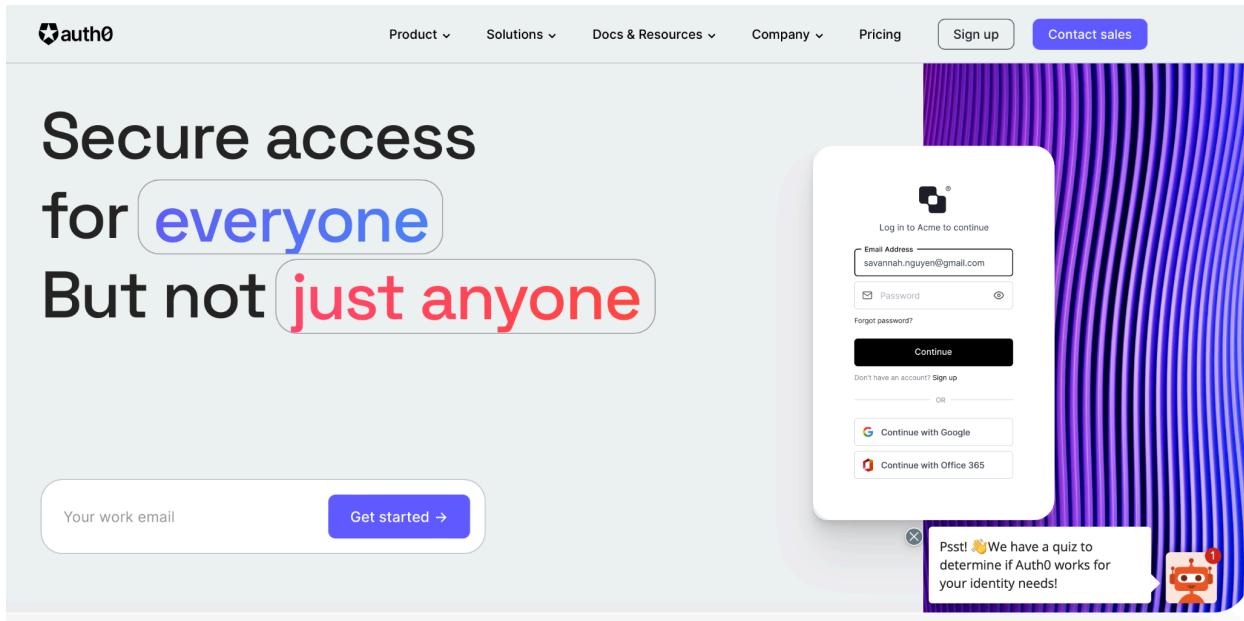


# Auth0 Configuration

Create an account at <https://auth0.com>



## Create a Tenant

The screenshot shows a user interface for managing tenants. At the top, there is a dark header bar with a logo on the left and a dropdown menu labeled "neu Development". Below this is a navigation sidebar on the left with various icons and labels: "Getting started", "Activity feed", "Applications", "Authentication", "Organization", "Users", "Branches", "Security", and "Actions". The "Create tenant" option under "Users" is highlighted with a light gray background and a plus sign icon. A tooltip-like overlay appears over this button, containing the text "+ Create tenant". To the right of the sidebar, the main content area displays a "neu" tenant entry for "US-3 Development" with an American flag icon. There are also links for "Settings", "Invite a member", "Run Readiness Checks", and "Quota Utilization". Further down, there is a "Create tenant" button with a right-pointing arrow, and a "Switch tenant" button with a circular arrow icon. The overall theme is modern and professional, typical of enterprise software.

## New Tenant



### Tenant Domain \*

example

.us.auth0.com

The domain name you choose here will be used for the API endpoints of your applications. Once chosen, a domain name cannot be renamed.

### Region \*



Australia



Europe



Japan



US

We can host all of your data in any of these regions. Useful if you want to specify where to have your data hosted for any reason (e.g. EU)

### Environment Tag \*



Development



Staging



Production

Differentiate between development, staging and production environments. Higher rate limits are applied to tenants tagged as Production with a paid subscription. [Learn more](#)

Cancel

Create

# Create Application

The screenshot shows the Auth0 dashboard interface. On the left, there is a sidebar with various navigation items such as Getting Started, Activity, Applications, Authentication, Organizations, User Management, Branding, Security, Actions, Auth Pipeline, Monitoring, Marketplace, Extensions, Settings, Get support, and Give feedback. The 'Getting Started' section is currently active. The main content area is titled 'Create Application' and contains four cards:

- Try your Login box**: A card showing a preview of a login form with fields for email and password. It includes links to 'Try it out' and 'Customize'.
- Invite your team members**: A card with a user icon, explaining how to add additional admins. It includes a link to 'Learn more about Tenant Administrator permissions' and a 'Create Application' button.
- Integrate Auth0 into your application**: A card with a gear icon, explaining how to add Auth0 to an application. It includes a link to 'Learn more about How Auth0 works' and a 'Create Application' button. A red arrow points to this button.
- Add a social login provider**: A card with a person icon, explaining how to give users social login options. It includes a link to 'Add Social Connections'.

