



# Iframe Integration Guide

## Table of Contents

- Introduction ..... 1
- Required Configuration and Components..... 1
- Interacting with Total Expert Domains ..... 1
- Using the Total Expert Public API to Authenticate ..... 2
- Accessing the CRM ..... 3
- Public API: OAuth 2.0 Endpoints..... 3
  - GET Authentication Code..... 3
    - Query Parameters ..... 3
    - Expected Behavior and Responses..... 4
  - POST Fetch an Access Token ..... 4
    - Using an Authentication Code ..... 4
    - Using a Refresh Token..... 5
    - Expected Behavior and Responses..... 5
- SSO ..... 5

## Introduction

This guide provides a detailed overview of the domains and API you will need to successfully embed the Total Expert (TE) platform in an iframe.

Total Expert uses a standard OAuth 2.0 authorization flow. To learn more about the OAuth 2.0 process, review [the documentation from Auth0](#).

## Required Configuration and Components

To successfully integrate with TE, the following items are needed:

- Your TE-generated client ID and secret pair for OAuth 2.0 authentication
- Your TE-registered redirect URI
- Your code that manages access token acquisition and use.

## Interacting with Total Expert Domains

For your integration, you will need to interact with 3 domains:

- totalexpert.net



- public.totalexpert.net
- apps.totalexpert.net.

To display the TE platform in an iframe, your interface will:

1. Obtain an access token for your user using the OAuth 2.0 authentication code grant flow (using <https://totalexpert.net/authorize> and <https://public.totalexpert.net/v1/token>).
2. Supply the user's token when you request a page from [apps.totalexpert.net](https://apps.totalexpert.net).
3. Refresh or re-acquire the user's token as needed and supply the new token when fetching pages via iframe.
  - a. Access tokens expire after 1 hour.
  - b. Refresh tokens expire after 2 weeks.

## Using the Total Expert Public API to Authenticate

TE will provide you with a client ID and secret pair unique to your integration for OAuth2.0 authentication. Use your client ID and secret to obtain an access token for your user. The authentication process follows these steps:

Your interface requests an authentication code with the `crm` scope from the authorize page (<https://totalexpert.net/authorize>).

1. TE redirects your user to a Total Expert login page.
  - a. If the user is already logged in, this step is bypassed and the access token is returned to the callback URL.
  - b. Optionally, you can supply a service provider identifier to log the user in automatically. See Public API: OAuth 2.0 Endpoints below for more details.
2. Your user logs in to the Total Expert system with their TE credentials.
  - a. After login, TE presents the user with a confirmation page. The user must select **Grant Access** to proceed.
3. TE redirects to your preregistered callback URI and supplies the authorization code as a URL query parameter.

Your interface uses the authentication code to request an access token from the token endpoint (<https://public.totalexpert.net/v1/token>).

### Note

It is important that you make this request from a Web server, and not from the browser. Because of CORS access rules, the browser will often refuse to execute this request.

The preregistered callback URI (redirect URI) in step 3 is your custom callback. This callback must be supplied to TE to be associated with your client ID and secret. *You must register your redirect URI with TE for authentication to succeed.*

See Public API: OAuth 2.0 Endpoints below for the API request and response details, and redirect URI query parameter examples.



# Accessing the CRM

After authenticating successfully, use your user's access token to obtain TE platform pages from [apps.totalexpert.net](https://apps.totalexpert.net) and display them in your iframe. Use <https://apps.totalexpert.net/static/iframehelper.html> as your iframe's source URL, supplying your user's access token as a query parameter of the URL. With this method, the iframehelper manages its own iframe pointed to <https://apps.totalexpert.net>, and takes care of forming the needed Authorization header.

An example URL for the iframehelper.html is:

<https://apps.totalexpert.net/static/iframehelper.html?token=4a8043345a6aaa00622ec11b0de208269fc4c0d3&route=loan%2Flist>

The iframehelper supports the following query parameters:

Parameter	Description
token	Access token
route	URI-encoded CRM route to initially display in the iframehelper's iframe

In the previous example, the iframehelper's iframe would contain the <https://apps.totalexpert.net/loan/list> page.

