Single Sign-On

Last Modified on 07/08/2024 11:15 am EDT

Many IT departments have configured some kind of Single Sign-On (SSO) for users to have a single set of credentials to access common applications. If you're looking to integrate DevResults more fully with your existing IT environment, you now have the option to integrate your SSO provider with DevResults to handle user credential authentication. We support SSO integration with Azure Active Directory, with Active Directory via OAuth2, and with Okta. This page will walk you through how DevResults-SSO integration works and provide instructions on how to set it up.

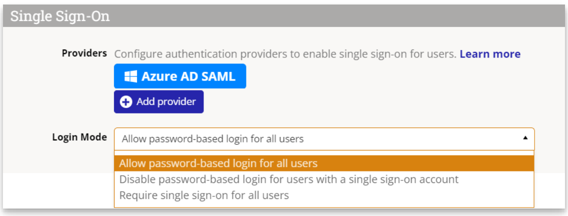
Jump to:

- **User Administration**
- **SSO Integration Setup**
  - **Azure Active Directory Setup**
  - **Active Directory via OAuth2 Setup**
  - **Okta Setup**
- **Frequently Asked Questions**

User Administration

User accounts in DevResults are created from the SSO provider once the user tries to log in to DevResults using those credentials and accepts the permissions.

The two accounts — SSO and DevResults — are linked but are independent. The **Login Mode** configuration determines whether or not password-based logins will still be allowed for some or any users once a SSO integration has been setup (see the following sections for instructions).



Single Sign-On

Providers: Configure authentication providers to enable single sign-on for users. Learn more

Login Mode: Allow password-based login for all users

Azure AD SAML
Add provider
Allow password-based login for all users
Disable password-based login for users with a single sign-on account
Require single sign-on for all users

- **Allow password-based login for all users**: any user can establish and use a password to log into DevResults.
- **Disable password-based login for users with a single sign-on account** any user with an active SSO account will be prevented from using a password-based login, but all other users (e.g. external partners, local staff, etc.) will still be able to log on using passwords.
- **Require single sign-on for all users:** password-based logins are fully disabled for all users; the SSO platform manages all access to DevResults for all users.

Please note that deleting or deactivating a user from Active Directory **does not** automatically delete or deactivate their account in DevResults. If they've only ever logged in using SSO, it will prevent them from logging in. However, if they ever established a DevResults password outside of SSO, they would still be able to access the site using that password. Therefore, we recommend still having a designated DevResults site administrator oversee permissions and account deactivation within DevResults.

back to top

SSO Integration Setup

We support SSO integration with Azure Active Directory and with Active Directory via OAuth2. For both setups, the overall process is:

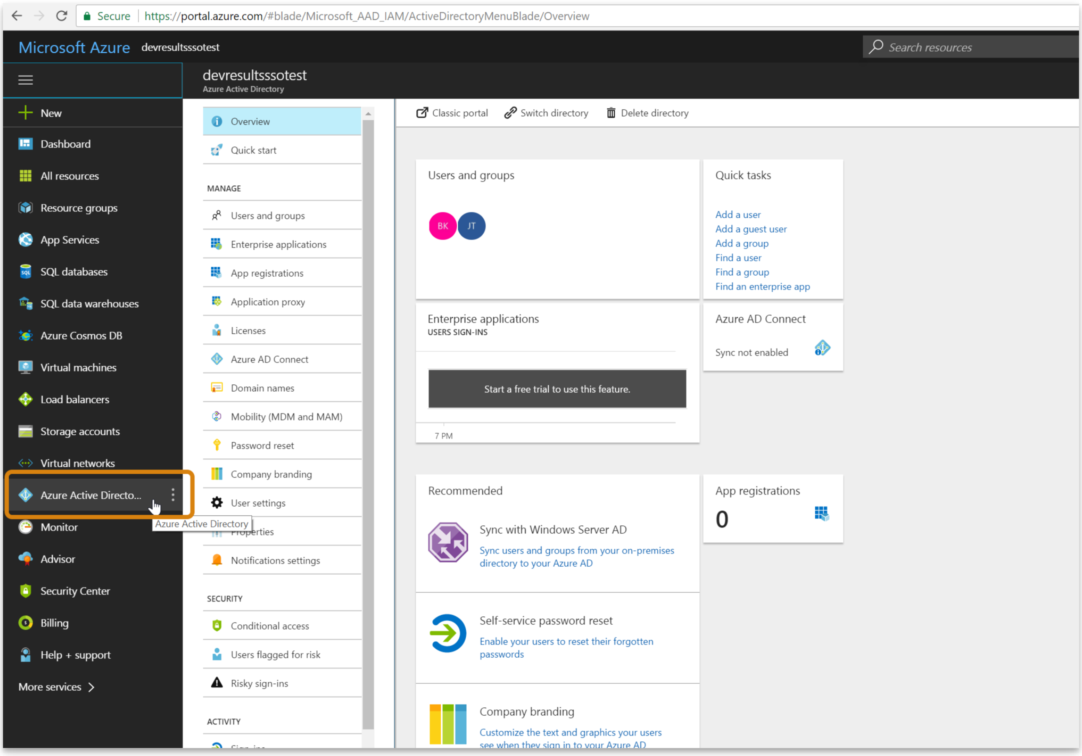
- Do some configuration and setup on your SSO provider side to gather the information you need.
- Enter some of that information in DevResults so it recognizes your SSO provider.

back to top

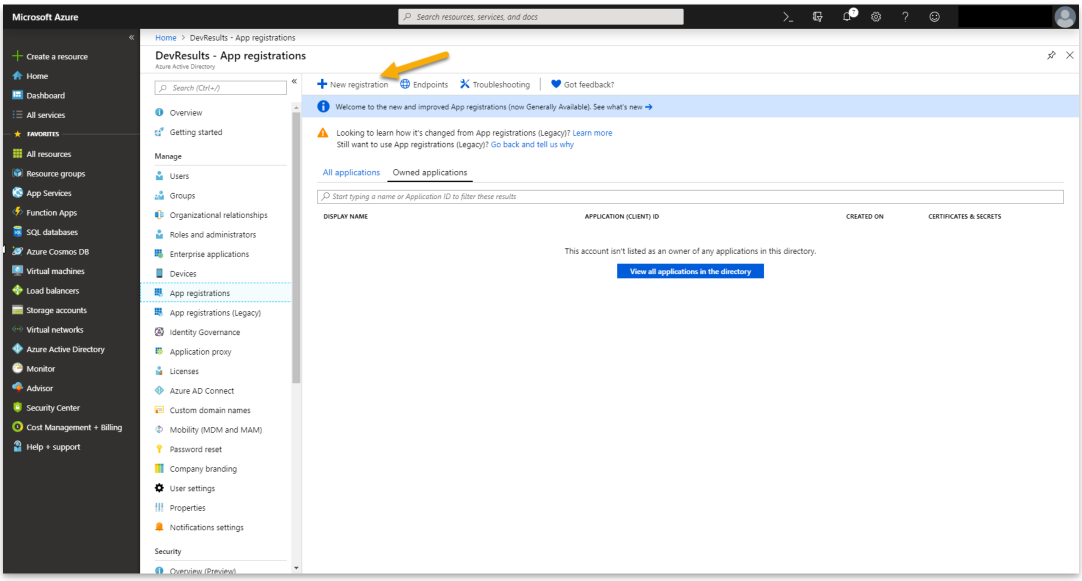
Azure Active Directory

For Azure Active Directory, you will need to have a registered application within Azure Active Directory, and you'll need to know the Application ID and have a Key generated. We'll provide a quick outline of those steps here as a sample and then walk you through the DevResults side of the setup.

In the Azure Portal, click "Azure Active Directory" in the sidebar to the left.



Select **App registrations**. Click on **New application registration** in the right-hand pane:

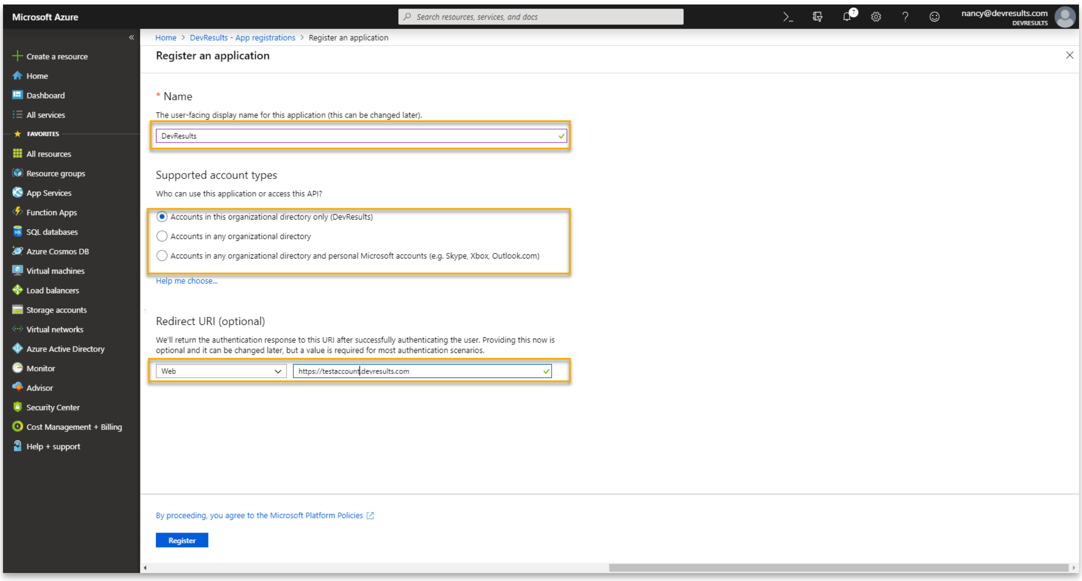


Welcome to the new and improved App registrations (now Generally Available). See what's new →

Looking to learn how it's changed from App registrations (Legacy)? Learn more
Still want to use App registrations (Legacy)? Go back and tell us why

This account isn't listed as an owner of any applications in this directory.

Enter a **Name** that will help you to identify this as your DevResults SSO config later. Next, select the accounts types you'd like to have access to DevResults. Click Register**.** For your **Redirect URI**, choose "Web" and enter your DevResults home page URL. Then click **Register.**



Register an application

* Name

The user-facing display name for this application (this can be changed later).

DevResults

Supported account types

Who can use this application or access this API?

● Accounts in this organizational directory only (DevResults)

○ Accounts in any organizational directory

○ Accounts in any organizational directory and personal Microsoft accounts (e.g. Skype, Xbox, Outlook.com)

Help me choose...

