

# Serverless Auth Strategies with Netlify Identity

*Divya Sasidharan*



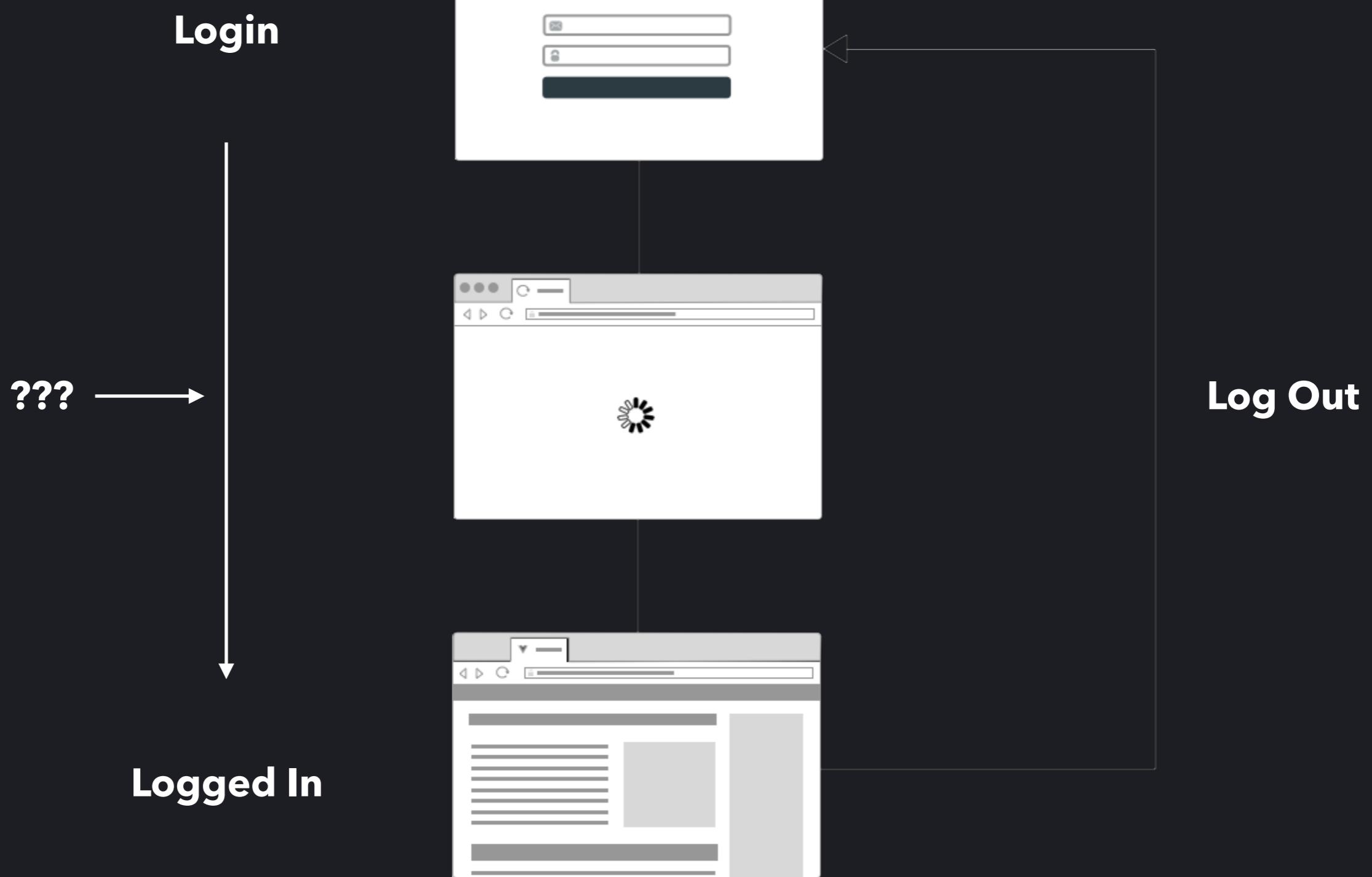
netlify

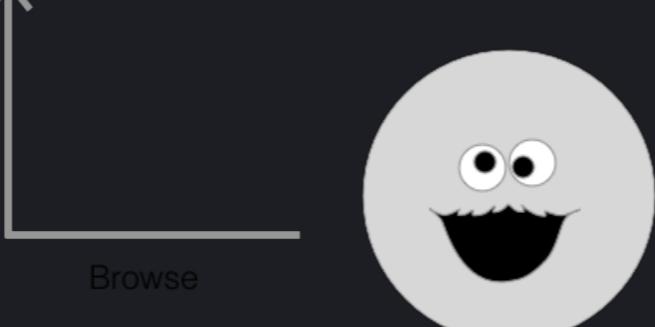
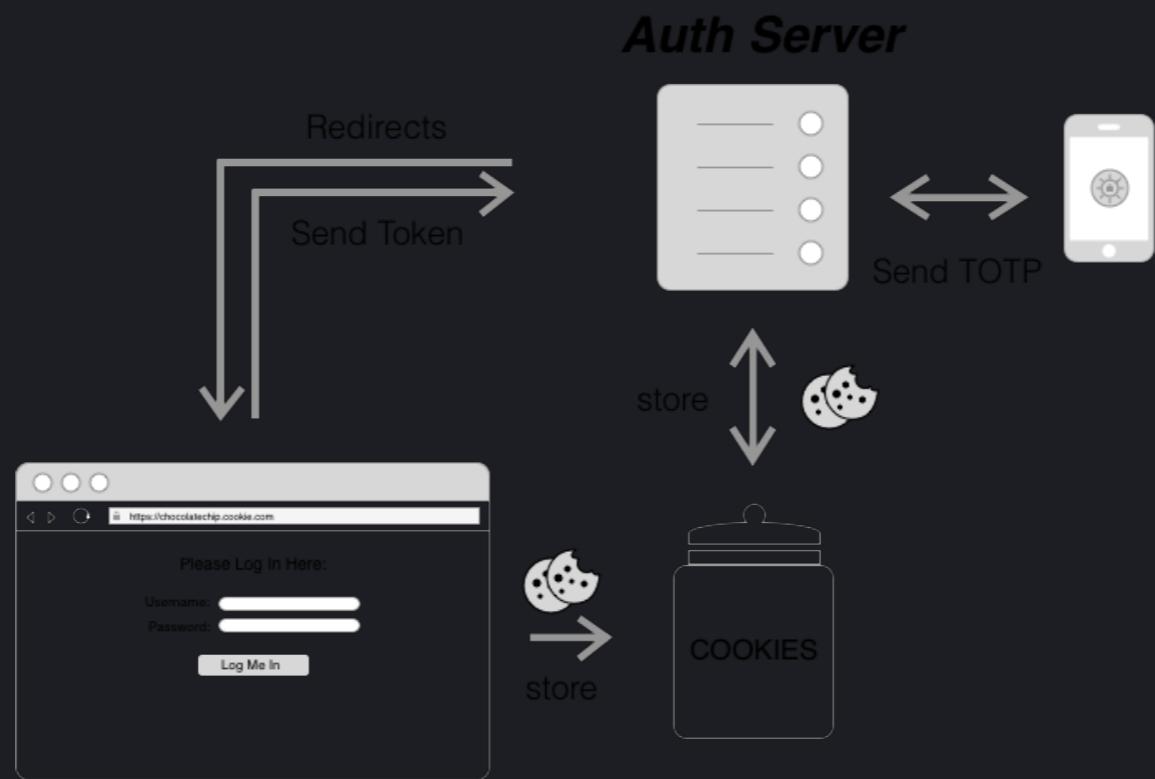
@shortdiv

