# react-google-login

5.2.2 • Public • Published a year ago

Readme

Explore BETA

2 Dependencies

192 Dependents

94 Versions

# React Google Login

> A Google oAUth Sign-in / Log-in Component for React

## Storybook

Demo Link

## Install

```
npm install react-google-login
```

## How to use

```
import React from 'react';
import ReactDOM from 'react-dom';
import GoogleLogin from 'react-google-login';
// or
import { GoogleLogin } from 'react-google-login';


const responseGoogle = (response) => {
  console.log(response);
}

ReactDOM.render(
  <GoogleLogin
    clientId="658977310896-knrl3gka66fldh83dao2rhgbblmd4un9.apps.googl
    buttonText="Login"
    onSuccess={responseGoogle}
    onFailure={responseGoogle}
    cookiePolicy={'single_host_origin'}
  />,
  document.getElementById('googleButton')
);
```

## Google button without styling or custom button

```
ReactDOM.render(
  <GoogleLogin
    clientId="658977310896-knrl3gka66fldh83dao2rhgbblmd4un9.apps.googl
    render={renderProps => (
      <button onClick={renderProps.onClick} disabled={renderProps.disa
    )}
    buttonText="Login"
    onSuccess={responseGoogle}
```

```
    onFailure={responseGoogle}
    cookiePolicy={'single_host_origin'}
  />,
  document.getElementById('googleButton')
);
```

## Stay Logged in

`isSignedIn={true}` attribute will call `onSuccess` callback on load to keep the user signed in.

```
<GoogleLogin
  clientId="658977310896-knrl3gka66fldh83dao2rhgbblmd4un9.apps.googleu
  onSuccess={responseGoogle}
  isSignedIn={true}
/>
```

## Login Hook

```
import { useGoogleLogin } from 'react-google-login'

const { signIn, loaded } = useGoogleLogin({
    onSuccess,
    onAutoLoadFinished,
    clientId,
    cookiePolicy,
    loginHint,
    hostedDomain,
    autoLoad,
    isSignedIn,
    fetchBasicProfile,
```

```
    redirectUri,
    discoveryDocs,
    onFailure,
    uxMode,
    scope,
    accessType,
    responseType,
    jsSrc,
    onRequest,
    prompt
  })
```

## Logout Hook

```
import { useGoogleLogout } from 'react-google-login'

const { signOut, loaded } = useGoogleLogout({
    jsSrc,
    onFailure,
    clientId,
    cookiePolicy,
    loginHint,
    hostedDomain,
    fetchBasicProfile,
    discoveryDocs,
    uxMode,
    redirectUri,
    scope,
    accessType,
    onLogoutSuccess
  })
```

## onSuccess callback

If responseType is not 'code', callback will return the GoogleAuth object.

If responseType is 'code', callback will return the authorization code that can be used to retrieve a refresh token from the server.

If you use the hostedDomain param, make sure to validate the id_token (a JSON web token) returned by Google on your backend server:

1. In the `responseGoogle(response) {...}` callback function, you should get back a standard JWT located at `response.tokenId` or `res.getAuthResponse().id_token`
2. Send this token to your server (preferably as an `Authorization` header)
3. Have your server decode the id_token by using a common JWT library such as **jwt-simple** or by sending a GET request to `https://www.googleapis.com/oauth2/v3/tokeninfo?id_token=YOUR_TOKEN_HERE`
4. The returned decoded token should have an `hd` key equal to the hosted domain you'd like to restrict to.

## Logout

Use GoogleLogout button to logout the user from google.

```
import { GoogleLogout } from 'react-google-login';
<GoogleLogout
  clientId="658977310896-knrl3gka66fldh83dao2rhgbblmd4un9.apps.goc
  buttonText="Logout"
  onLogoutSuccess={logout}
>
</GoogleLogout>
```

## Login Props

| params | value | default value |
|---|---|---|
| clientId | string | REQUIRED |
| jsSrc | string | https://apis.google.com/js/api.js |
| hostedDomain | string | - |
| scope | string | profile email |
| responseType | string | permission |
| accessType | string | online |
| onSuccess | function | REQUIRED |
| onFailure | function | REQUIRED |
| onScriptLoadFailure | function | - |
| onRequest | function | - |
| onAutoLoadFinished | function | - |
| buttonText | string | Login with Google |

| params | value | default value |
| --- | --- | --- |
| className | string | - |
| style | object | - |
| disabledStyle | object | - |
| loginHint | string | - |
| prompt | string | - |
| tag | string | button |
| type | string | button |
| autoLoad | boolean | false |
| fetchBasicProfile | boolean | true |
| disabled | boolean | false |
| discoveryDocs | - | https://developers.google.com/discovery/v1/us |
| uxMode | string | popup |
| theme | string | light |
| icon | boolean | true |

| params | value | default value |
| --- | --- | --- |
| redirectUri | string | - |
| isSignedIn | boolean | false |
| render | function | - |
| Google Scopes List: scopes | | |