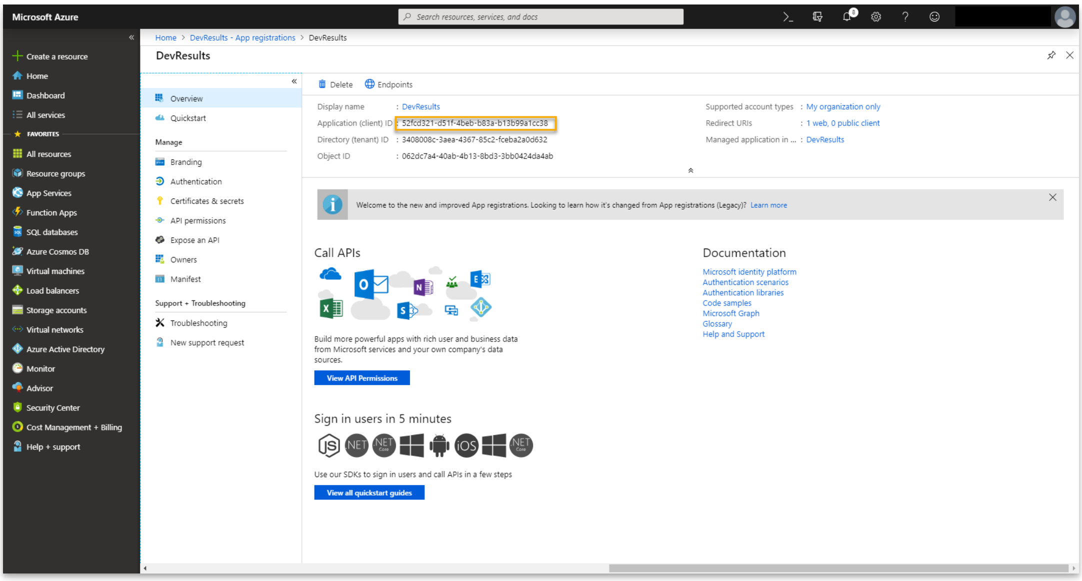
Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web | https://testaccount.devresults.com

By proceeding, you agree to the Microsoft Platform Policies

Once the application is created, click on it to open details. Copy the **Application ID**--you'll need this for the DevResults side of configuration.



Display name : DevResults

Application (client) ID : 52fcd321-d51f-4beb-b83a-b13b99a1cc38

Directory (tenant) ID : 3408008c-3aea-4367-85c2-fceba2a0d632

Object ID : 062dc7a4-40ab-4b13-8bd3-3bb0424da4ab

Supported account types : My organization only

Redirect URIs : 1 web, 0 public client

Managed application in ... : DevResults

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? Learn more

Call APIs

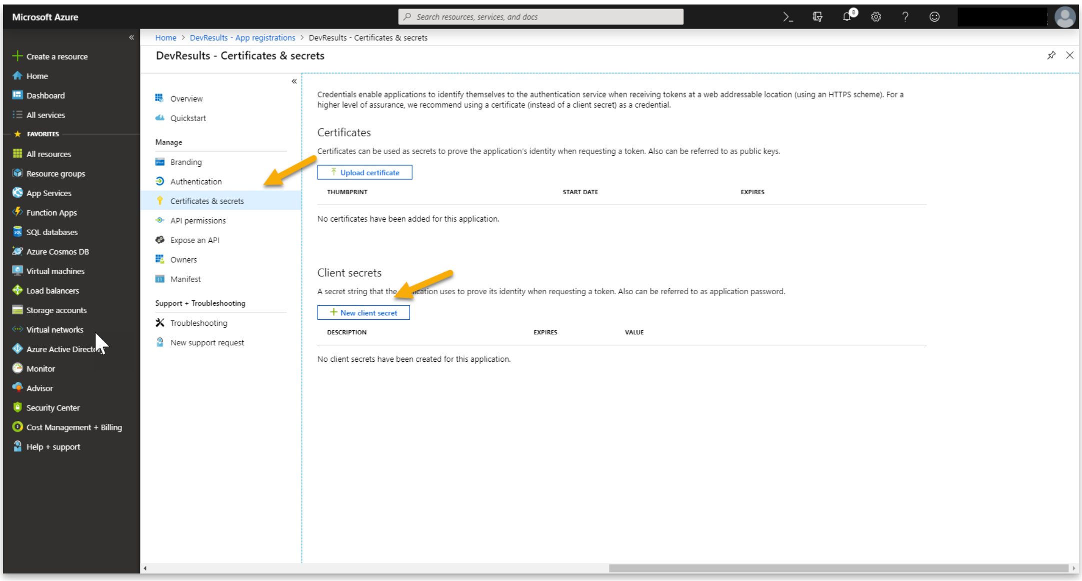
Build more powerful apps with rich user and business data from Microsoft services and your own company's data sources.

Documentation

Sign in users in 5 minutes

Use our SDKs to sign in users and call APIs in a few steps

Click **Certificates & Secrets** and then **New Client Secret**.



DevResults - Certificates & secrets

Credentials enable applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred to as public keys.

THUMBPRINT	START DATE	EXPIRES

No certificates have been added for this application.

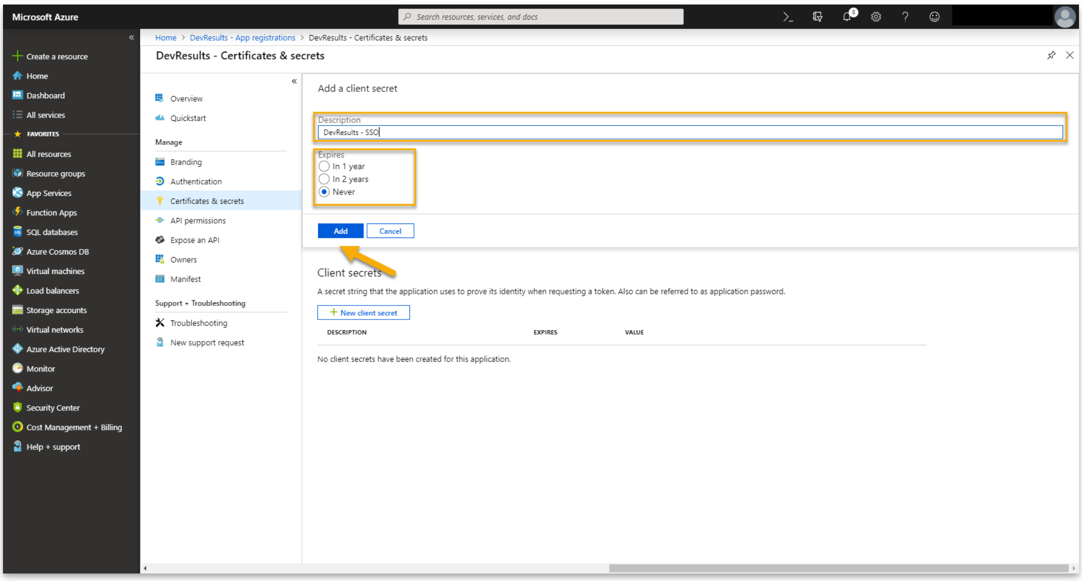
Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

DESCRIPTION	EXPIRES	VALUE

No client secrets have been created for this application.

Enter DevResults-SSO (or something else useful) in the **Description**. Select a **Duration** consistent with your organization's security policies. Then click **Add**.



DevResults - Certificates & secrets

Add a client secret

Description: DevResults - SSO

Expires
○ In 1 year
○ In 2 years
● Never

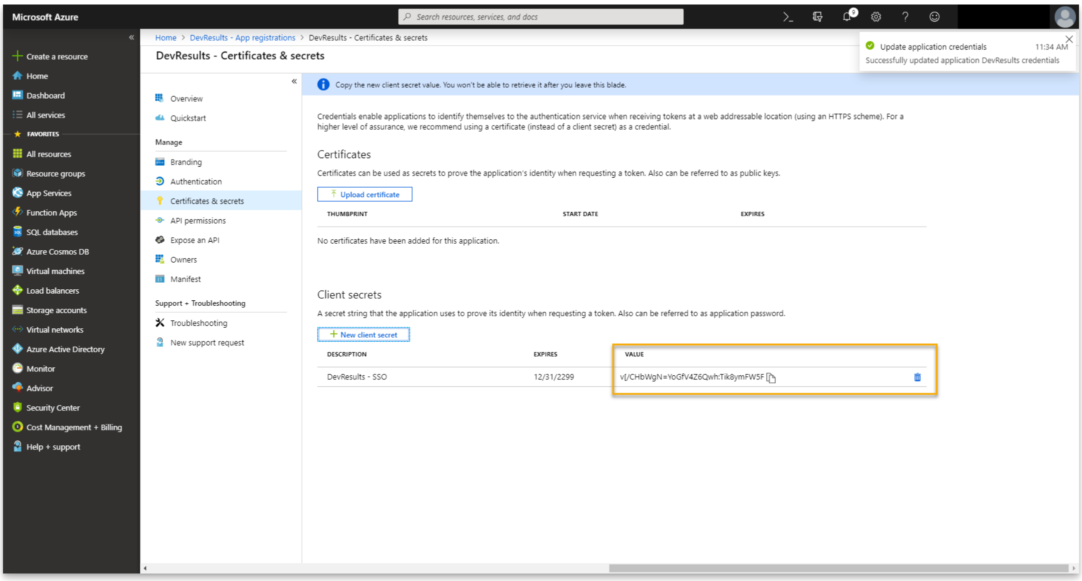
Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

DESCRIPTION	EXPIRES	VALUE

No client secrets have been created for this application.

When you click **Save**, Azure will generate a Key in the Value. Copy that **Value**. (If you don't copy this, you'll need to regenerate it later to add it to DevResults.)



Update application credentials 11:34 AM

Successfully updated application DevResults credentials

Copy the new client secret value. You won't be able to retrieve it after you leave this blade.

Credentials enable applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred to as public keys.

THUMBPRINT	START DATE	EXPIRES

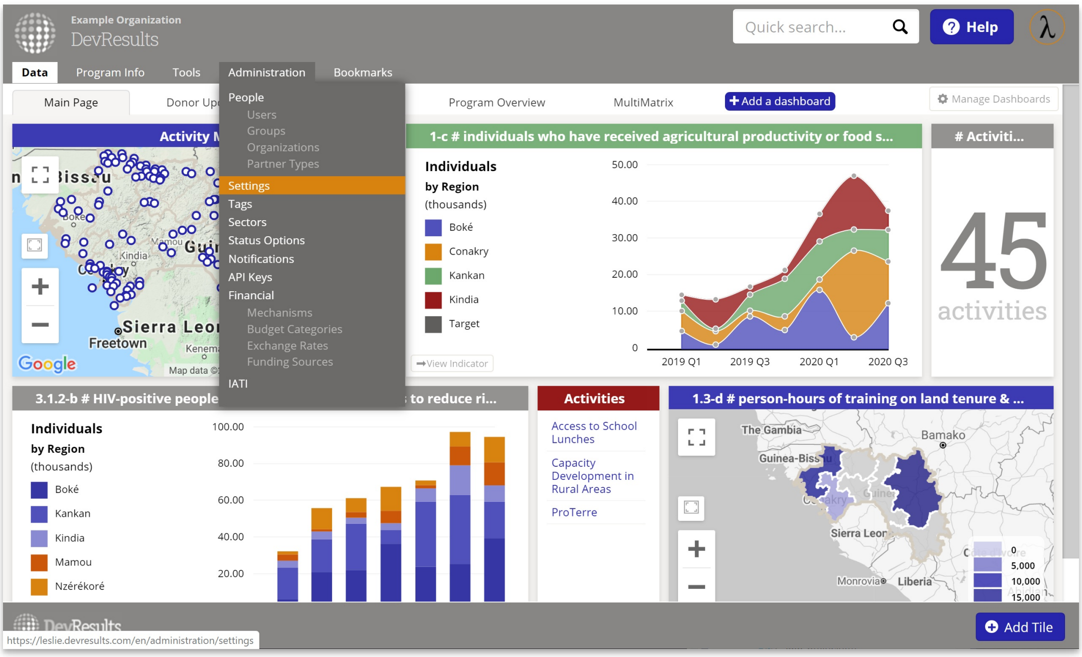
No certificates have been added for this application.