*Why Authentication?*

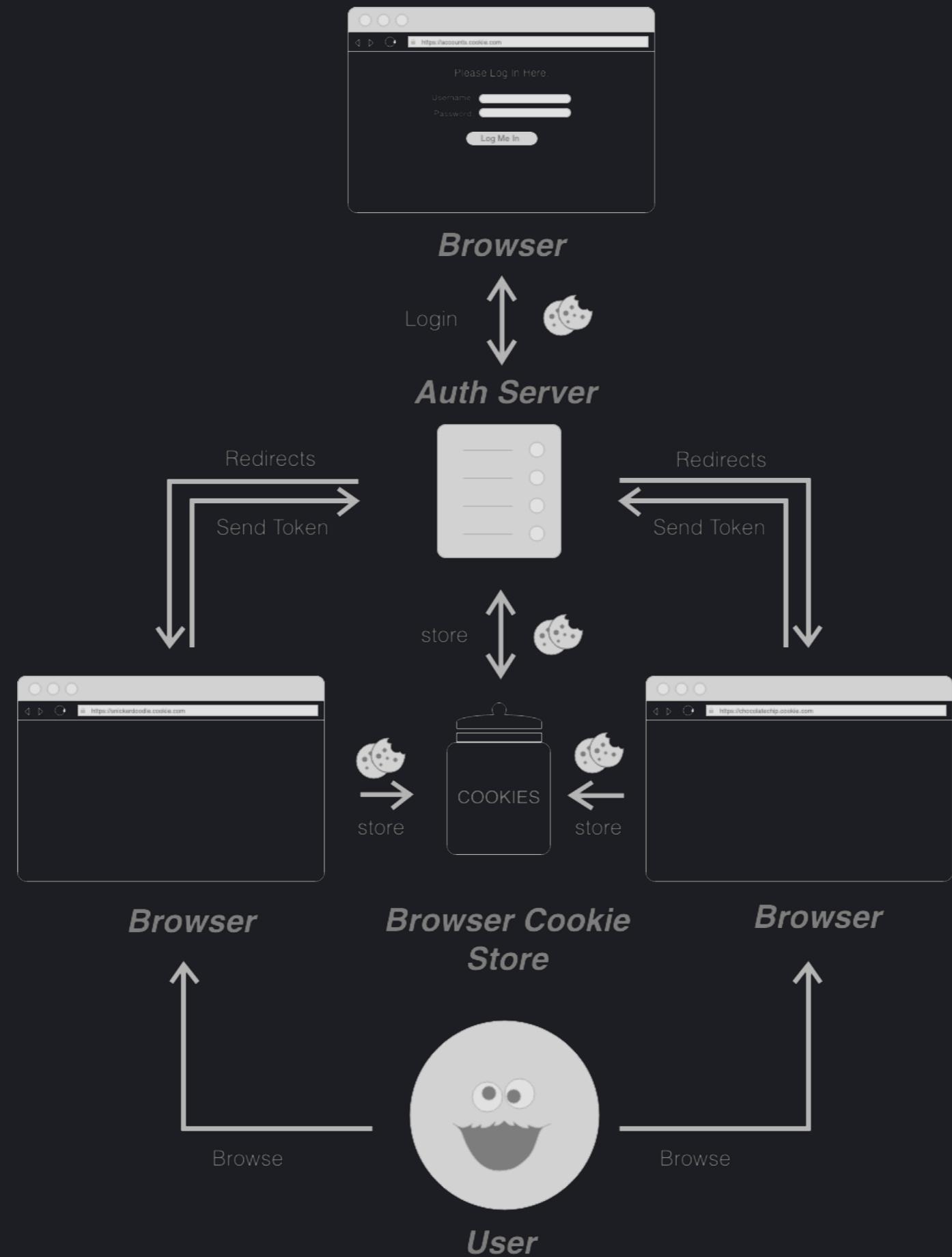


*What is* **Authentication?**





**User**



**SSO**

**MFA**

**Silent Auth**

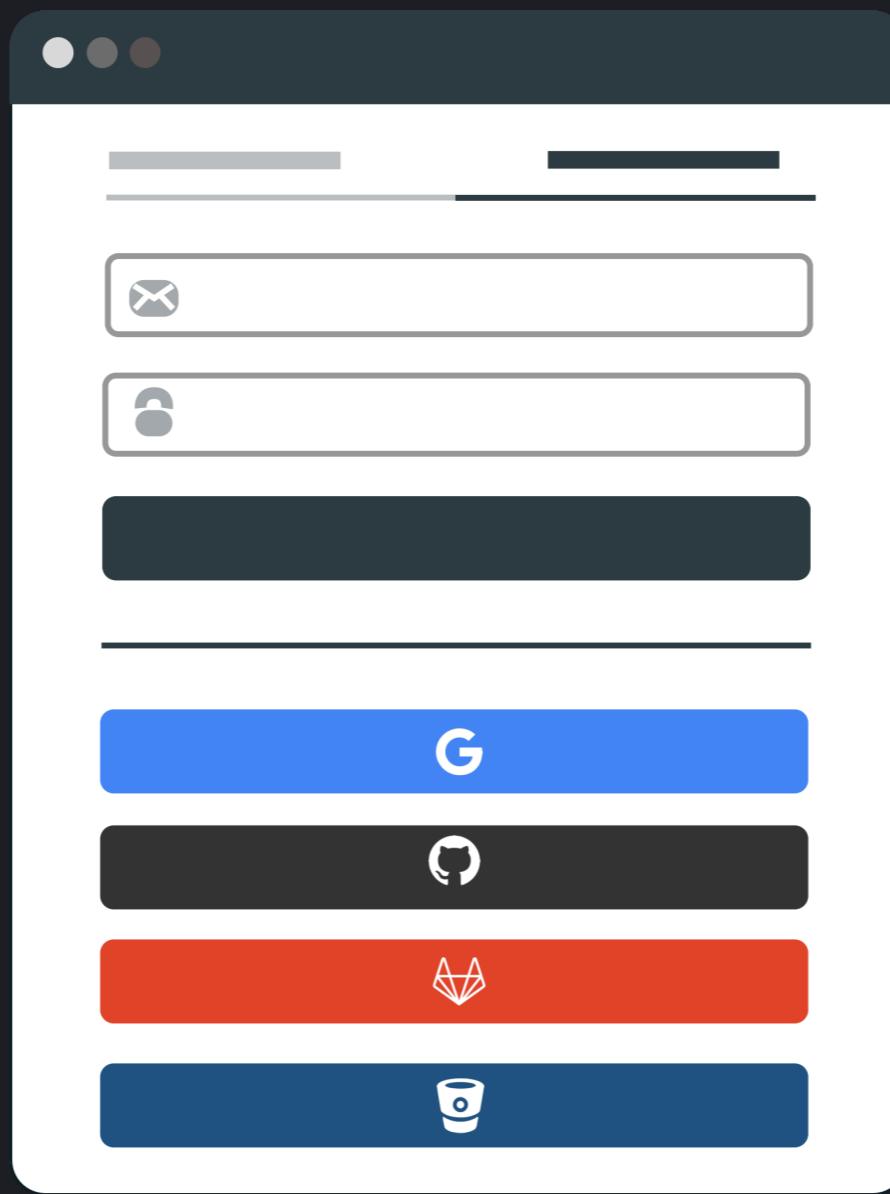
**Social Login**

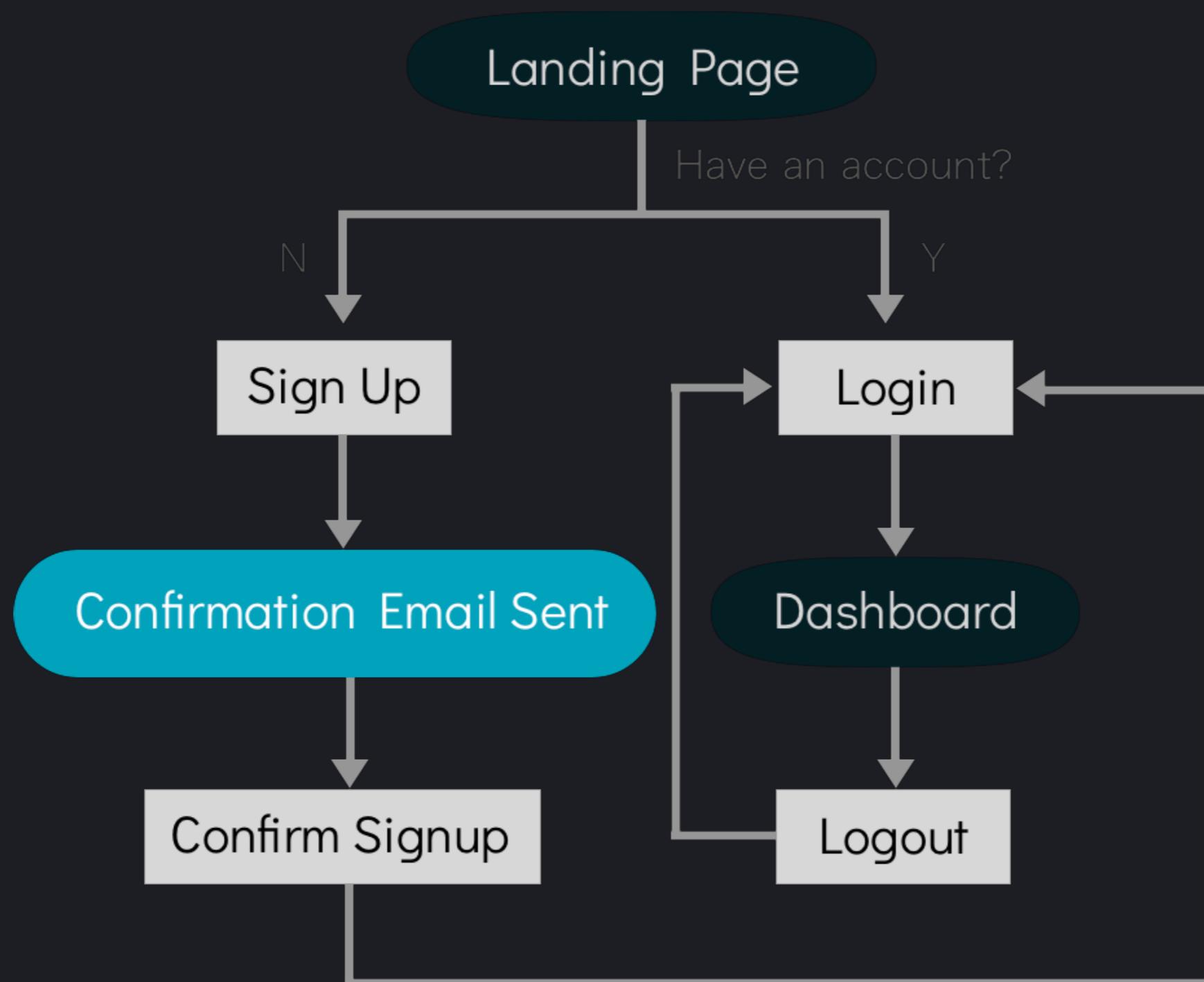
**Passwordless**

<https://www.netlify.com/blog/>

2018/11/28/authentication-for-the-  
rest-of-us/

# Netlify **Identity**







divya › you-auth-to-know

D

Overview Deployes Functions **Identity** Forms Large Media Split Testing Settings

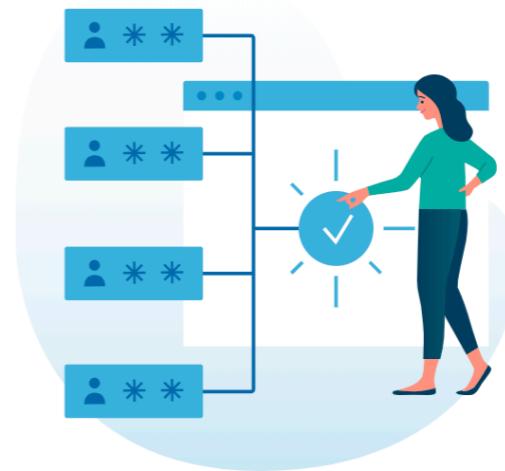
## Identity

Manage signups, logins, password recovery, and more — all without rolling your own authentication service.

Register and authenticate visitors to your site so you can gate content, enable CMS functionality, make authenticated calls to outside services, and more. Securely integrate with any service that understands JSON Web Tokens.

[Learn more in the documentation ↗](#)

[Enable Identity](#)



## Getting started

Check out some of our favorite resources to get started with Identity

[Documentation](#)



[Example code](#)





divya › you-auth-to-know

D

Overview Deployes Functions Identity Forms Large Media Split Testing Settings

## Identity

Identity Free

API endpoint: <https://you-auth-to-know.netlify.com/.netlify/identity>

[Learn more in the documentation ↗](#)

[Settings and usage](#)

[Identity audit log](#)

[Search users](#)

[Invite users](#)

### No users found

Your site has no registered users. To get started, make sure your site code is set up to handle Identity requests by using something like the [Netlify Identity widget](#) or [gotrue-js](#). Then, try inviting yourself as the first user.

[Getting started](#)





divya › you-auth-to-know

D

Overview Deployes Functions Identity Forms Large Media Split Testing Settings

## Identity

Identity Free

API endpoint: <https://you-auth-to-know.netlify.com/.netlify/identity>

[Learn more in the documentation ↗](#)

[Settings and usage](#)

[Identity audit log](#)

[Search users](#)

[Invite users](#)



Divya  
divya@netlify.com

No role set  
Created on Mar 14 (19 days ago)