The top right of the dashboard features a search bar, a 'Discuss your needs' button, a 'Docs' button, a notification bell, and a user profile icon.

## Create Application



Name \*

TodoApp

You can change the application name later in the application settings.

Choose an application type



Native

Mobile, desktop,  
CLI and smart  
device apps  
running natively.

e.g.: iOS,  
Electron, Apple  
TV apps



Single Page Web  
Applications

A JavaScript  
front-end app  
that uses an API.

e.g.: Angular,  
React, Vue



Regular Web  
Applications

Traditional web  
app using  
redirects.

e.g.: Node.js  
Express,  
ASP.NET, Java,  
Rails



Machine to  
Machine  
Applications

CLIs, daemons or  
services running  
on your backend.

e.g.: Shell script

Cancel

Create

The screenshot shows the Auth0 application dashboard for the 'TodoApp' project. At the top, there's a navigation bar with tabs: 'Quick Start' (which is active and highlighted in blue), 'Settings', 'Addons', 'Connections', and 'Organizations'. Below the navigation, there's a section titled 'Single-Page App' with the heading 'What technology are you using for your web app?'. This section features a search bar labeled 'Search by technology name' and four circular icons representing different technologies: Angular (blue circle with 'A'), JavaScript (yellow circle with 'JS'), React (dark gray circle with a blue atom icon), and Vue (light green circle with a green 'V'). A red arrow points to the React icon. The React icon is also highlighted with a red border.

- Go back to Applications, we are going to delete the generic App that is created by default

## Applications

+ Create Application

Setup a mobile, web or IoT application to use Auth0 for Authentication. [Learn more →](#)

The screenshot shows the 'Applications' list on the Auth0 dashboard. It lists two applications: 'Default App' (Generic) and 'TodoApp' (Single Page Application). For each application, there is a small icon, the application name, its type, and its Client ID. The 'TodoApp' entry has a Client ID of 'ohcQaQqxY4EFypq2Wyzjtbx0a1KVIqx1'. To the right of the application list, a context menu is open over the 'TodoApp' entry. The menu items are: 'Quickstart', 'Settings' (which is currently selected and highlighted in dark gray), 'Addons', 'Connections', 'Organizations', and 'Run Readiness checks'. The 'Settings' item is the active option.

Sets the inactivity lifetime of a `refresh_token` (in seconds).

Advanced Settings

Save Changes

**Danger Zone**

**Delete this application**  
All your apps using this client will stop working.

**Rotate secret**  
All authorized apps will need to be updated with the new client secret.

**Delete** **Rotate**



- Now let's go back to our TodosApp and configure the following Application URIs

**Getting Started**

- Activity** **FIRST**
- Applications** **▼**
  - Applications
  - APIs
  - SSO Integrations
- Authentication** **>**
- Organizations**
- User Management** **>**
  - Branding
  - Security
  - Actions
  - Auth Pipeline
  - Monitoring
- Marketplace
- Extensions
- Settings

Get support  
 Give feedback

### Application URIs

**Application Login URI**  
  
In some scenarios, Auth0 will need to redirect to your application's login page. This URI needs to point to a route in your application that should redirect to your tenant's `/authorize` endpoint. [Learn more](#)

**Allowed Callback URLs**

After the user authenticates we will only call back to any of these URLs. You can specify multiple valid URLs by comma-separating them (typically to handle different environments like QA or testing). Make sure to specify the protocol (`https://`) otherwise the callback may fail in some cases. With the exception of custom URI schemes for native clients, all callbacks should use protocol `https://`. You can use [Organization URL](#) parameters in these URLs.