Client secrets

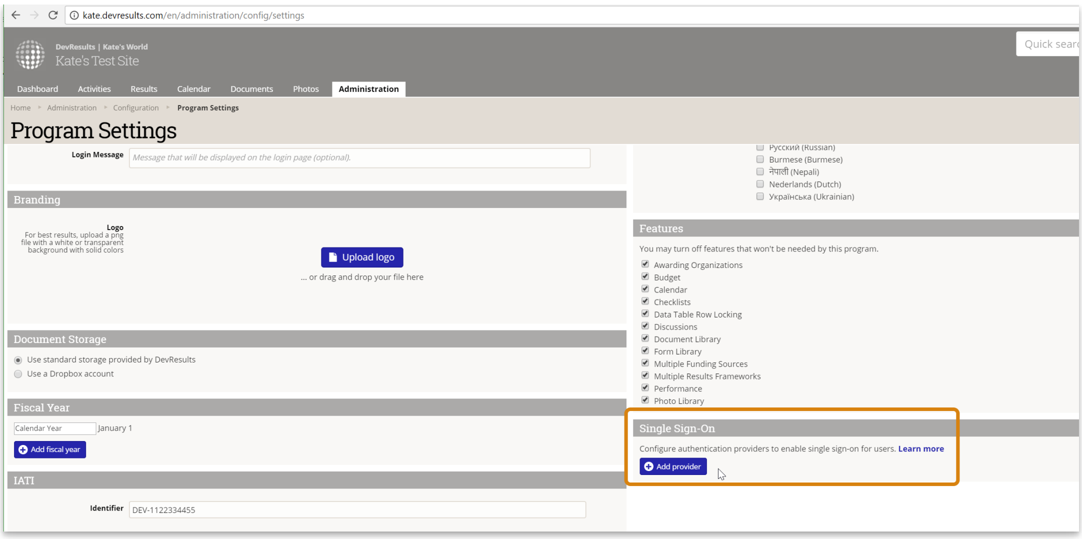
A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

DESCRIPTION	EXPIRES	VALUE
DevResults - SSO	12/31/2299	v[/CHbWgN=YoGfV4Z6Qwh:Tik8ymFW5F

We'll need to return to Azure Portal in a moment, but our next steps happen in the DevResults app. Now that there's an application in Azure Active Directory, we can set up the DevResults portion. In your DevResults site, go to **Administration > Settings**.



In the **Single Sign-On** section, click the **Add provider** button.



Program Settings

Login Message: Message that will be displayed on the login page (optional).

Branding

Logo
For best results, upload a png file with a white or transparent background with solid colors

... or drag and drop your file here

Document Storage

● Use standard storage provided by DevResults
○ Use a Dropbox account

Fiscal Year

Calendar Year | January 1

IATI

Identifier: DEV-1122334455

□ Русский (Russian)
□ Burmese (Burmese)
□ नेपाली (Nepali)
□ Nederlands (Dutch)
□ Українська (Ukrainian)

Features

You may turn off features that won't be needed by this program.

☑ Awarding Organizations
☑ Budget
☑ Calendar
☑ Checklists
☑ Data Table Row Locking
☑ Discussions
☑ Document Library
☑ Form Library
☑ Multiple Funding Sources
☑ Multiple Results Frameworks
☑ Performance
☑ Photo Library

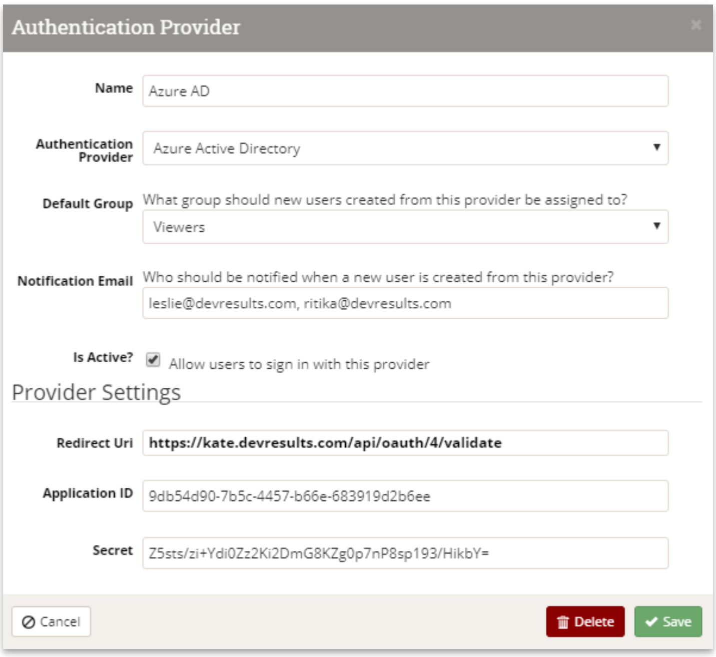
Single Sign-On

Configure authentication providers to enable single sign-on for users. Learn more

This will open the Authentication Provider pop-up. You'll need to complete these sections to configure the integration:

- **Name**: Provide a name for this SSO. We used Azure AD in our example. This label will appear to end-users on their login screen ("Use my {Name} account") so be sure it's something your users will understand!
- **Authentication Provider**: Currently DevResults supports Azure Active Directory and Active Directory via OAuth2. For this example, select Azure Active Directory.
- **Default Group**: When new users are created in DevResults from Active Directory, what group should they be added to by default? For our example, we used our standard **Users** group, but you can choose any existing group in your DevResults site.
- **Notification Email**: If you want anyone to be notified when a new DevResults user is created from Active Directory, enter their email address here. (Optional)
- **Is Active?**: You can uncheck this box if you don't want this provider to be currently used; otherwise, check the box to make sure it's going to be used.
- **Redirect Url:** This will autopopulate once you've completed the rest of the form. You'll need to copy this value - we'll be adding to to Azure Active Directory in a moment.
- **Provider Settings: Application ID**: Paste in the Application ID we copied from Azure Portal earlier
- **Provider Settings: Secret**: Paste in the Key's Value from the last set of steps in the Azure Portal directions

Once these fields are filled out, click the **Save** button to save your configuration settings.



Authentication Provider

Name: Azure AD

Authentication Provider: Azure Active Directory

Default Group: What group should new users created from this provider be assigned to?
Viewers

Notification Email: Who should be notified when a new user is created from this provider?
leslie@devresults.com, ritika@devresults.com

Is Active? ☑ Allow users to sign in with this provider

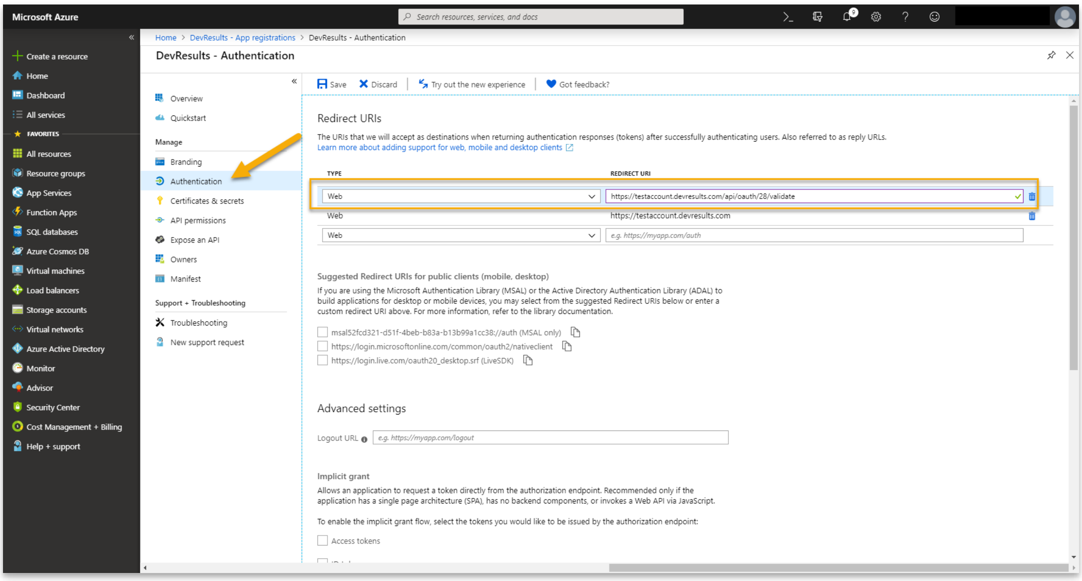
Provider Settings

Redirect Uri: https://kate.devresults.com/api/oauth/4/validate

Application ID: 9db54d90-7b5c-4457-b66e-683919d2b6ee

Secret: Z5sts/zi+Ydi0Zz2Ki2DmG8KZg0p7nP8sp193/HikbY=

Now, we'll need to return to Azure Active Directory. Click **Authentication**. Then add the value you copied from the Redirect URI field above. You can ignore the remainder of the Authentication page.



DevResults - Authentication

Redirect URIs

The URIs that we will accept as destinations when returning authentication responses (tokens) after successfully authenticating users. Also referred to as reply URLs.
Learn more about adding support for web, mobile and desktop clients

TYPE	REDIRECT URI
Web	https://testaccount.devresults.com/api/oauth/28/validate
Web	https://testaccount.devresults.com
Web	e.g. https://myapp.com/auth

Suggested Redirect URIs for public clients (mobile, desktop)

If you are using the Microsoft Authentication Library (MSAL) or the Active Directory Authentication Library (ADAL) to build applications for desktop or mobile devices, you may select from the suggested Redirect URIs below or enter a custom redirect URI above. For more information, refer to the library documentation.

☐ msal52fcd321-d51f-4beb-b83a-b13b99a1cc38://auth (MSAL only)
☐ https://login.microsoftonline.com/common/oauth2/nativeclient
☐ https://login.live.com/oauth20_desktop.srf (LiveSDK)

Advanced settings

Logout URL e.g. https://myapp.com/logout

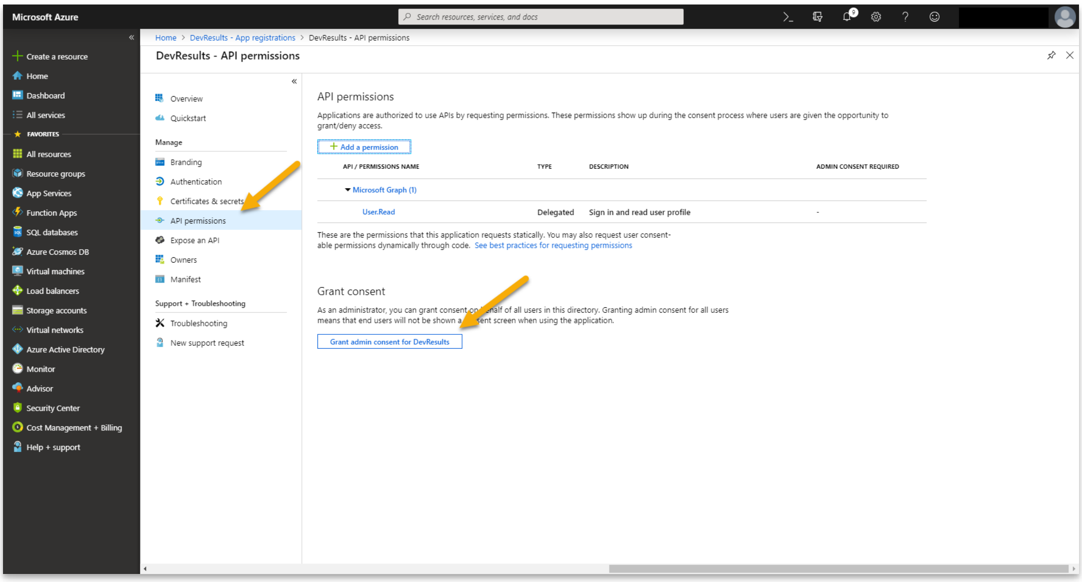
Implicit grant

Allows an application to request a token directly from the authorization endpoint. Recommended only if the application has a single page architecture (SPA), has no backend components, or invokes a Web API via JavaScript.