Div  
hello@shortdiv.com

No role set  
Created on Feb 11 (2 months ago)



```
<script type="text/javascript"  
src="https://identity.netlify.com/v1/  
netlify-identity-widget.js"></script>
```

```
yarn add netlify-identity-widget
```

```
const netlifyIdentity = require('netlify-identity-widget');

netlifyIdentity.open();

const user = netlifyIdentity.currentUser();

netlifyIdentity.on('init', user => {
  console.log('init', user)
});

netlifyIdentity.on('login', user => {
  console.log('login', user)
});

netlifyIdentity.on('logout', user => {
  console.log('logout')
});

netlifyIdentity.close();
```



```
const netlifyIdentity = require('netlify-identity-widget');

netlifyIdentity.open();

const user = netlifyIdentity.currentUser();

netlifyIdentity.on('init', user => {
  console.log('init', user)
});

netlifyIdentity.on('login', user => {
  console.log('login', user)
});

netlifyIdentity.on('logout', user => {
  console.log('logout')
});

netlifyIdentity.close();
```

```
const netlifyIdentity = require('netlify-identity-widget');

netlifyIdentity.open();

const user = netlifyIdentity.currentUser();

netlifyIdentity.on('init', user => {
  console.log('init', user)
});

netlifyIdentity.on('login', user => {
  console.log('login', user)
});

netlifyIdentity.on('logout', user => {
  console.log('logout')
});

netlifyIdentity.close();
```

```
const netlifyIdentity = require('netlify-identity-widget');

const user = netlifyIdentity.currentUser();

netlifyIdentity.on('init', user => {
  console.log('init', user)
});

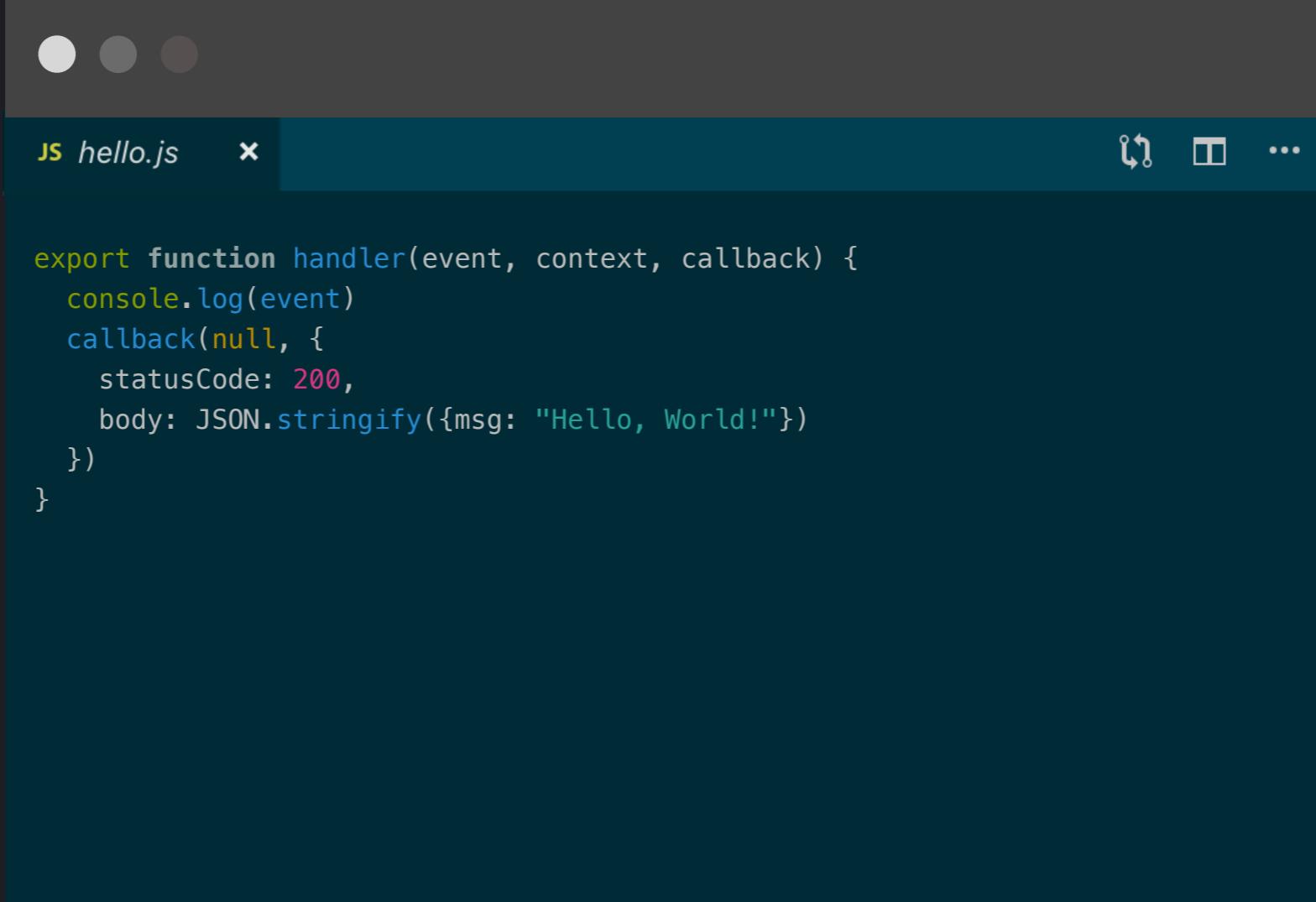
netlifyIdentity.on('login', user => {
  console.log('login', user)
});

netlifyIdentity.on('logout', user => {
  console.log('logout')
});
```

