece **JWT firmato**, cambia solo il formato della **Chiave Segreta** che non sarà una semplice stringa casuale, ma una chiave crittografica in formato PEM o JWK, da inviare al tecnico del sistema esterno. Questa chiave viene generata solo dopo il salvataggio.



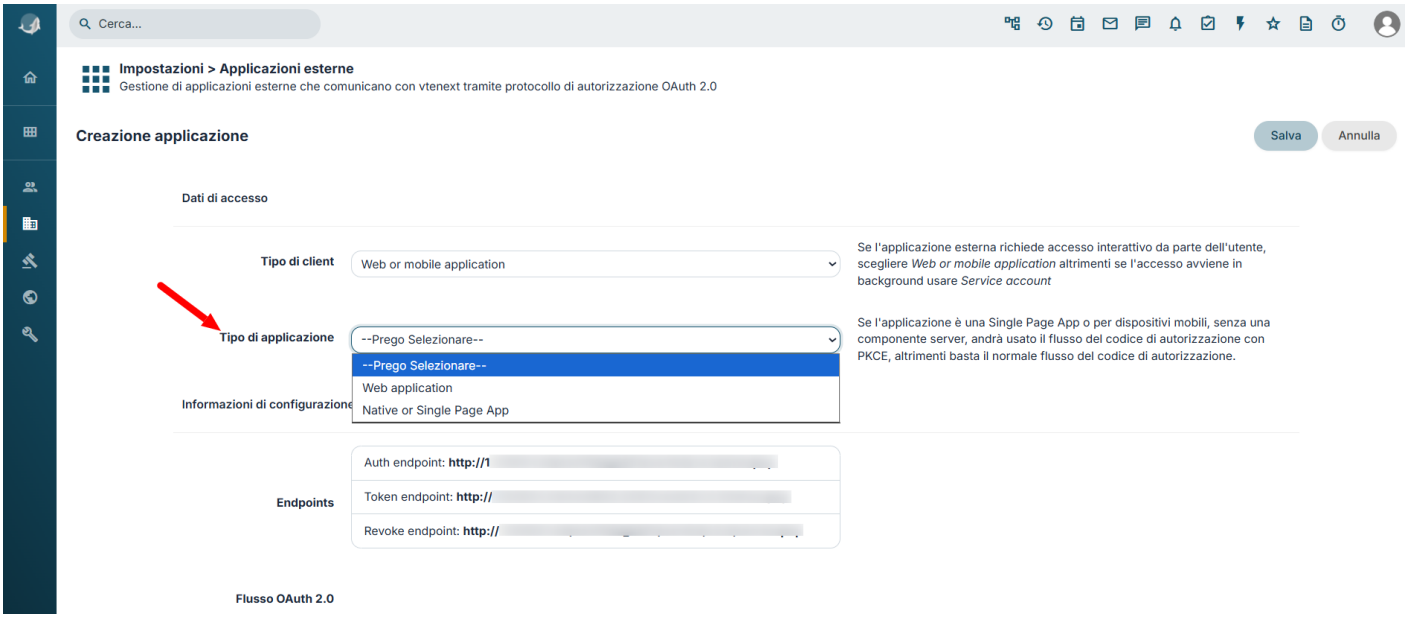

## Configurazione Web or Mobile Application




*Cliccare sul pulsante AGGIUNGI sulla destra*

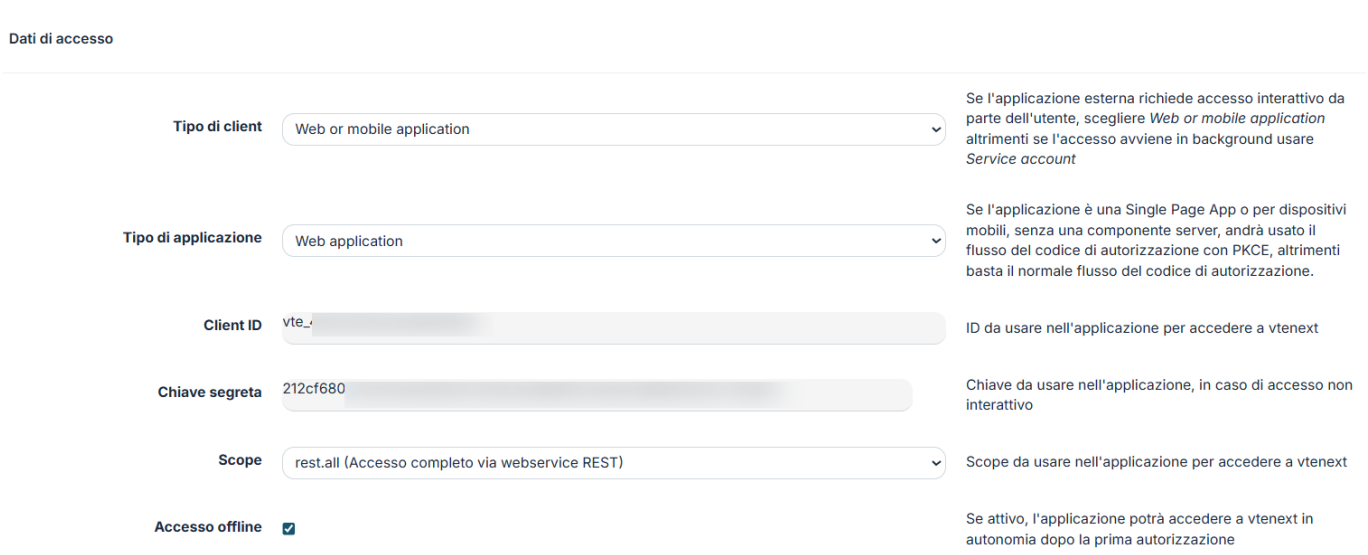