To enable the implicit grant flow, select the tokens you would like to be issued by the authorization endpoint:

☐ Access tokens

Now, click **API Permissions.** Then, click **Grant admin consent for DevResults**. (The text of this button may be different if you named the app something other than 'DevResults' in the initial configuration.)



DevResults - API permissions

API permissions

Applications are authorized to use APIs by requesting permissions. These permissions show up during the consent process where users are given the opportunity to grant/deny access.

+ Add a permission

API / PERMISSIONS NAME	TYPE	DESCRIPTION	ADMIN CONSENT REQUIRED
▾ Microsoft Graph (1)			
User.Read	Delegated	Sign in and read user profile	-

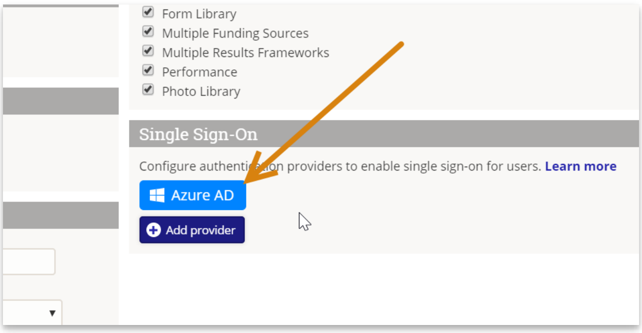
These are the permissions that this application requests statically. You may also request user consent-able permissions dynamically through code. See best practices for requesting permissions

Grant consent

As an administrator, you can grant consent on behalf of all users in this directory. Granting admin consent for all users means that end users will not be shown a consent screen when using the application.

Grant admin consent for DevResults

That's it! You've done everything you need to do in Azure. Back in DevResults, you'll see the provider appear on the DevResults Administration > Settings page. You can click to edit or delete it.

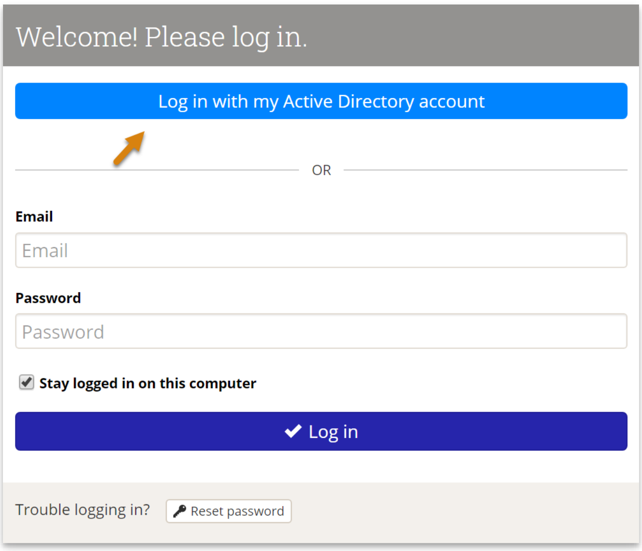


☑ Form Library
☑ Multiple Funding Sources
☑ Multiple Results Frameworks
☑ Performance
☑ Photo Library

Single Sign-On

Configure authentication providers to enable single sign-on for users. Learn more

With the SSO set up and enabled, your DevResults Login page will look slightly different. It will now show a "Log in with my ___ account" option as well as the regular login. Users can either use their DevResults username and password (if they already have one) or their SSO.



Welcome! Please log in.

Email

Email

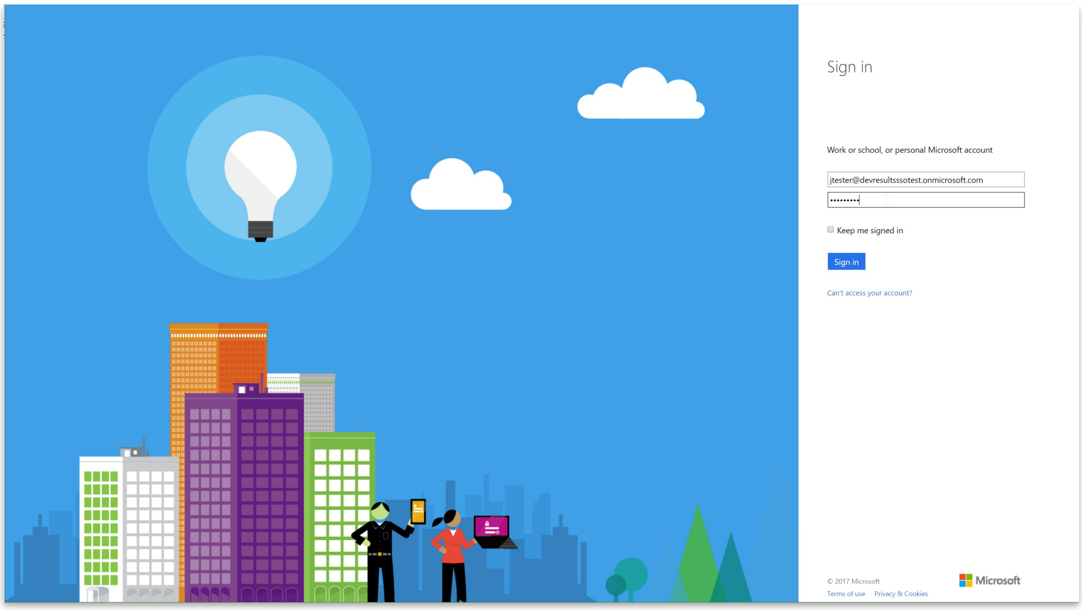
Password

Password

☑ Stay logged in on this computer

Trouble logging in?

The first time they log in using the Azure AD account, they'll be redirected to a Microsoft login screen where they'll need to enter their credentials for Active Directory.



Sign in

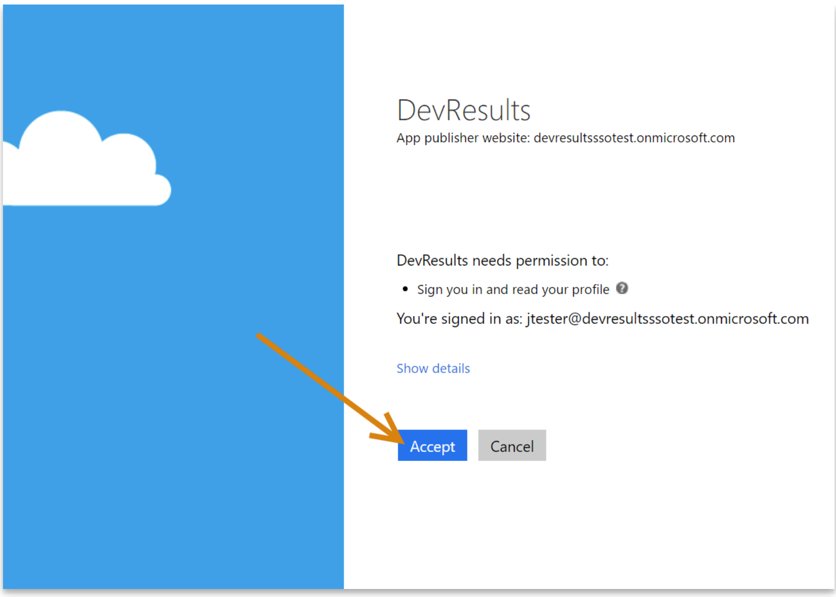
Work or school, or personal Microsoft account

jtester@devresultsssotest.onmicrosoft.com

•••••••••

□ Keep me signed in

Once they've entered credentials, they will need to grant DevResults permission to "sign you in and read your profile". This is only necessary the first time the user logs in using this method.



DevResults

App publisher website: devresultsssotest.onmicrosoft.com

DevResults needs permission to:

- Sign you in and read your profile

You're signed in as: jtester@devresultsssotest.onmicrosoft.com

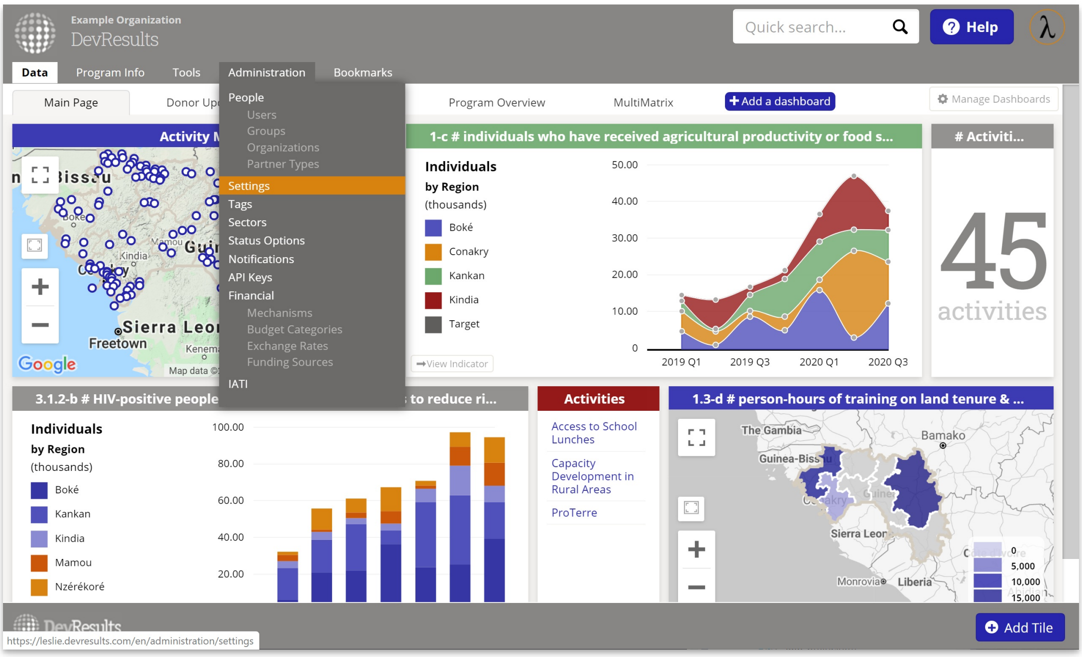
Once that's done, all future sign-ins using the Use my Azure AD account should just work. The page should now redirect to the DevResults site.