**Allowed Logout URLs**

A set of URLs that are valid to redirect to after logout from Auth0. After a user logs out from Auth0 you can redirect them with the `returnTo` query parameter. The URL that you use in `returnTo` must be listed here. You can specify multiple valid URLs by comma-separating them. You can use the star symbol as a wildcard for subdomains (`*.google.com`). Query strings and hash information are not taken into account when validating these URLs. Read more about this at [https://auth0.com/docs/authenticate/login/logout](#)

**Allowed Web Origins**

Comma-separated list of allowed origins for use with [Cross-Origin Authentication](#), [Device Flow](#), and [web message response mode](#), in the form of `<scheme> ":" <host> [ ":" <port> ]`, such as `https://login.mydomain.com` or `http://localhost:3000`. You can use wildcards at the subdomain level (e.g.: `https://*.contoso.com`). Query strings and hash information are not taken into account when validating these URLs.

**Allowed Origins (CORS)**

#### Allowed Web Origins

http://localhost:3000

Comma-separated list of allowed origins for use with [Cross-Origin Authentication](#), [Device Flow](#), and [web message response mode](#), in the form of <scheme> "://" <host> [ ":" <port> ], such as https://login.mydomain.com or http://localhost:3000. You can use wildcards at the subdomain level (e.g.: https://\*.contoso.com). Query strings and hash information are not taken into account when validating these URLs.

#### Allowed Origins (CORS)

http://localhost:3000

Allowed Origins are URLs that will be allowed to make requests from JavaScript to Auth0 API (typically used with CORS). By default, all your callback URLs will be allowed. This field allows you to enter other origins if you need to. You can specify multiple valid URLs by comma-separating them or one by line, and also use wildcards at the subdomain level (e.g.: https://\*.contoso.com). Query strings and hash information are not taken into account when validating these URLs.. You can use [Organization URL](#) placeholders in these URLs.

- Scroll down and save those changes

## Create a new user.

- Go to User Management and Create a new User

⚡ Getting Started

📈 Activity FIRST

💻 Applications >

🔒 Authentication >

🌐 Organizations >

>User Management >

- Users
- Roles

✍️ Branding >

🛡️ Security >

🔗 Actions >

🔧 Auth Pipeline >

📊 Monitoring >

🏪 Marketplace >

🧩 Extensions >

⚙️ Settings >

ⓘ Get support

ⓘ Give feedback

## Users

An easy to use UI to help administrators manage user identities including password resets, creating and provisioning, blocking and deleting users.  
[Learn more →](#)

You don't have any users yet.

All of your users will be found here, regardless of the authentication method they use to access your applications.

+ Create User

[Learn More](#)

## Create user



Email \*

cristian.penarrieta@gmail.com

Password \*

.....

Repeat Password \*

.....

Connection \*

Username-Password-Authentication

Cancel

Create

- Verify your user by confirming the email

[← Back to Users](#)



cristian.penarrieta@gmail.com

user\_id: auth0|6236b438cd1229006a1e65a6

Actions ▾

Details Devices History Raw JSON Authorized Applications Permissions Roles

Name cristian.penarrieta@gmail.com <a href="#">Edit</a>	Email cristian.penarrieta@gmail.com (pending) <a href="#">Edit</a>	Signed Up March 19th 2022, 9:57:28 PM
Primary Identity Provider Database	Latest Login Never	Accounts Associated None
Browser -		

Multi-Factor Authentication

## Create an API in Auth0

- Getting Started
- Activity FIRST
- Applications
- APIs
- SSO Integrations
- Authentication
- Organizations
- User Management
- Branding
- Security
- Actions
- Auth Pipeline
- Monitoring
- Marketplace
- Extensions
- Settings
- [Get support](#)

Thank you for signing up for Auth0! You have 22 days left in your trial to experiment with features that are not in the Free plan. Like what you're seeing? Please enter your [billing information here](#).

[View Plans](#)

### APIs

Define APIs that you can consume from your authorized applications.

 <b>Auth0 Management API</b> System API	API Audience: <a href="https://neu.us.auth0.com/api/v2/">https://neu.us.auth0.com/api/v2/</a> <a href="#">Edit</a>	...
---	--	-----

[+ Create API](#)

## New API

**Name \***

A friendly name for the API.

**Identifier \***

A logical identifier for this API. We recommend using a URL but note that this doesn't have to be a publicly available URL, Auth0 will not call your API at all. **This field cannot be modified.**

**Signing Algorithm \***

Algorithm to sign the tokens with. When selecting RS256 the token will be signed with Auth0's private key.

**Cancel** **Create**

- Go to the permissions tab and add the following permissions

## List of Permissions (Scopes)

These are all the permissions (scopes) that this API uses.