Si accede dunque nuovamente al Wizard e selezioneremo **Web or mobile application**. Comparirà la schermata seguente:

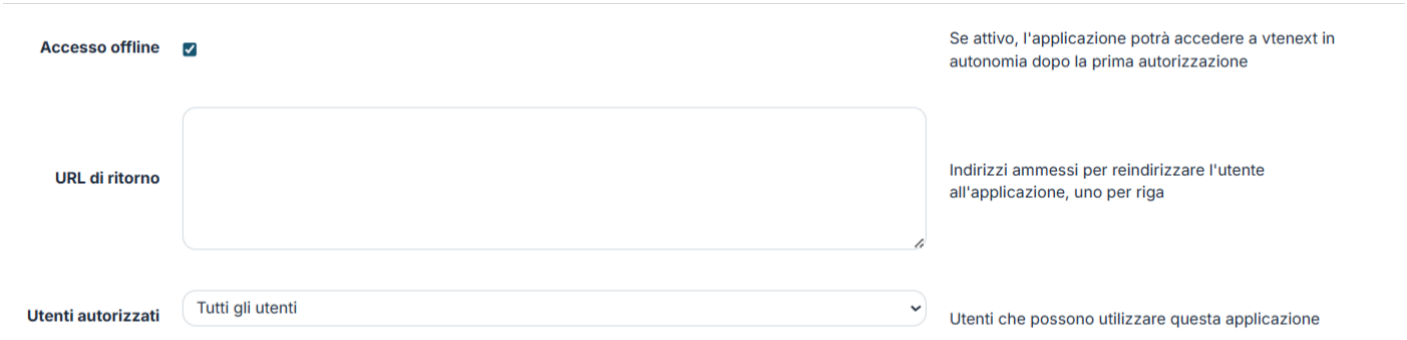



In **Tipo di applicazione**, è possibile scegliere **Web application** se si tratta di un App che funziona tramite browser, oppure **Native or Single Page App** se invece si tratta di un'applicazione nativa come un'app presente su un dispositivo mobile o una Single Page Application.