back to top

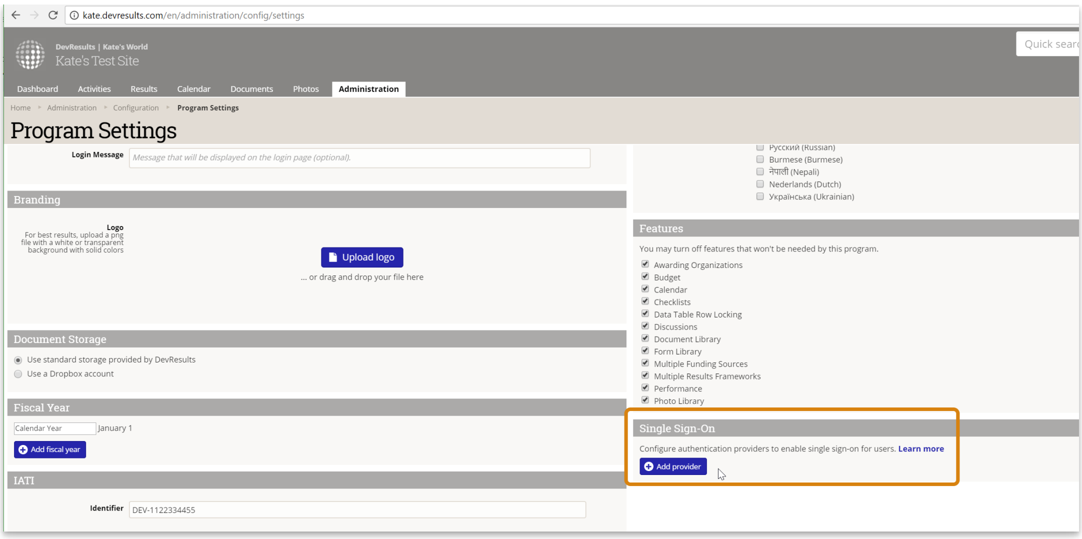
Active Directory via OAuth2

Configuring an Active Directory authentication via OAuth2 is fairly similar to the Azure Active Directory setup, except the configuration on the AD side must be done via Powershell script. **Here**'s a good general set of instructions on how to do so. You'll need to specify the endpoint, a resource name, and the full Redirect URI.

Once these are set up, you can set up the DevResults portion. In your DevResults site, go to **Administration > Settings**.



In the **Single Sign-On** section, click the **Add provider** button.



kate.devresults.com/en/administration/config/settings

DevResults | Kate's World
Kate's Test Site

Quick searc

Dashboard · Activities · Results · Calendar · Documents · Photos · **Administration**

Home ▸ Administration ▸ Configuration ▸ **Program Settings**

Program Settings

Login Message *Message that will be displayed on the login page (optional).*

Branding

Logo
For best results, upload a png file with a white or transparent background with solid colors

Upload logo
... or drag and drop your file here

Document Storage

● Use standard storage provided by DevResults
○ Use a Dropbox account

Fiscal Year

Calendar Year January 1

Add fiscal year

IATI

Identifier DEV-1122334455

☐ Русский (Russian)
☐ Burmese (Burmese)
☐ नेपाली (Nepali)
☐ Nederlands (Dutch)
☐ Українська (Ukrainian)

Features

You may turn off features that won't be needed by this program.

☑ Awarding Organizations
☑ Budget
☑ Calendar
☑ Checklists
☑ Data Table Row Locking
☑ Discussions
☑ Document Library
☑ Form Library
☑ Multiple Funding Sources
☑ Multiple Results Frameworks
☑ Performance
☑ Photo Library

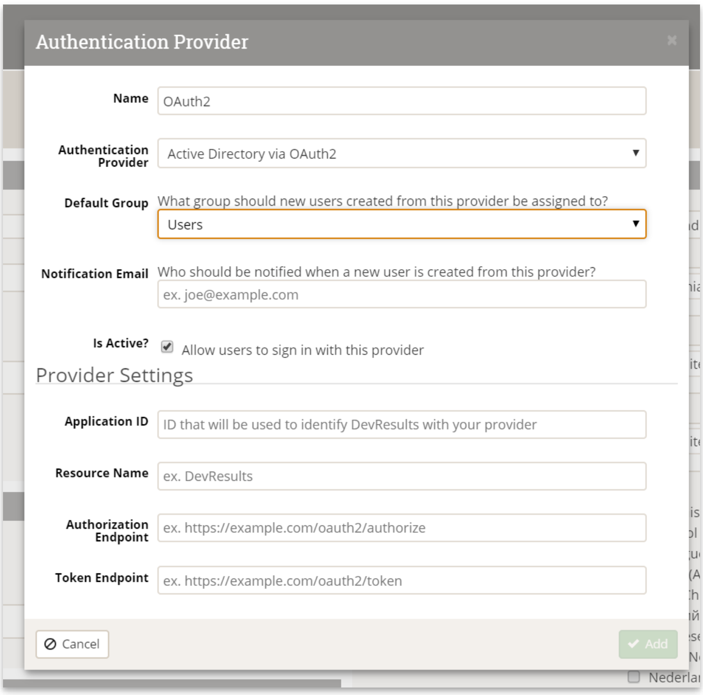
Single Sign-On

Configure authentication providers to enable single sign-on for users. **Learn more**

Add provider

This will open the Authentication Provider pop-up. You'll need to complete these sections to configure the integration:

- **Name**: Provide a name for this SSO. We used OAuth2 in our example. This label will appear to end-users on their login screen ("Use my {Name} account") so be sure it's something your users will understand!
- **Authentication Provider**: Currently DevResults supports Azure Active Directory and Active Directory via OAuth2. For this example, select Active Directory via OAuth2.
- **Default Group**: When new users are created in DevResults from Active Directory, what group should they be added to by default? For our example, we used our standard **Users** group, but you can choose any existing group in your DevResults site.
- **Notification Email**: If you want anyone to be notified when a new DevResults user is created from Active Directory, enter their email address here. (Optional)
- **Is Active?**: You can uncheck this box if you don't want this provider to be currently used; otherwise, check the box to make sure it's going to be used.
- **Provider Settings: Application ID**: Paste in the Application ID you configured in your PowerShell script
- **Provider Settings: Resource Name**: Use the Resource Name you configured in your PowerShell script
- **Provider Settings: Authorization Endpoint:** Use the Endpoint you configured in your PowerShell script
- **Token Endpoint**: Use the token endpoint you configured in your PowerShell script



Authentication Provider

Name OAuth2

Authentication Provider Active Directory via OAuth2 ▾

Default Group What group should new users created from this provider be assigned to?
Users ▾

Notification Email Who should be notified when a new user is created from this provider?
ex. joe@example.com

Is Active? ☑ Allow users to sign in with this provider

Provider Settings

Application ID ID that will be used to identify DevResults with your provider

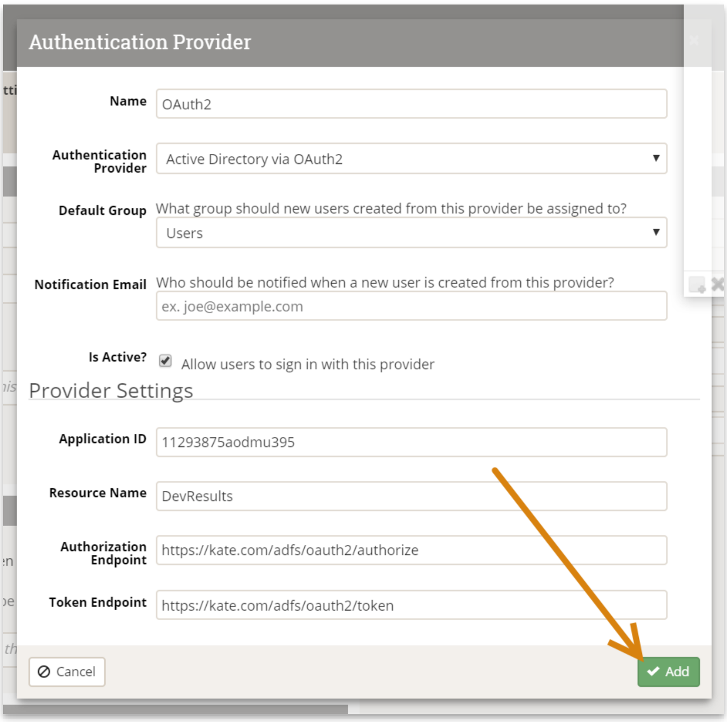
Resource Name ex. DevResults

Authorization Endpoint ex. https://example.com/oauth2/authorize

Token Endpoint ex. https://example.com/oauth2/token

Cancel · ✔ Add

Once these fields are filled out, click the **Add** button to save your configuration settings.



Authentication Provider

Name: OAuth2

Authentication Provider: Active Directory via OAuth2

Default Group: What group should new users created from this provider be assigned to? Users

Notification Email: Who should be notified when a new user is created from this provider? ex. joe@example.com

Is Active? ☑ Allow users to sign in with this provider

Provider Settings

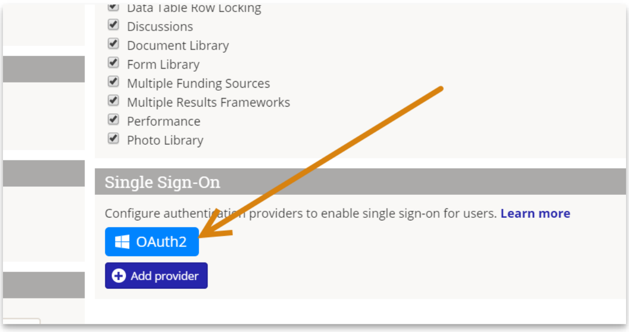
Application ID: 11293875aodmu395

Resource Name: DevResults

Authorization Endpoint: https://kate.com/adfs/oauth2/authorize

Token Endpoint: https://kate.com/adfs/oauth2/token

Once it's added, you'll see the provider appear on the DevResults Administration / Settings page. You can click to edit or delete it.

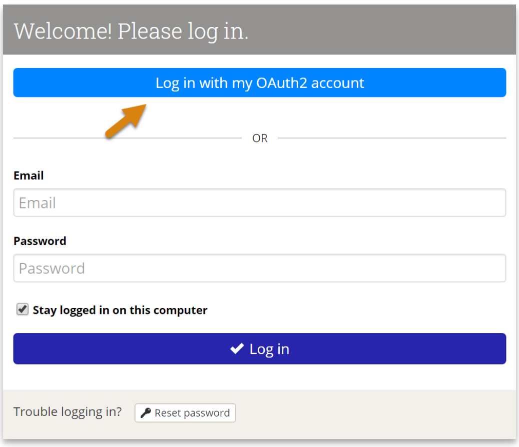


☑ Data Table Row Locking
☑ Discussions
☑ Document Library
☑ Form Library
☑ Multiple Funding Sources
☑ Multiple Results Frameworks
☑ Performance
☑ Photo Library

Single Sign-On

Configure authentication providers to enable single sign-on for users. Learn more

With the SSO set up and enabled, your DevResults Login page will look slightly different. It will now show a "Log in with my ___ account" option as well as the regular login. Users can either use their DevResults username and password (if they already have one) or their SSO.



Welcome! Please log in.