## Logout Props

| params | value | default value |
| --- | --- | --- |
| clientId | string | REQUIRED |
| jsSrc | string | https://apis.google.com/js/api.js |
| hostedDomain | string | - |
| scope | string | profile email |
| accessType | string | online |

| params | value | default value |
| --- | --- | --- |
| onLogoutSuccess | function | REQUIRED |
| onFailure | function | REQUIRED |
| onScriptLoadFailure | function | - |
| buttonText | string | Logout of Google |
| className | string | - |
| disabledStyle | object | - |
| loginHint | string | - |
| tag | string | button |
| type | string | button |
| fetchBasicProfile | boolean | true |
| disabled | boolean | false |
| discoveryDocs | - | https://developers.google.com/discovery/v1/usi |
| uxMode | string | popup |
| theme | string | light |
| icon | boolean | true |

| params | value | default value |
| --- | --- | --- |
| redirectUri | string | - |
| isSignedIn | boolean | false |
| render | function | - |
| Google Scopes List: scopes | | |

## onSuccess callback ( w/ offline false)

onSuccess callback returns a GoogleUser object which provides access to all of the GoogleUser methods listed here: **https://developers.google.com/identity/sign-in/web/reference#users** .

You can also access the returned values via the following properties on the returned object.

| property name | value | definition |
| --- | --- | --- |
| googleId | string | Google user ID |
| tokenId | string | Token Id |
| accessToken | string | Access Token |
| tokenObj | object | Token details object |
| profileObj | object | Profile details object |

## onSuccess callback ( w/ offline true)

| property name | value | definition |
| --- | --- | --- |
| code | object | offline token |

You can also pass child components such as icons into the button component.

```
<GoogleLogin
  clientId={'658977310896-knrl3gka66fldh83dao2rhgbblmd4un9.apps.goog
  onSuccess={responseGoogle}
  onFailure={responseGoogle}
>
  <FontAwesome
    name='google'
  />
  <span> Login with Google</span>
</GoogleLogin>
```

## onFailure callback

onFailure callback is called when either initialization or a signin attempt fails.

| property name | value | definition |
| --- | --- | --- |
| error | string | Error code |
| details | string | Detailed error description |

Common error codes include:

| error code | description |
| --- | --- |

| error code | description |
| --- | --- |
| `idpiframe_initialization_failed` | initialization of the Google Auth API failed (this will occur if a client doesn't have third party cookies enabled) |
| `popup_closed_by_user` | The user closed the popup before finishing the sign in flow. |
| `access_denied` | The user denied the permission to the scopes required |
| `immediate_failed` | No user could be automatically selected without prompting the consent flow. |

More details can be found in the official Google docs:

- GoogleAuth.then(onInit, onError)
- GoogleAuth.signIn(options)
- GoogleAuth.grantOfflineAccess(options)

## Dev Server

```
npm run start
```

Default dev server runs at localost:8080 in browser. You can set IP and PORT in webpack.config.dev.js

## Run Tests

```
npm run test:watch
```

## Production Bundle

```
npm run bundle
```

## Deploy Storybook

```
npm run deploy-storybook
```

**Checkout my other login:** React Instagram Login

**Checkout keppelen's** React Facebook Login

### Follow me on Twitter: @anthonyjgrove

### Keywords

react reactjs react-component google-login google-oAuth2 google-oAuth

**Install**

```
> npm i react-google-login
```

**Repository**

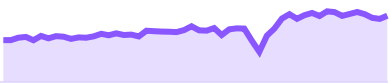
github.com/anthonyjgrove/react-google-login

**Homepage**

github.com/anthonyjgrove/react-google-login

**Weekly Downloads**

240,137



| Version | License |
| --- | --- |
| 5.2.2 | MIT |

| Unpacked Size | Total Files |
| --- | --- |
| 131 kB | 36 |

| Issues | Pull Requests |
| --- | --- |
| 147 | 24 |

**Last publish**

a year ago

**Collaborators**

>_Try on RunKit

Report malware

## Support

Help

Advisories

Status

Contact npm

## Company

About

Blog

Press

## Terms & Policies

Policies

Terms of Use

Code of Conduct

Privacy