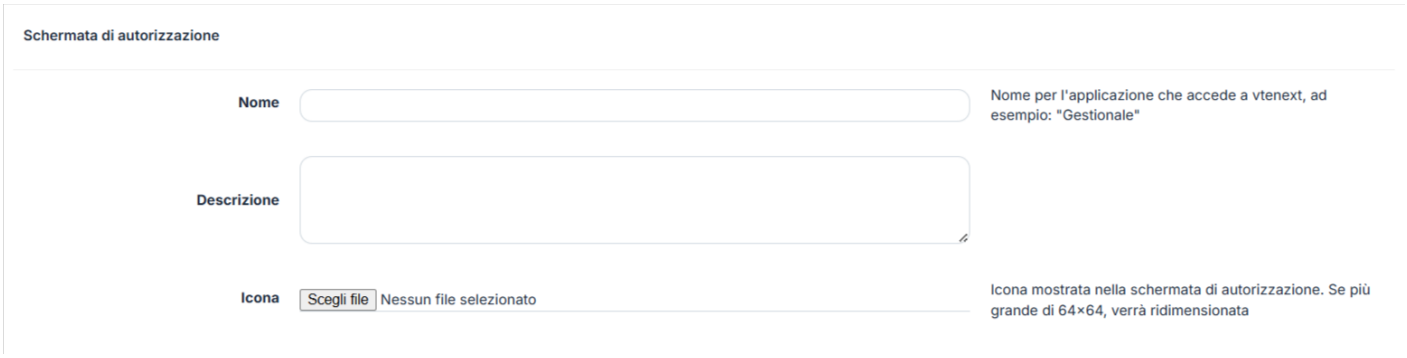
Scegliendo **Web application**, verrannno generati come per la configurazione vista in precedenza, un **Client ID** ed una **Chiave Segreta** e sarà possibile determinare il tipo di accesso, **Scope**, in sola lettura, sola scrittura o lettura/scrittura.