☑ Stay logged in on this computer

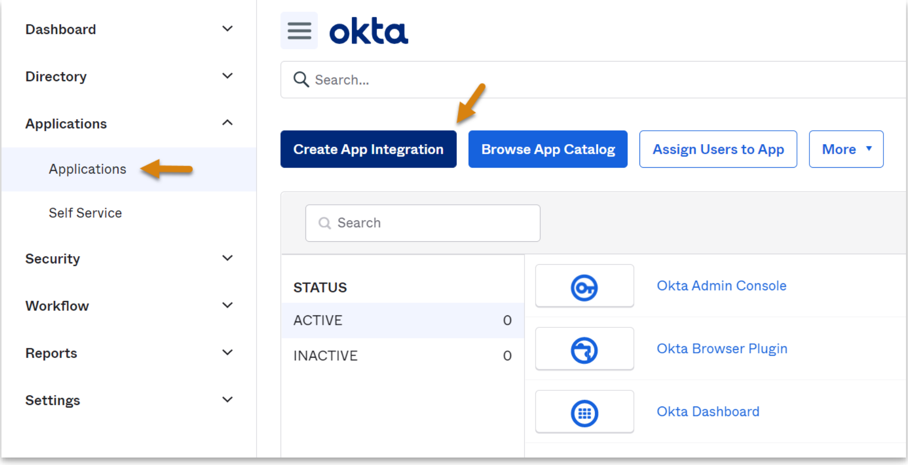
The first time they log in using the OAuth2 account, they'll be redirected to page based on your OAuth2 settings to enter credentials and grant DevResults access.

back to top

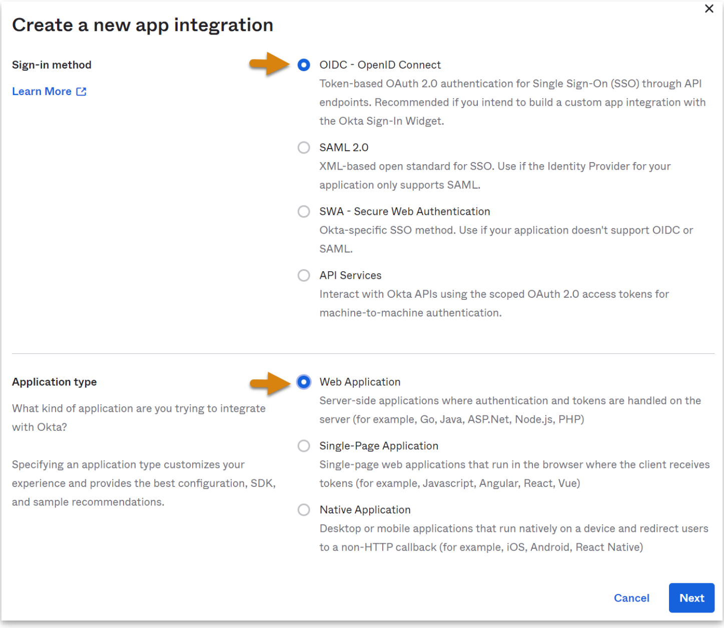
Okta Setup

DevResults can be integrated with your existing Okta environment in just a few easy steps. The first part of this process is also documented in **Okta's developer documentation**, replicated here for your convenience.

1. Sign into your Okta Admin Console and go to **Applications->Applications** (from the left hand side menu) and click on **Create App Integration**.



2. Select either **ODIC - OpenID Connect** or **SAML 2.0** as the sign-in method and **Web Application** as the application type, then click **Next**.



Create a new app integration

Sign-in method

● OIDC - OpenID Connect
Token-based OAuth 2.0 authentication for Single Sign-On (SSO) through API endpoints. Recommended if you intend to build a custom app integration with the Okta Sign-In Widget.

○ SAML 2.0
XML-based open standard for SSO. Use if the Identity Provider for your application only supports SAML.

○ SWA - Secure Web Authentication
Okta-specific SSO method. Use if your application doesn't support OIDC or SAML.

○ API Services
Interact with Okta APIs using the scoped OAuth 2.0 access tokens for machine-to-machine authentication.

Application type

What kind of application are you trying to integrate with Okta?

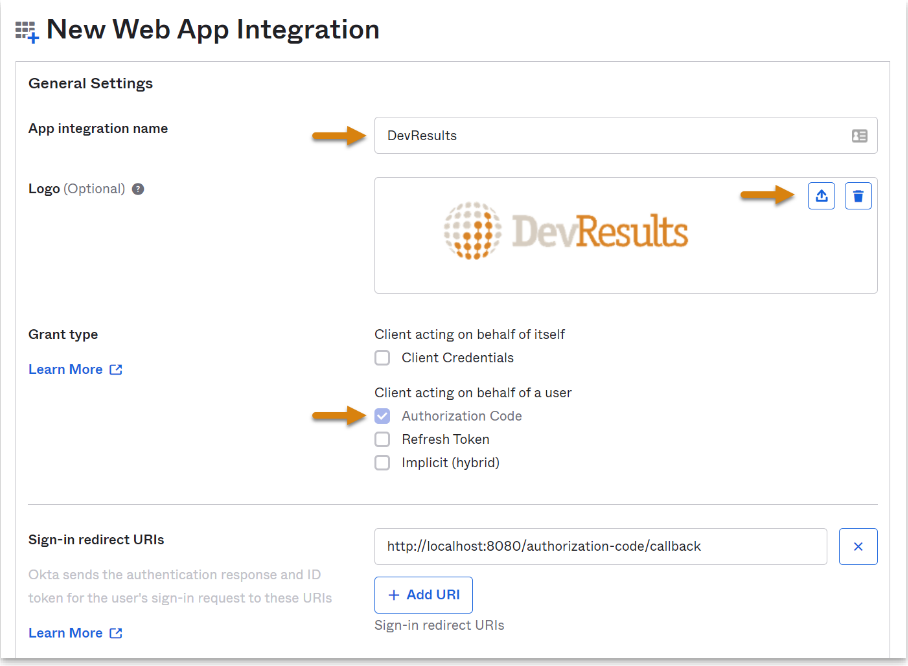
Specifying an application type customizes your experience and provides the best configuration, SDK, and sample recommendations.

● Web Application
Server-side applications where authentication and tokens are handled on the server (for example, Go, Java, ASP.Net, Node.js, PHP)

○ Single-Page Application
Single-page web applications that run in the browser where the client receives tokens (for example, Javascript, Angular, React, Vue)

○ Native Application
Desktop or mobile applications that run natively on a device and redirect users to a non-HTTP callback (for example, iOS, Android, React Native)

3. If this is an OIDC connectionn, specify the **App integration name** (e.g. DevResults), upload the DevResults logo (optional), leave the **Grant type** set to **Authorization Code**, and click **Save**. Later, you'll need to revise the **Sign-in redirect URI**, but you'll have to finish the next step in DevResults before you'll be able to change these fields.



New Web App Integration

General Settings

App integration name: DevResults

Logo (Optional)

Grant type

Learn More

Client acting on behalf of itself

□ Client Credentials

Client acting on behalf of a user

☑ Authorization Code

□ Refresh Token

□ Implicit (hybrid)

Sign-in redirect URIs

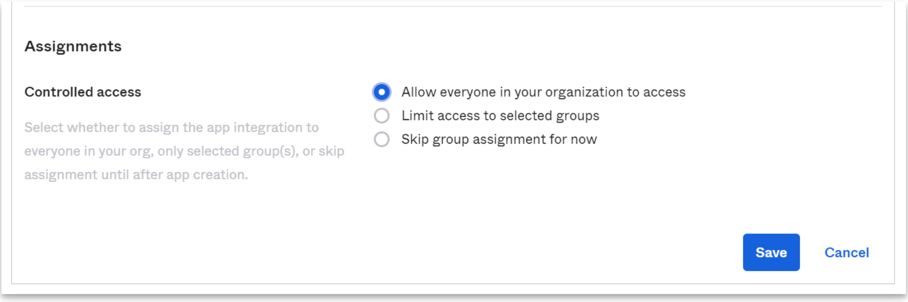
Okta sends the authentication response and ID token for the user's sign-in request to these URIs

Learn More

http://localhost:8080/authorization-code/callback

Sign-in redirect URIs

Note: if you have already established access groups, you can **limit access to select groups** at the bottom of this page. If not, you can **allow everyone in your organization to access**DevResults via Okta SSO.



Assignments

Controlled access

Select whether to assign the app integration to everyone in your org, only selected group(s), or skip assignment until after app creation.

● Allow everyone in your organization to access

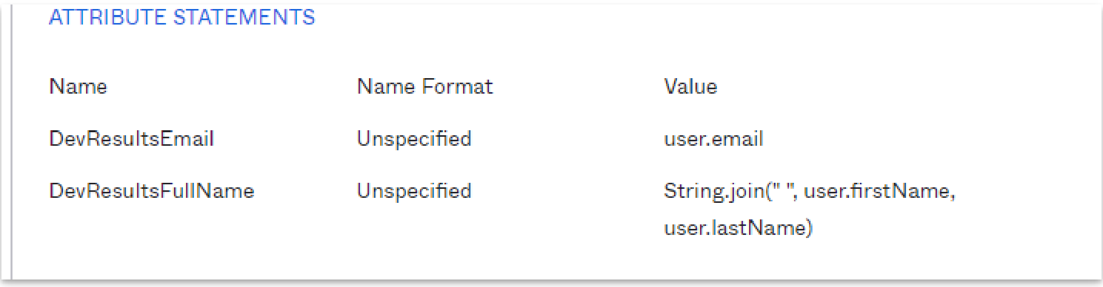
○ Limit access to selected groups

○ Skip group assignment for now

If you're connecting using SAML 2.0:

- Okta's **Single Sign-On URL** -> DevResults **Redirect URI**
- Okta's Audience URI -> https://{your instance subdomain}. **devresults.com/saml-sp**

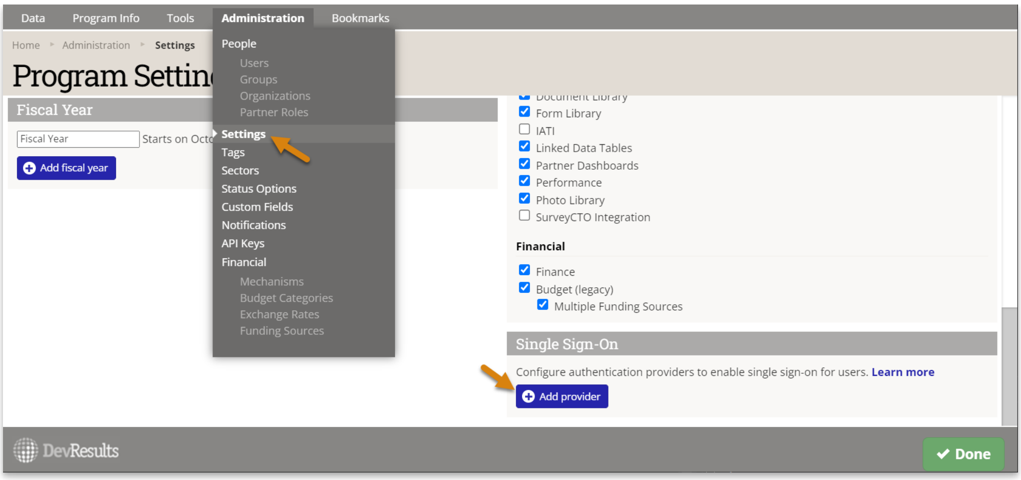
Add an Attribute Statement with Name "DevResultsEmail" and Value "user.email" and a second Attribute Statement with Name "DevResultsFullName" and Value String.join(" ", user.firstName, user.lastName) as below. Be careful, these are case sensitive!