Permission	Description	
read:todoitem	read todo item	
read:user	read user data	
edit:todoitem	edit todoitem	
edit:user	edit user data	
delete:todoitem	delete todo item	
delete:user	delete user	
write:user	Create new user	
write:todoitem	create new todo item	

## Create a Role

- Getting Started
- Activity **FIRST**
- Applications >
- Authentication >
- Organizations >
- User Management **SECOND**
  - Users
  - Roles
- Branding >
- Security >
- Actions >
- Auth Pipeline >
- Monitoring >
- Marketplace >
- Extensions >
- Settings >
- Get support

### Roles

Create and manage Roles for your applications. Roles contain collections of Permissions and can be assigned to Users.



You don't have any roles yet.

Create roles to represent the types of users that access your applications. Assign permissions to those roles to control what users are allowed to do in your apps.

[+ Create Role](#)

## New Role

X

Name \*

Admin

Description \*

Full Access

Cancel

Create

- Go to the Permissions tab and assign all the permissions into that role
- Click Add Permissions

← Back to Roles

## Admin

Role ID: `rol_nsJqjmV4e41HQh70`

Settings    **Permissions**    Users

Add Permissions to this Role. Users who have this Role will receive all Permissions below that match the API of their login request.

Add Permissions

Permission ^	Description	API
No Permissions assigned		

## Add Permissions

X

Select permissions from existing APIs

Select an API...

TodosAPI

<https://api.todos>

## Add Permissions

X

Select permissions from existing APIs



Warning! This API does not have RBAC enabled. Roles and permissions will not be evaluated during the authorization transaction. You can enable this on the [API settings page](#) under Access Settings.

TodosAPI



### Permissions

Select: [All](#)

[None](#)

Filter Permissions

read:todoitem

read:user

edit:todoitem

edit:user

delete:todoitem

delete:user

write:user

write:todoitem

**Add Permissions**

### Enable RBAC to our API

- go back to the settings tab of our API and Enable RBAC

Getting Started

- Activity FIRST**
- Applications
- APIs
- SSO Integrations
- Authentication
- Organizations
- User Management
- Branding
- Security
- Actions
- Auth Pipeline
- Monitoring
- Marketplace
- Extensions
- Settings

Token Expiration for Browser Flows (Seconds) ▾

Expiration value (in seconds) for `access tokens` issued for this API via Implicit or Hybrid Flows. Cannot be greater than the Token Lifetime value.

Signing Algorithm

Algorithm to be used when signing the `access tokens` for this API. You can find more information [in this document](#).

---

RBAC Settings

Enable RBAC

If this setting is enabled, RBAC authorization policies will be enforced for this API. Role and permission assignments will be evaluated during the login transaction.

Add Permissions in the Access Token

If this setting is enabled, the Permissions claim will be added to the access token. Only available if RBAC is enabled for this API.

---

Access Settings

Allow Skipping User Consent

- Click Save

## Universal login

- Go to Branding -> Universal login, Make sure that the New experience is selected

Getting Started

- Activity FIRST**
- Applications
- Authentication
- Organizations
- User Management
- Branding
- Universal Login
- Custom Domains
- Email Templates
- Email Provider
- Security
- Actions
- Auth Pipeline
- Monitoring
- Marketplace
- Extensions
- Settings

## Universal Login

Create a beautiful universal login page where you can redirect to authenticate your users. [Learn more →](#)

Settings    Login    Password Reset    Multi-factor Authentication

**Experience**

The default look and feel for your Universal Login pages. Please note that the **New Universal Login Experience** is not yet at feature parity with the **Classic Experience**. [Learn more](#)

**New**

New and improved visuals, flows, and functionality enhancing your end user experience. [Learn More](#)

- Lightweight and faster
- No JavaScript required



**Classic**

Our existing experience with the look and feel you and your users are familiar with. [Learn More](#)



## Create a Custom Rule

The screenshot shows the Neu Platform's Library page. At the top right, there are links for 'Search', 'Discuss your needs', 'Documentation', and a user profile. Below the header, the left sidebar has sections like 'Getting Started', 'Activity', 'Applications', 'Authentication', 'Organizations', 'User Management', 'Branding', 'Security', and 'Actions'. Under 'Actions', there are 'Flows' and 'Library'. A red arrow points from the left towards 'Library'. At the top right of the main content area, there are two buttons: 'Build Custom' and '+ Add Action'. Another red arrow points from the right towards the '+ Add Action' button.