# Demo Time !

**MUCH FUN  
SUCH FUNCTIONS**

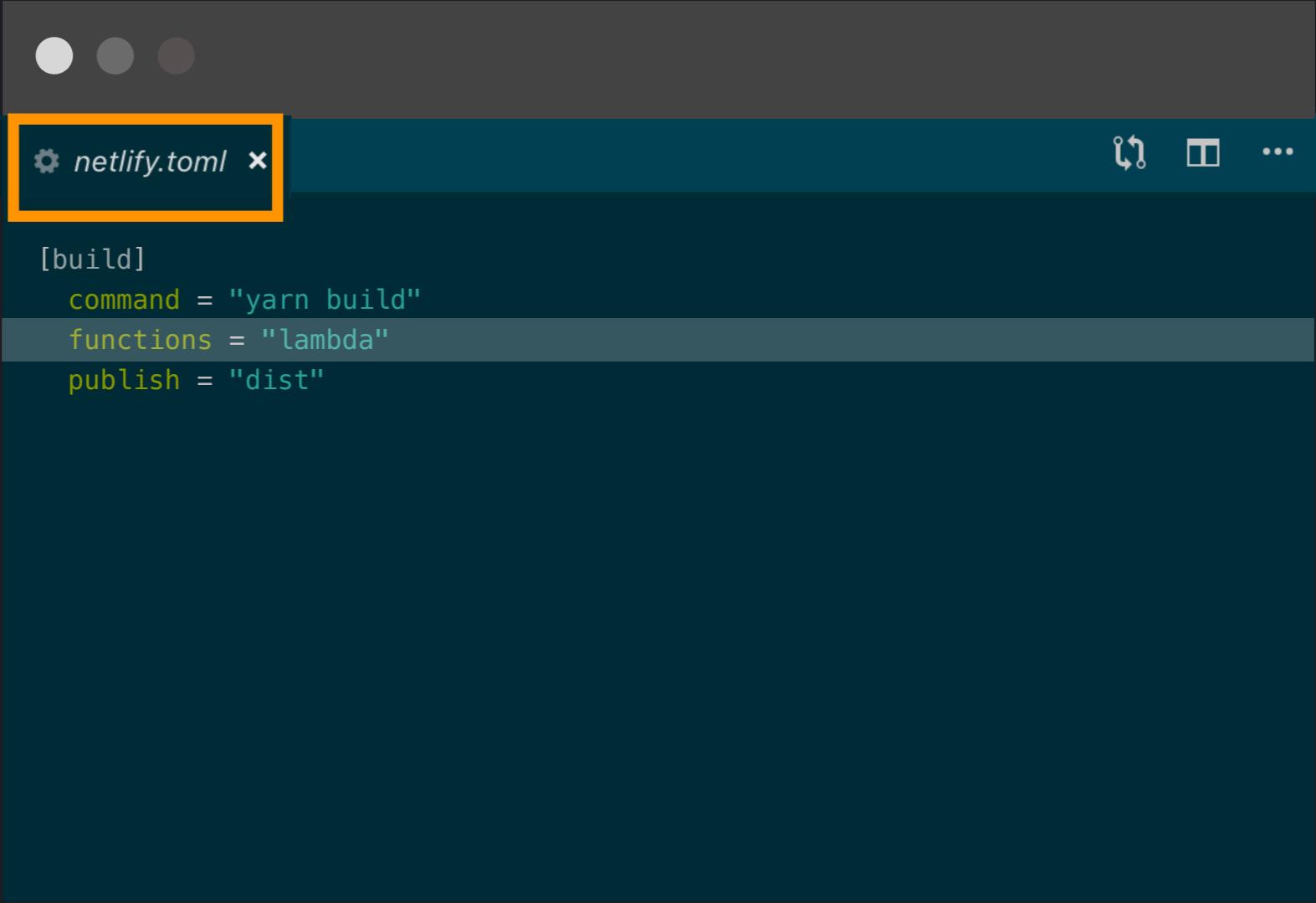


A screenshot of a code editor window titled "JS hello.js". The code editor displays the following JavaScript code:

```
export function handler(event, context, callback) {
  console.log(event)
  callback(null, {
    statusCode: 200,
    body: JSON.stringify({msg: "Hello, World!"})
  })
}
```

The code defines a Lambda function named "handler" that logs the event object to the console and returns a response with a status code of 200 and a body containing the string "Hello, World!".

src/lambda/hello.js



A screenshot of a code editor window showing a file named `netlify.toml`. The file tab is highlighted with a yellow border. The content of the file is as follows:

```
[build]
command = "yarn build"
functions = "lambda"
publish = "dist"
```

# netlify.toml



divya › you-auth-to-know

D

Overview Deploys Functions Identity Forms Large Media Split Testing Settings

## Functions

Functions Free

5 Lambda functions actively running in production.

[Learn more about Functions in the docs ➔](#)

⚙️ [Settings and usage](#)

🔍  Search by branch or Deploy Preview

[handle-signup.js](#)

Created on Mar 14 (19 days ago)



[hello.js](#)

Created on Mar 14 (19 days ago)



[identity-login.js](#)

Created on Mar 14 (19 days ago)



[lock-site.js](#)

Created on Mar 14 (19 days ago)



[super\\_special\\_function.js](#)



`./netlify/functions/{function_name}`

`./netlify/functions/login`



# Custom Webhooks

## API Call



divya's team > you-auth-to-know

D

Overview Deployes Functions Identity Forms Large Media Split Testing Settings

## you-auth-to-know

- <https://you-auth-to-know.netlify.com>

Deploys from GitHub. Last published on Mar 20.

⚙ Site settings

⚙ Domain settings

### ⌚ Production deploys

Production: master@7f7d453 PUBLISHED

Mar 20: typo

Production: master@1839c4d FAILED

Mar 20: fixes

Production: master@58ddec1

Mar 20: add dashboard page

Production: master@cc17049

Mar 20: add page and asset

### 💡 Deploy Previews

#### Auto-preview your PR or branch

Every time you open a pull request, or push new changes to a branch, Netlify automatically builds a preview with a unique URL. Like a staging environment for every PR or branch, previews are perfect for testing and collaboration.

[Learn more about Deploy Previews ↗](#)



# Custom Webhooks



## API Call

```
netlifyIdentity.on('signup', user => {
  axios({
    url: `/identity/functions/login`,
    method: "GET"
  })
}) ;
```

```
netlifyIdentity('signUp', user => {
  axios({
    url: `/.identity/functions/login`,
    method: "GET"
  })
});
```