ATTRIBUTE STATEMENTS

Name	Name Format	Value
DevResultsEmail	Unspecified	user.email
DevResultsFullName	Unspecified	String.join(" ", user.firstName, user.lastName)

4. In DevResults, go to **Administration > Settings**. In the **Single Sign-On** section, click the **Add provider** button.



- ☑ Document Library
- ☑ Form Library
- □ IATI
- ☑ Linked Data Tables
- ☑ Partner Dashboards
- ☑ Performance
- ☑ Photo Library
- □ SurveyCTO Integration

Financial

- ☑ Finance
- ☑ Budget (legacy)
  - ☑ Multiple Funding Sources

Configure authentication providers to enable single sign-on for users. Learn more

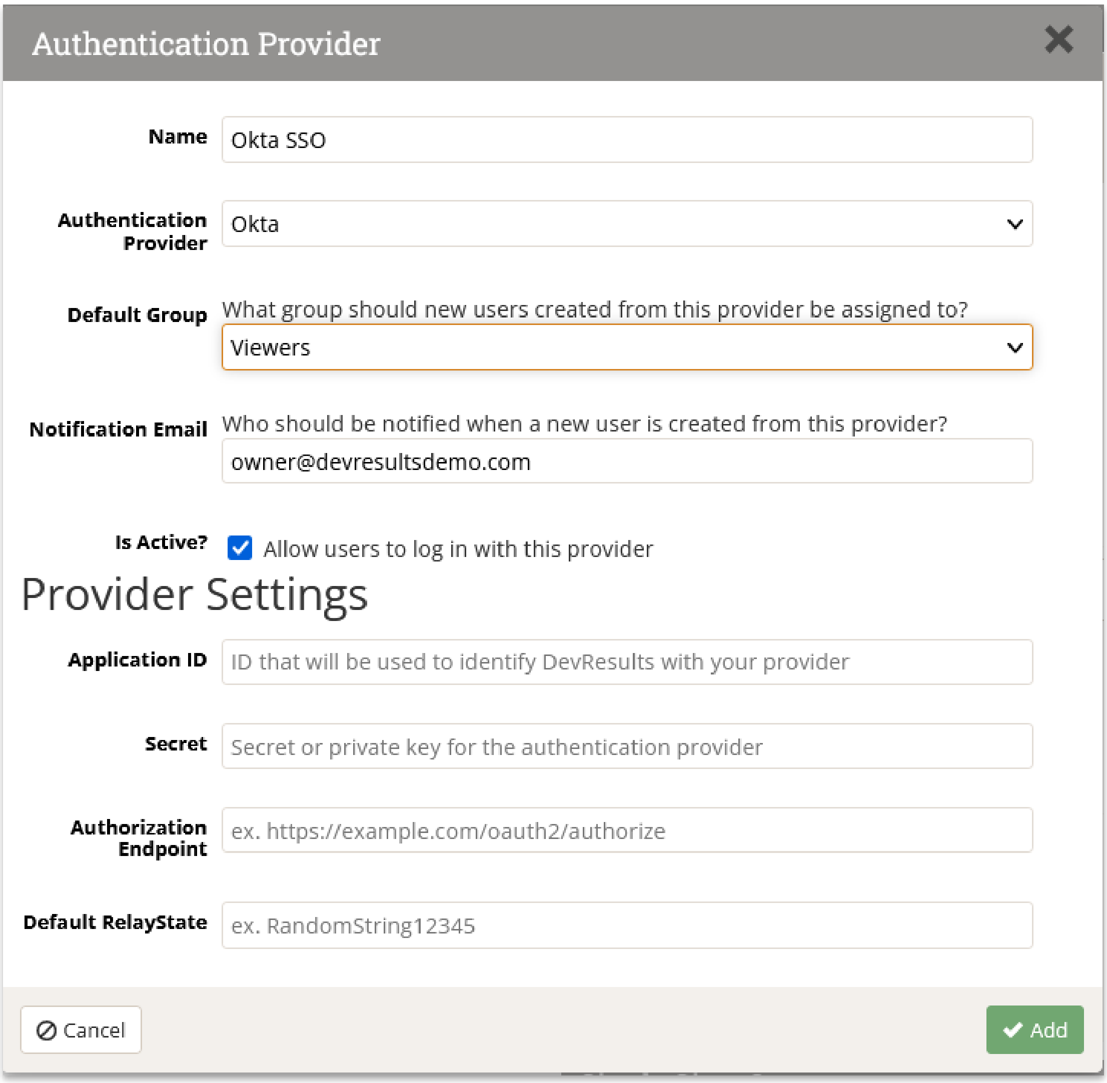
5. In the following screen, select **Okta** as the **Authentication Provider** and fill out the other fields accordingly. If connecting using ODIC, you'll need to get the following fields from Okta (after saving your configuration in step 3 above) to fill in the **Provider Settings** section:

- Okta's **Client ID** -> DevResults **Application ID**
- Okta's **Client secret** -> DevResults **Secret**
- Okta **domain** -> **Authorization Endpoint**, in the format **https://{Okta domain goes here}/oauth2**, e.g. *https://okta-server/oauth2*.

If using **SAML 2.0:**

- Okta's **Single Sign-On URL ->** DevResults **Authorization Endpoint**
- Okta's **Entity ID** -> DevResults **Application ID**
- Okta's **X.509 Certificate ->** DevResults **Secret** *(NOTE: include ----BEGIN CERTIFICATE----- and -----END CERTIFICATE----)*
- **Default RelayState** to any string value you want, just remember to add it to Okta's **Default Relay State**

Once every field is populated, click **Add**.



Authentication Provider

Name: Okta SSO

Authentication Provider: Okta

Default Group: What group should new users created from this provider be assigned to?
Viewers

Notification Email: Who should be notified when a new user is created from this provider?
owner@devresultsdemo.com

Is Active? ☑ Allow users to log in with this provider

Provider Settings

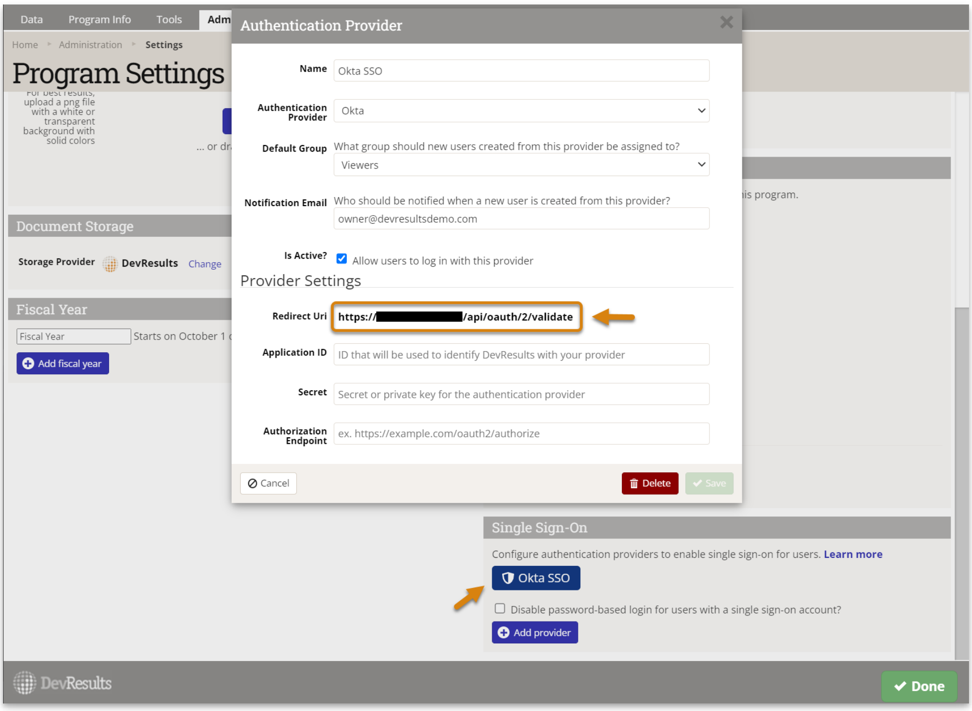
Application ID: ID that will be used to identify DevResults with your provider

Secret: Secret or private key for the authentication provider

Authorization Endpoint: ex. https://example.com/oauth2/authorize

Default RelayState: ex. RandomString12345

6. You should see a new authentication provider, with the name you provided in the previous step (e.g. Okta SSO). Click this button and copy the **Redirect URI** that has been created.



Data
Program Info
Tools
Adm
Home
Administration
Settings
Program Settings
For best results, upload a png file with a white or transparent background with solid colors
... or dr

Authentication Provider

Name: Okta SSO

Authentication Provider: Okta

Default Group: What group should new users created from this provider be assigned to?
Viewers

Notification Email: Who should be notified when a new user is created from this provider?
owner@devresultsdemo.com

Is Active? ☑ Allow users to log in with this provider

Provider Settings

Redirect Uri: https://[redacted]/api/oauth/2/validate

Application ID: ID that will be used to identify DevResults with your provider

Secret: Secret or private key for the authentication provider

Authorization Endpoint: ex. https://example.com/oauth2/authorize

Document Storage

Storage Provider: DevResults Change

Fiscal Year

Fiscal Year: Starts on October 1

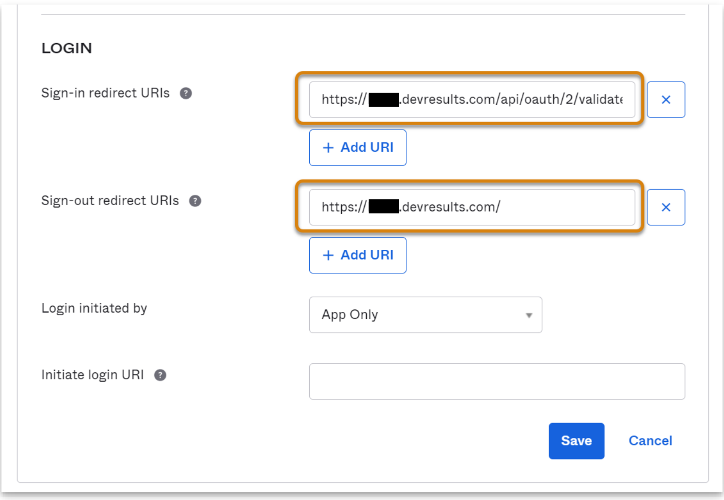
...is program.

Single Sign-On

Configure authentication providers to enable single sign-on for users. Learn more

□ Disable password-based login for users with a single sign-on account?

7. Back in Okta, click on **Edit** in the **General Settings** box and paste the Redirect URI from DevResults into the **Sign-in redirect URI** field. In the **Sign-out redirect URI** field, paste the same text but take off everything after devresults.com (in otherwords, just include the root domain).