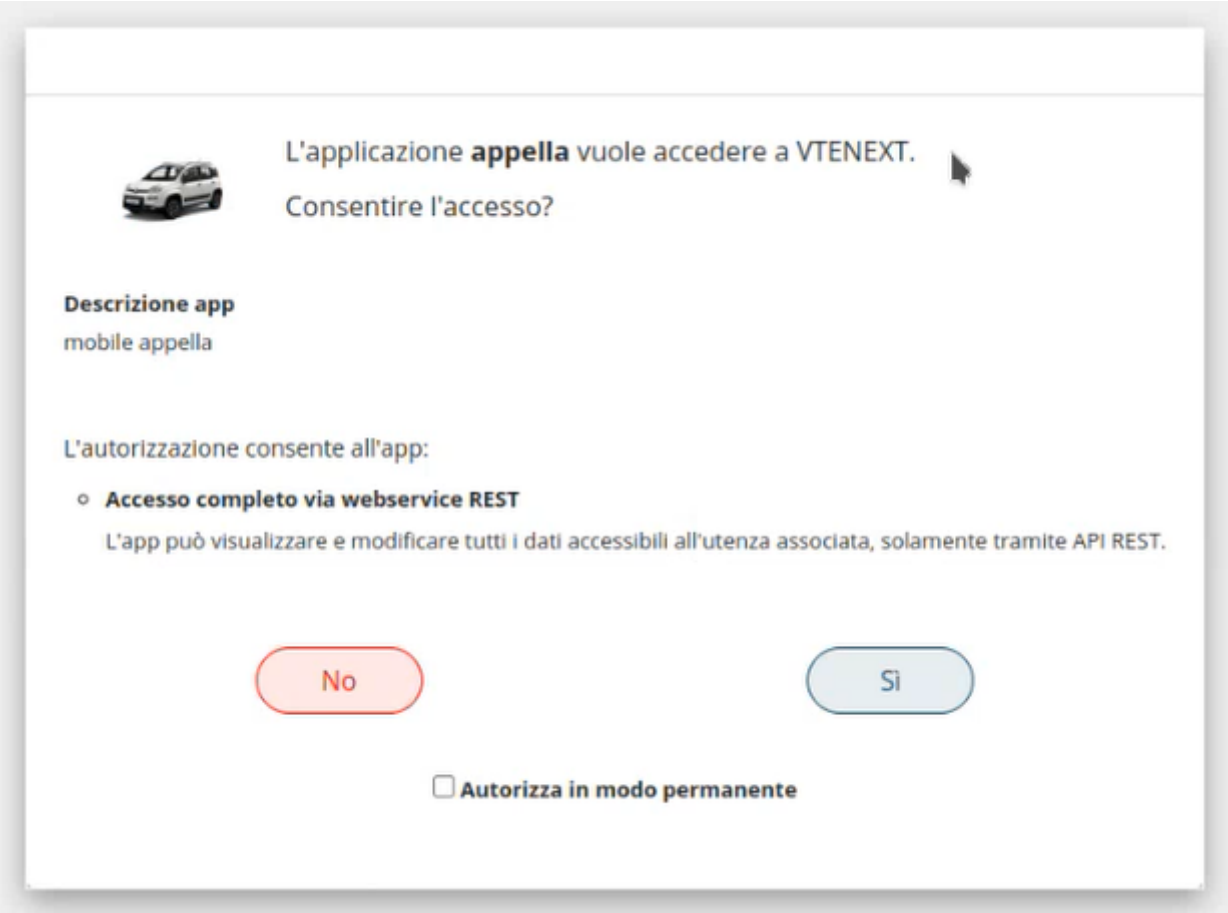

E' possibile decidere le attivare l'**Accesso offline**, ovvero invece di far scadere il Token come si diceva in precedenza (il Token ha una durata limitata e va rinnovato), in questo modo è possibile dare accesso a tempo indeterminato all'app verso vtenext. Quindi cosa succede? Il server chiede il "refresh token" e quindi l'app può richiedere nuovamente il token allo scadere, senza interazione con l'utente.




Quindi poi si configurerà l'**URL di ritorno**, che permetterà di reindirizzare l'utente, dopo il Login al Server (ovvero vtenext), direttamente all'app in cui deve accedere (es. Facebook o altro). E' anche possibile decidere quali sono gli **Utenti autorizzati** ad utilizzare questa configurazione. Possono essere Tutti gli utenti, oppure quelli che si vogliono selezionare dalla lista Gruppi o Utenti.



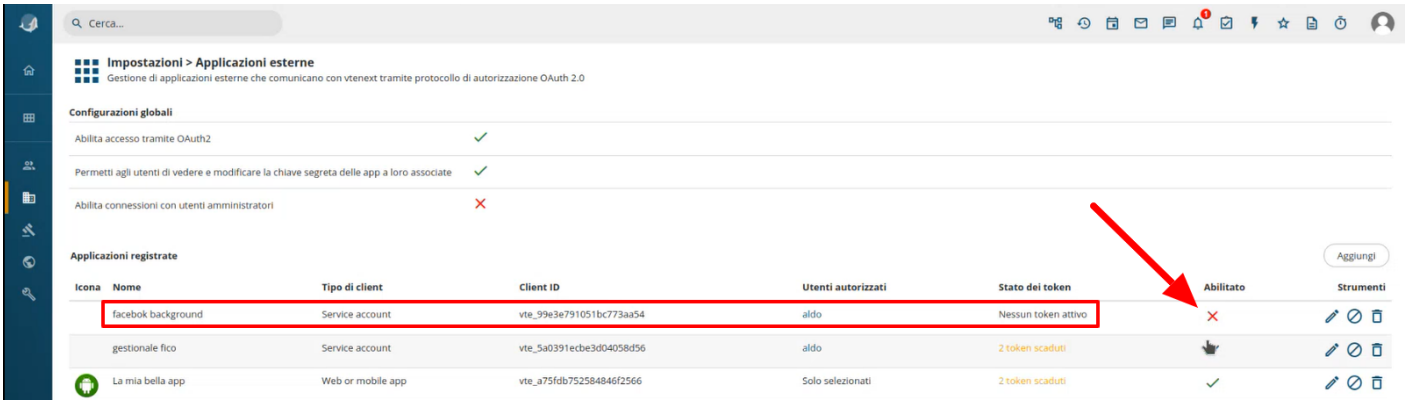

E' possibile configurare la Schermata di autorizzazione, ovvero quando l'utente viene indirizzato alla richiesta di autorizzazione (es. l'applicazione Facebook richiede di accedere ai dati di vtenext, vuoi autorizzare?), comparirà una schermata che è possibile personalizzare con un **Nome, una Descrizione ed un'Icona**.




Tornando all'inizio di questo tipo di settaggio, in **Tipo di applicazione** è possibile selezionare anche **Native or Single Page App**, la cui configurazione è identica a quella appena illustrata, con l'unica differenza che il flusso OAuth si aspetta di ricevere il PKCE (una misura di sicurezza aggiuntiva durante lo scambio dell'access token).