# Custom Webhooks

API Call



Automagic web hooks

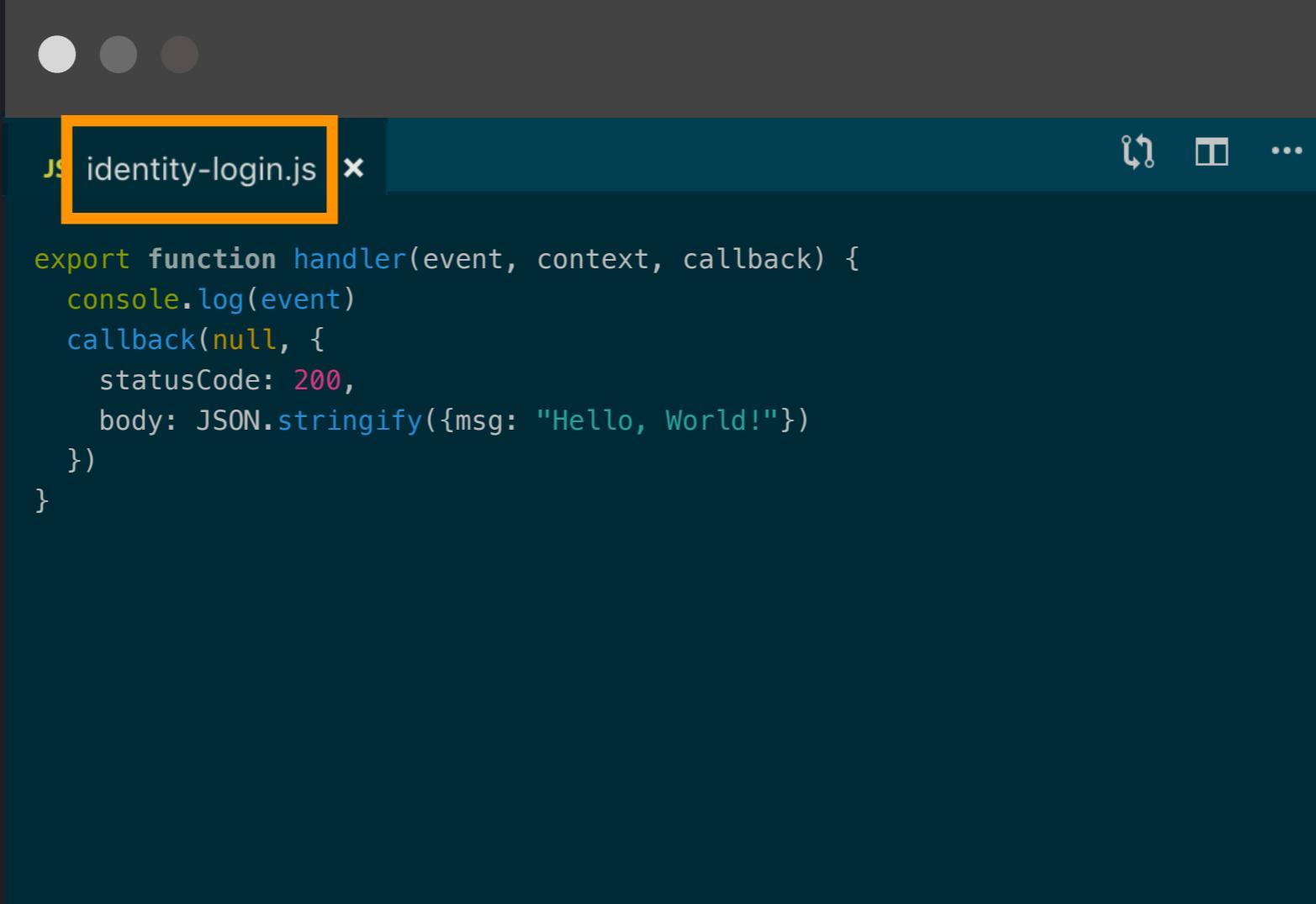
validate

---

signup

---

login



A screenshot of a code editor window titled "identity-login.js". The file contains the following JavaScript code:

```
export function handler(event, context, callback) {
  console.log(event)
  callback(null, {
    statusCode: 200,
    body: JSON.stringify({msg: "Hello, World!"})
  })
}
```

src/lambda/identity-login.js

```
exports.handler = function(event, context, callback) {  
  const {identity, user} = context.clientContext;  
  
  // do some stuff //  
  
  callback(null, {  
    statusCode: 200,  
    body: JSON.stringify({ msg: "hello" })  
  });  
};
```

```
exports.handler = function(event, context, callback) {  
  const {identity, user} = context.clientContext;  
  
  // do some stuff //  
  
  callback(null, {  
    statusCode: 200,  
    body: JSON.stringify({ msg: "hello" })  
  });  
};
```

```
exports.handler = function(event, context, callback) {  
  const {identity, user} = context.clientContext;  
  
  // do some stuff //  
  
  callback(null, {  
    statusCode: 200,  
    body: JSON.stringify({ msg: "hello" })  
  });  
};
```



divya › you-auth-to-know

D

Overview Deployes Functions Identity Forms Large Media Split Testing Settings

## Identity

Identity Free

API endpoint: <https://you-auth-to-know.netlify.com/.netlify/identity>

[Learn more in the documentation ↗](#)

[Settings and usage](#)

[Identity audit log](#)

[Search users](#)

[Invite users](#)



Divya  
divya@netlify.com

No role set

Created on Mar 14 (19 days ago)



Div  
hello@shortdiv.com

No role set

Created on Feb 11 (2 months ago)





```
exports.handler = function(event, context, callback) {
  const data = JSON.parse(event.body);
  const { user } = data;

  console.log(user.email);
  console.log("identity yourself", context.clientContext.identity);

  const validateUser = email => {
    if (email.split("@")[1] === "netlify.com") {
      return ["editor"];
    } else {
      return ["visitor"];
    }
};

const roles = validateUser(user.email);

const responseBody = {
  app_metadata: {
    roles,
    my_user_info: "this is user info that the user can't change from the UI"
  },
  user_metadata: {
    ...user.user_metadata,
    custom_data_from_function: "hurray this is some extra metadata"
  }
};

callback(null, {
  statusCode: 200,
  body: JSON.stringify(responseBody)
});
};
```

```
exports.handler = function(event, context, callback) {
  const data = JSON.parse(event.body);
  const { user } = data;

  console.log(user.email);