## Public API: OAuth 2.0 Endpoints

This section documents how to retrieve an access token by using the OAuth2 Authorization Code Grant.

For more detail, visit [Total Expert Developer Resources](#) and the [Vendor OAuth Integration Guide](#).

### GET Authentication Code

Obtain an OAuth 2.0 authentication code.

```
https://totalexpert.net/authorize?response_type=code&client_id=AppClientID&scope=crm&state=myState
```

#### Query Parameters

This endpoint supports the following query parameters:

Parameter	Required?	Description
response_type	yes	Always code.
client_id	yes	The application's ID assigned by Total Expert.
scope	yes	A space-delimited list of strings (for example: <code>leadSurveyInteraction postLeads</code> ) that defines access that the application is requesting.
state	yes	The application's state. This is not modified and is returned to the redirect callback with the code.



Parameter	Required?	Description
<code>redirect_uri</code>	no	The application's redirect callback URI. This must match the registered string, which is the value used if this parameter is not supplied.
<code>serviceProvider</code>	no	The client's single sign-on (SSO) service provider to be used for SSO services.

### Expected Behavior and Responses

If the request is valid, the API returns a login page. Once the login is processed, the API redirects the caller to the registered `redirect_uri`, appending a certain set of query parameters to the URI.

The following is an example of a redirect URI for a successful login:

```
https://your.domain.com/oauth2/callback?code=bc9fc71b592e7e2bbbcc9453a2f0540b268ce7f5&state=myState
```

If the request is invalid, or if the user denies or fails the login attempt, the redirect still occurs, but the URI has different parameters. The following is an example of a redirect URI for a failed authentication request:

```
https://your.domain.com/oauth2/callback?error=access_denied&error_description=The+user+denied+access+to+your+application&state=myState
```

The query parameters that can appear in callback URIs are:

Parameter	When Included	Description
<code>code</code>	On success	Authorization code
<code>state</code>	Always	Application's state
<code>error</code>	On failure	Reason for failure
<code>error_description</code>	On failure	Description of failure

## POST Fetch an Access Token

Obtain an OAuth 2.0 access token.

```
https://public.totalexpert.net/v1/token
```

### Using an Authentication Code

This request requires the Authorization header and a body. The Authorization header should contain the string `Basic {{basic_auth}}`, where `{{basic_auth}}` is the Base64-encoded string `appClientID:appClientSecret`. The body should contain two key/value pairs:

- `grant_type` - `authorization_code`
- `code` - `{{authorization code}}`, which is the value sent to the redirect URI in the `code` query parameter



## Using a Refresh Token

An access token is only good for a certain amount of time, after which it expires and can no longer be used to access the system. If the client needs a new access token, the client can use the `refresh_token` returned from `/token` to request a new access token.

This request also requires the Authorization header and a body. The Authorization header should contain the string `Basic {{basic_auth}}` where `{{basic_auth}}` is the Base64-encoded string `appClientID:appClientSecret`. The body should contain two key/value pairs:

- `grant_type` - `refresh_token`
- `refresh_token` - `{{refresh_token}}`, which is the `refresh_token` value from the original response

## Expected Behavior and Responses

If the token is granted, the response JSON body contains:

Parameter	Description
<code>access_token</code>	A token to include in the Authorization header when accessing the system.
<code>expires_in</code>	Time, in seconds, until the token expires. If attempting to access the system with an expired token, access will be denied.
<code>token_type</code>	Always <code>Bearer</code> .
<code>scope</code>	A space-delimited list of strings that define the access scope of the token.
<code>refresh_token</code>	A token used to request a new access token.

If the request is invalid or fails, the response will include an `error` and an `error_description` in the response JSON body.

## SSO

Total Expert supports SSO capabilities. For your integration to leverage those SSO capabilities, you must also pass Total Expert the service provider name:

```
https://totalexpert.net/authorize?serviceProvider=<providerName>
```

This service provider name should be configurable in your platform and passed when requesting the OAuth token.