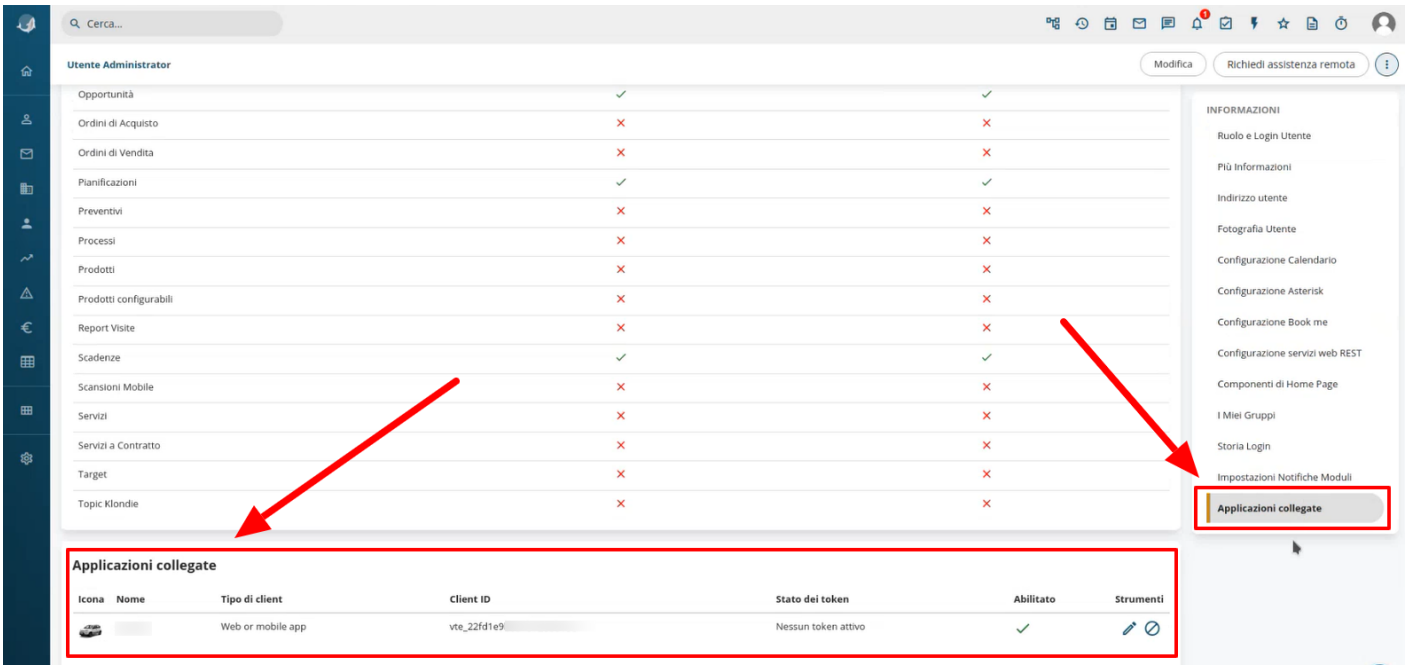
---

Dalla schermata inziale, **Impostazioni > Applicazioni esterne**, è possibile vedere la lista delle **Applicazioni registrate** ed è possibile decidere se disattivarle. Ad esempio, se non è più richiesto che Facebook possa accedere a vtenext, nonostante l'autorizzazione sia già stata configurata e concessa, è possibile disabilitarla cliccando sulla spunta verde sulla destra della lista.




In questa lista è anche possibile vedere se ci sono **Token scaduti** oppure di **non attivi**. Nel caso di Token scaduti l'utente è tenuto a rifare il login al sistema. Nella stessa lista, sulla destra, troviamo gli **Strumenti**:

- serve per modificare la configurazione
- per revocare tutti i Token
- per eliminare la configurazione eseguita




Nelle **Preferenze Utente**, c'è una voce sulla destra, denominata **Applicazioni collegate**, che permette di visualizzare tutte le applicazioni che sono associate a quell'utente, sia in modo statico (per il flusso Client Credentials), che dinamico (per il flusso Authorization Code, con token attivi). Da quella stessa related, è possibile vedere il dettaglio della configurazione (ma non modificarla), ed è possibile anche disabilitarla (se si è Amministratori) o revocarne i token.

---

Revision #7
Created 2026-01-28 14:54:41 UTC by Admin
Updated 2026-04-27 15:58:29 UTC by Alberto