The screenshot shows the 'Create Action' dialog box. It has three main sections: 'Name \*' with a input field containing 'Augment User'; 'Trigger \*' with a dropdown menu showing 'Login / Post Login'; and 'Runtime \*' with a dropdown menu showing 'Node 18 (Recommended)'. At the bottom are 'Cancel' and 'Create' buttons. The 'Runtime' section is highlighted with a blue border.

Name *
Augment User

Trigger *
Login / Post Login

Runtime *
Node 18 (Recommended)

Cancel Create

The screenshot shows the Neu Platform's development environment. On the left, a sidebar navigation includes 'Getting Started', 'Activity', 'Applications', 'Authentication', 'Organizations', 'User Management', 'Branding', 'Security', 'Actions' (selected), 'Flows', 'Library', 'Auth Pipeline', 'Monitoring', 'Marketplace', 'Extensions', and 'Settings'. Below this are links for 'Get support' and 'Give feedback'. The main content area is titled 'Custom Actions' and shows a specific action named 'Augment User'. The title bar includes a back arrow, a 'Version History' button, a 'Save Draft' button, and a 'Deploy' button. A message indicates 'You have unsaved changes'. The code editor displays the following JavaScript code:

```

1 /**
2  * Handler that will be called during the execution of a PostLogin flow.
3 *
4  * @param {Event} event - Details about the user and the context in which they are logging in.
5  * @param {PostLoginAPI} api - Interface whose methods can be used to change the behavior of the login.
6 */
7 exports.onExecutePostLogin = async (event, api) => {
8   const namespace = 'https://api.todos/';
9
10  api.accessToken.setCustomClaim(namespace + 'email', event.user.email);
11  api.accessToken.setCustomClaim(namespace + 'name', event.user.name);
12
13  if (event.authorization) {
14    api.idToken.setCustomClaim(namespace + 'roles', event.authorization.roles);
15  }
16};
17
18
19
20 /**
21 * Handler that will be invoked when this action is resuming after an external redirect. If your
22 * onExecutePostLogin function does not perform a redirect, this function can be safely ignored.

```

A 'View Samples' button is located at the bottom right of the code editor.

```

exports.onExecutePostLogin = async (event, api) => {
  const namespace = 'https://api.todos/';

  api.accessToken.setCustomClaim(namespace + 'email', event.user.email);
  api.accessToken.setCustomClaim(namespace + 'name', event.user.name);

  if (event.authorization) {
    api.idToken.setCustomClaim(namespace + 'roles', event.authorization.roles);
  }
};

```

## Library

Manage your actions and configuration.

[Build Custom](#) [+ Add Action](#)

Installed	Custom										
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left; padding: 5px;">Name ^</th> <th style="text-align: left; padding: 5px;">Trigger</th> <th style="text-align: left; padding: 5px;">Last Update</th> <th style="text-align: left; padding: 5px;">Last Deploy</th> <th style="text-align: left; padding: 5px;">Status</th> </tr> </thead> <tbody> <tr> <td style="padding: 5px;"><a href="#"> Augment User</a></td> <td style="padding: 5px;">Login / Post Login</td> <td style="padding: 5px;">2 minutes ago</td> <td style="padding: 5px;">a few seconds ago</td> <td style="padding: 5px; text-align: center;"><span style="border: 1px solid green; padding: 2px 10px; border-radius: 5px;">DEPLOYED</span></td> </tr> </tbody> </table>		Name ^	Trigger	Last Update	Last Deploy	Status	<a href="#"> Augment User</a>	Login / Post Login	2 minutes ago	a few seconds ago	<span style="border: 1px solid green; padding: 2px 10px; border-radius: 5px;">DEPLOYED</span>
Name ^	Trigger	Last Update	Last Deploy	Status							
<a href="#"> Augment User</a>	Login / Post Login	2 minutes ago	a few seconds ago	<span style="border: 1px solid green; padding: 2px 10px; border-radius: 5px;">DEPLOYED</span>							
1 actions											

neu | DEVELOPMENT

Search Discuss your needs Documentation

Getting Started

- Activity
- Applications
- Authentication
- Organizations
- User Management
- Branding
- Security
- Actions
  - Flows
  - Library
- Auth Pipeline
- Monitoring
- Marketplace
- Extensions
- Settings

Get support Give feedback

Choose flow

## Login

Customize the login flow for your applications.

All changes are live

Add Action

Installed Custom

Search actions...

Augment User

Start User Logged In

Complete Token Issued

?

```
graph TD; Start((Start  
User Logged In)) --> AugmentUser[Augment User]; AugmentUser --> Complete((Complete  
Token Issued))
```