  console.log("identity yourself", context.clientContext.identity);

  const validateUser = email => {
    if (email.split("@")[1] === "netlify.com") {
      return ["editor"];
    } else {
      return ["visitor"];
    }
};

const roles = validateUser(user.email);

const responseBody = {
  app_metadata: {
    roles,
    my_user_info: "this is user info that the user can't change from the UI"
  },
  user_metadata: {
    ...user.user_metadata,
    custom_data_from_function: "hurray this is some extra metadata"
  }
};

callback(null, {
  statusCode: 200,
  body: JSON.stringify(responseBody)
});
};
```

```
exports.handler = function(event, context, callback) {
  const data = JSON.parse(event.body);
  const { user } = data;

  console.log(user.email);
  console.log("identity yourself", context.clientContext.identity);

  const validateUser = email => {
    if (email.split("@")[1] === "netlify.com") {
      return ["editor"];
    } else {
      return ["visitor"];
    }
};

const roles = validateUser(user.email);

const responseBody = {
  app_metadata: {
    roles,
    my_user_info: "this is user info that the user can't change from the UI"
  },
  user_metadata: {
    ...user.user_metadata,
    custom_data_from_function: "hurray this is some extra metadata"
  }
};

callback(null, {
  statusCode: 200,
  body: JSON.stringify(responseBody)
});
};
```

```
exports.handler = function(event, context, callback) {
  const data = JSON.parse(event.body);
  const { user } = data;

  console.log(user.email);
  console.log("identity yourself", context.clientContext.identity);

  const validateUser = email => {
    if (email.split("@")[1] === "netlify.com") {
      return ["editor"];
    } else {
      return ["visitor"];
    }
};

const roles = validateUser(user.email);

const responseBody = {
  app_metadata: {
    roles,
    my_user_info: "this is user info that the user can't change from the UI"
  },
  user_metadata: {
    ...user.user_metadata,
    custom_data_from_function: "hurray this is some extra metadata"
  }
};

callback(null, {
  statusCode: 200,
  body: JSON.stringify(responseBody)
});
};
```

```
{  
  app_metadata: {  
    roles,  
    ...user.app_metadata,  
    my_user_info: "some user info"  
  },  
  user_metadata: {  
    ...user.user_metadata,  
    custom_data_from_function: "hurray extra metadata!"  
  }  
};
```

```
{  
  app_metadata: {  
    roles,  
    ...user.app_metadata,  
    my_user_info: "some user info"  
  },  
  user_metadata: {  
    ...user.user_metadata,  
    custom_data_from_function: "hurray extra metadata!"  
  }  
};
```