LOGIN

Sign-in redirect URIs: https://[redacted].devresults.com/api/oauth/2/validate

Sign-out redirect URIs: https://[redacted].devresults.com/

Login initiated by: App Only

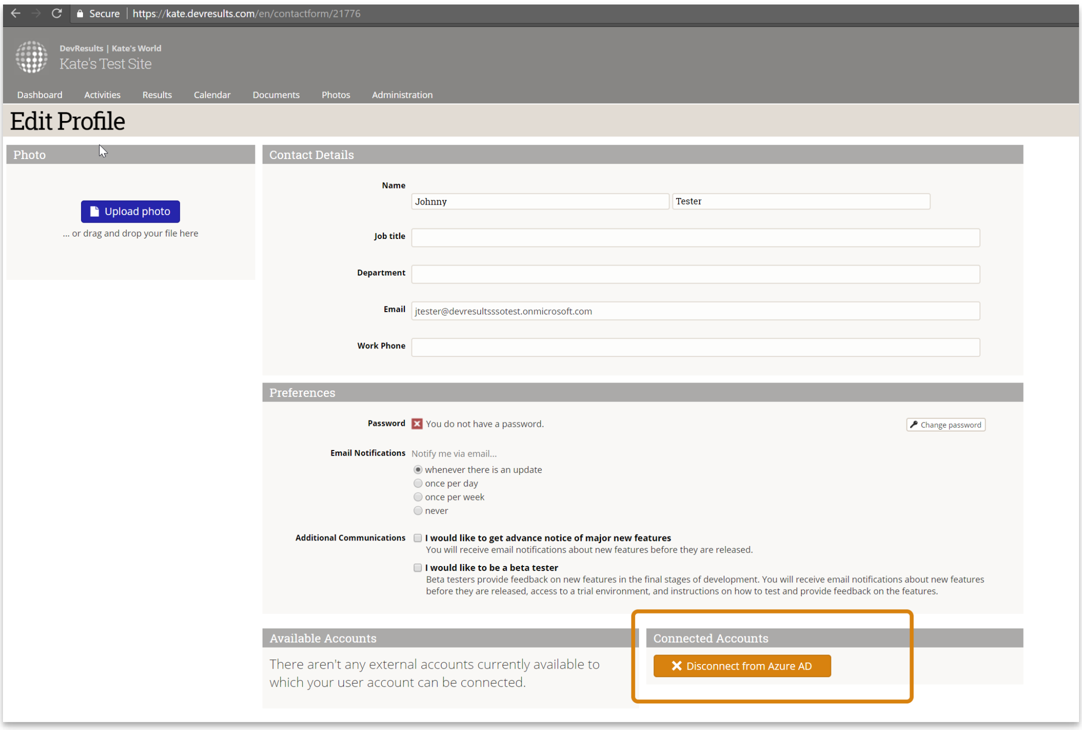
Initiate login URI

back to top

Frequently Asked Questions

How can I tell if I'm using a DevResults account or their Active Directory Account?

A user can tell if their DevResults account is related to an Active Directory account in their Profile details. Click on your profile picture in the upper right and select **Edit Profile**. The Connected Accounts section will have an entry if you're using Active Directory:



https://kate.devresults.com/en/contactform/21776

DevResults | Kate's World

Kate's Test Site

Edit Profile

Photo

... or drag and drop your file here

Contact Details

Name: Johnny Tester

Job title

Department

Email: jtester@devresultsssotest.onmicrosoft.com

Work Phone

Preferences

Password: ☒ You do not have a password.

Email Notifications: Notify me via email...

● whenever there is an update
○ once per day
○ once per week
○ never

Additional Communications:

□ **I would like to get advance notice of major new features**
You will receive email notifications about new features before they are released.

□ **I would like to be a beta tester**
Beta testers provide feedback on new features in the final stages of development. You will receive email notifications about new features before they are released, access to a trial environment, and instructions on how to test and provide feedback on the features.

Available Accounts

There aren't any external accounts currently available to which your user account can be connected.

Connected Accounts

You can disconnect this relationship by clicking the **Disconnect from...** button here.

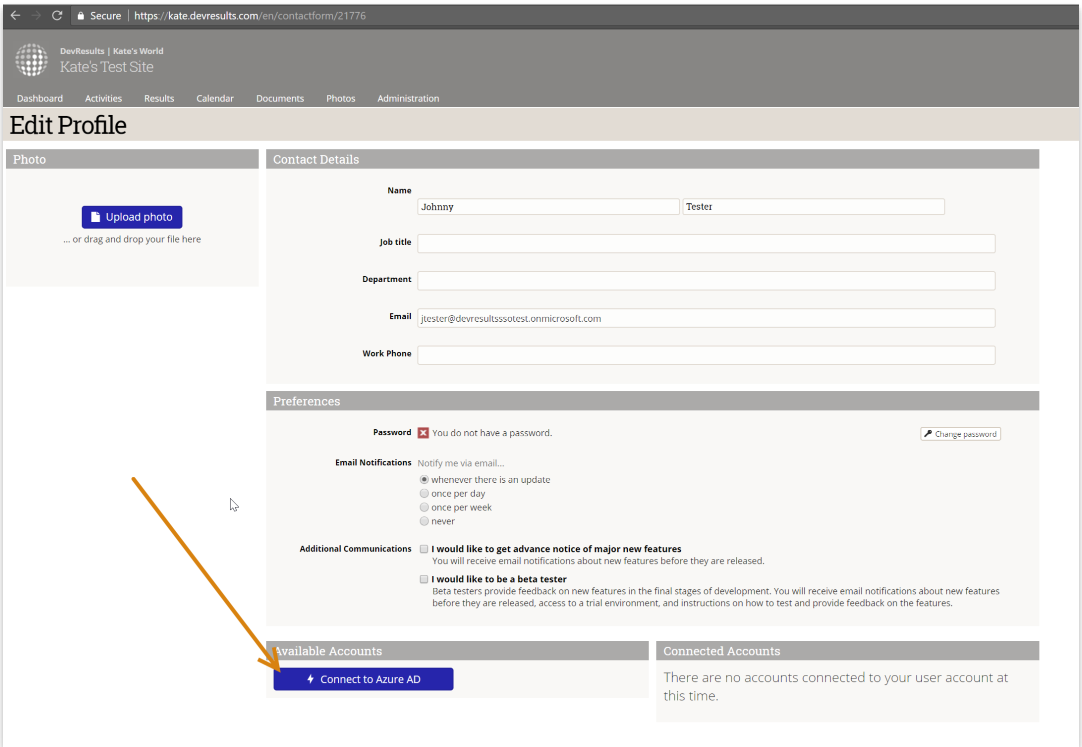
If you're a site administrator and you'd like to be able to see a list of users and whether they're using a connected SSO account or not, let us know--we'll be happy to **create a custom query** that meets your needs.

I already had a DevResults user account before we added SSO. Can I link those accounts somehow?

Individual users can link their DevResults account to an Active Directory account, provided the email addressees are the same.

Click on your profile picture in the upper right and select **Edit Profile**.

If you don't currently have an Active Directory account linked to your DevResults account, you'll have nothing in the **Connected Accounts** section and the **Available Accounts** section will have a **Connect to {Name}** button.



Edit Profile

Photo

Contact Details

Name: Johnny | Tester

Job title:

Department:

Email: jtester@devresultssotest.onmicrosoft.com

Work Phone:

Preferences

Password ☒ You do not have a password.

Email Notifications Notify me via email...
● whenever there is an update
○ once per day
○ once per week
○ never

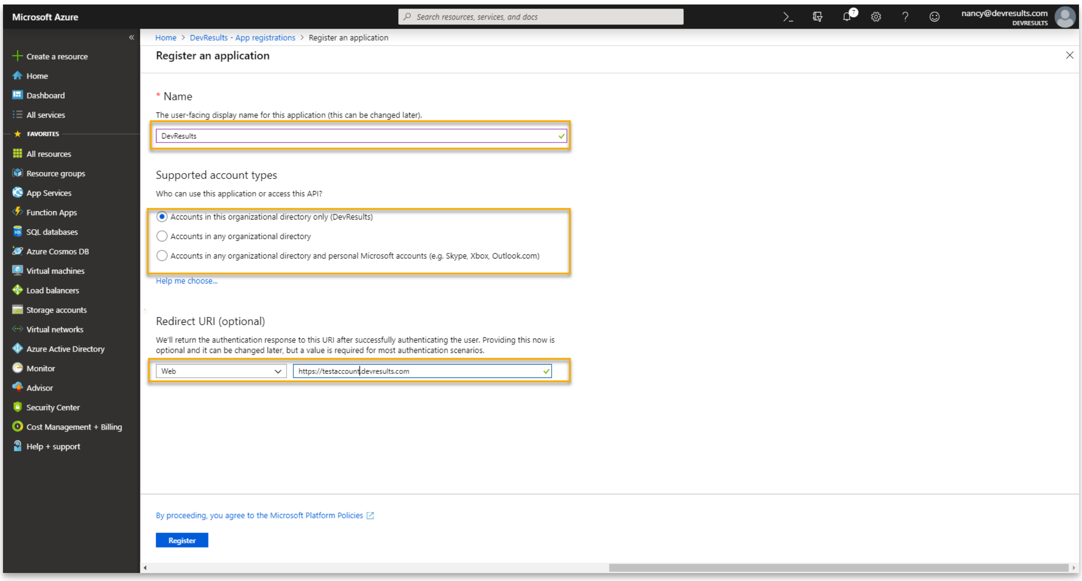
Additional Communications
□ **I would like to get advance notice of major new features**
You will receive email notifications about new features before they are released.
□ **I would like to be a beta tester**
Beta testers provide feedback on new features in the final stages of development. You will receive email notifications about new features before they are released, access to a trial environment, and instructions on how to test and provide feedback on the features.

Available Accounts

Connected Accounts

There are no accounts connected to your user account at this time.

Clicking that button will take you to the Microsoft login screen where you can enter their Active Directory credentials and grant DevResults permission to use them. Once you've done this, moving forward you'll click the **Use my {SSO} account** when you go to log into DevResults.



Register an application

* Name

The user-facing display name for this application (this can be changed later).

DevResults

Supported account types

Who can use this application or access this API?

● Accounts in this organizational directory only (DevResults)
○ Accounts in any organizational directory
○ Accounts in any organizational directory and personal Microsoft accounts (e.g. Skype, Xbox, Outlook.com)

Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web | https://testaccount.devresults.com

By proceeding, you agree to the Microsoft Platform Policies

How do I configure my SSO on the training site after it has been refreshed with data from my live site?

There are a few different options for ensuring that the training site remains properly configured with your SSO after a **training site refresh**:

- In most cases, you only need to provide your SSO application with two "Sign-in redirect URIs," one that has <your-site>.devresults.com as the host and one that has <your-site>.training-devresults.com as the host. The protocol and path should be the same for both.
- If you'd like to have different groups of people have access to training and live sites, use a separate SSO provider/application, or just have different settings between training and production, the solution is a bit more complicated. One solution would be to use the DevResults API to re-create the login settings on the training site after the refresh completes. The general idea is as follows:
  - Create an API Key with Owner permissions on the production site (this way it is always present in training after the refresh)
  - After the training refresh, use the API key to delete the live site login and re-create the training site login
  - On the SSO provider side, you can either use a wildcard in the "Sign-in redirect URI" to handle the fact that the number path may change over time, or you can update the SSO app with the correct URI after creating the settings on the DevResults side.

back to top

Didn't answer your question? Please email us at**help@devresults.com**.

Related Articles