# Demo Time !

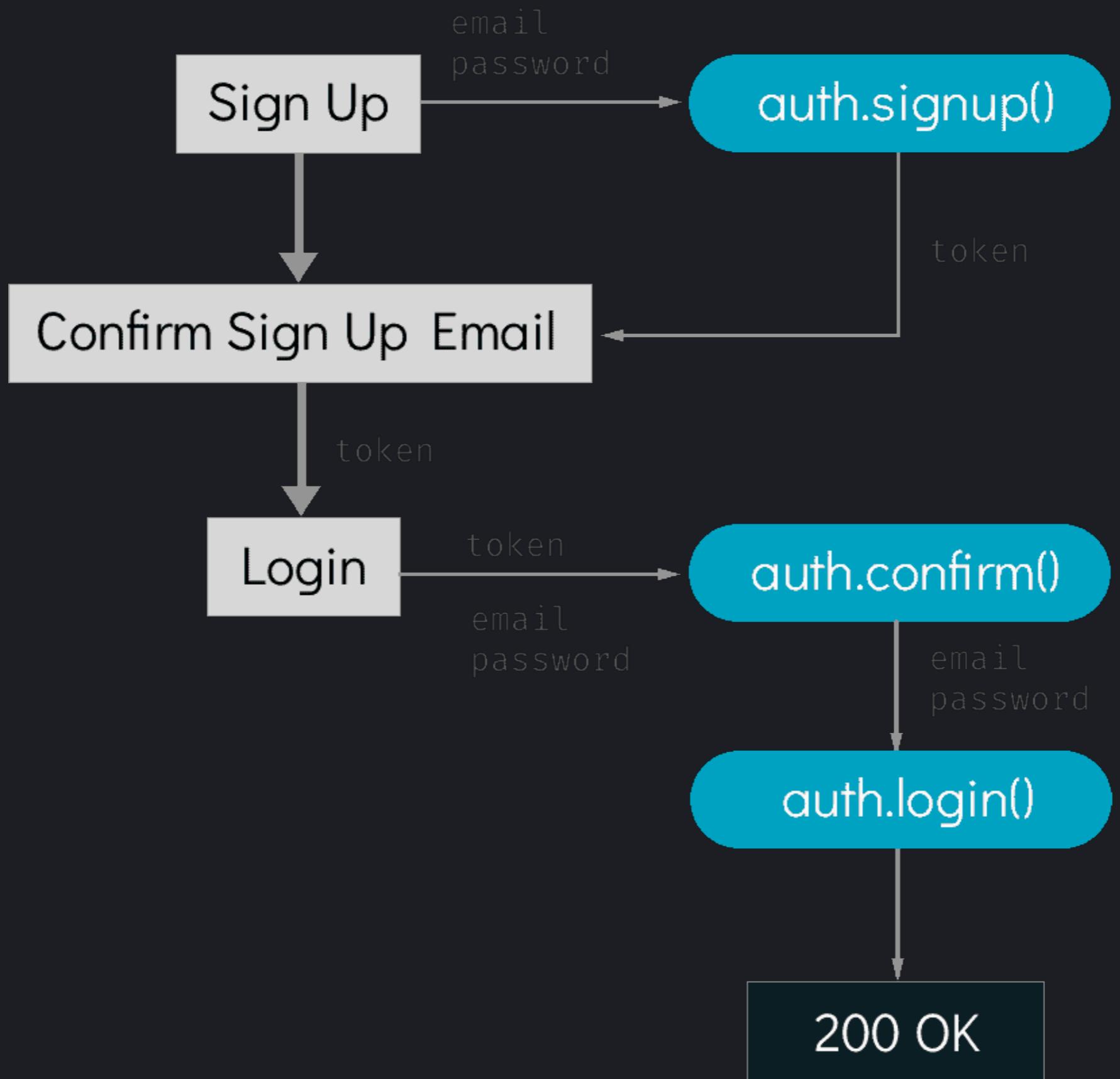
*What about local development?*



No<sup>+</sup>city

# GoTrue

An SWT based API for managing  
users and issuing SWT tokens



```
import GoTrue from "gotrue-js";

export const auth = new GoTrue({
  APIUrl: "https://auth-to-know.com/.netlify/identity",
  audience: "",
  setCookie: false
});
```

```
const attemptLogin = ({ commit, dispatch }, creds) => {
  return new Promise((resolve, reject) => {
    auth
      .login(creds.email, creds.password)
      .then(response => {
        resolve(response);
        commit("SET_CURRENT_USER", response);
      })
      .catch(error => {
        reject(error.json());
      });
  });
};
```

<https://www.netlify.com/blog/>

2018/12/07/gotrue-js---bringing-

authentication-to-static-sites-with-

just-3kb-of-js/



# serverless chicago

<https://noti.st/shortdiv>

*Divya Sasidharan*



@shortdiv