TrackIt

Whitepaper

Cognito & Amplify
Login & Authentication Framework

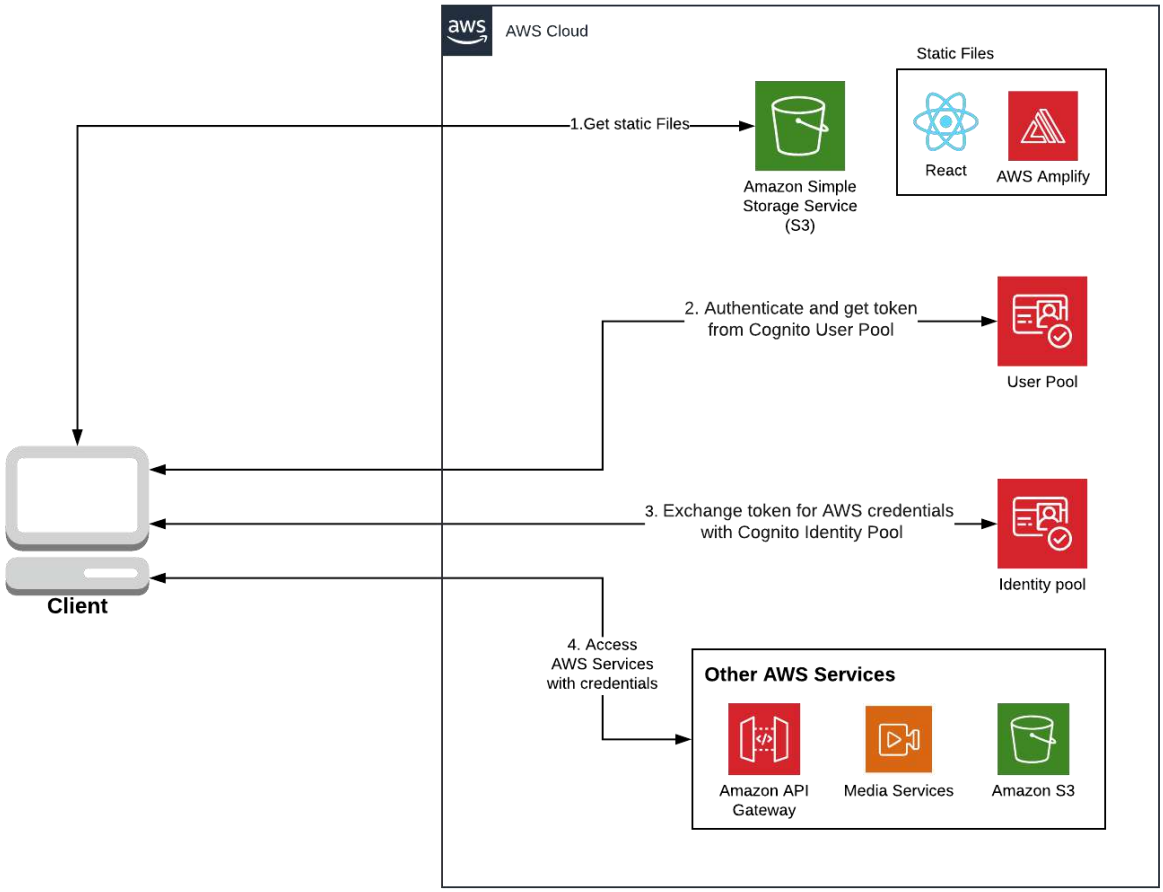
aws partner network

Advanced
Consulting
Partner

Building An Integration Framework

Over the course of many client engagements, we at TrackIt noticed that AWS Cognito login and authentication flows were being used on a recurring basis. To streamline future solutions development, we decided to build a boilerplate/integration framework that would allow clients to kickstart their deployments by quickly interfacing their applications with Cognito.

"During my multiple engagements with clients, I found myself having to do the same technical work over and over again with Cognito. I felt that the natural solution would be to build a tool that makes our development efforts easier." - Jéremy, Front End Software Engineer, TrackIt



https://raw.githubusercontent.com/trackit/aws-cognito-boilerplate/master/aws-cognito-boilerplate.png

Cognito

Amazon Cognito provides authentication, authorization, and user management for web and mobile apps. Amazon Cognito has two main components: user pools and identity pools. User pools are user directories that provide sign up and sign in options for app users. Identity pools on the other hand help grant users access to other AWS services. Identity pools and user pools can be used separately or together.

Amplify

AWS Amplify helps build secure and scalable mobile and web applications. Amplify simplifies the process of authenticating users, securely storing data and user metadata, authorizing selective access to data, integrating machine learning, analyzing application metrics, and executing server-side code.

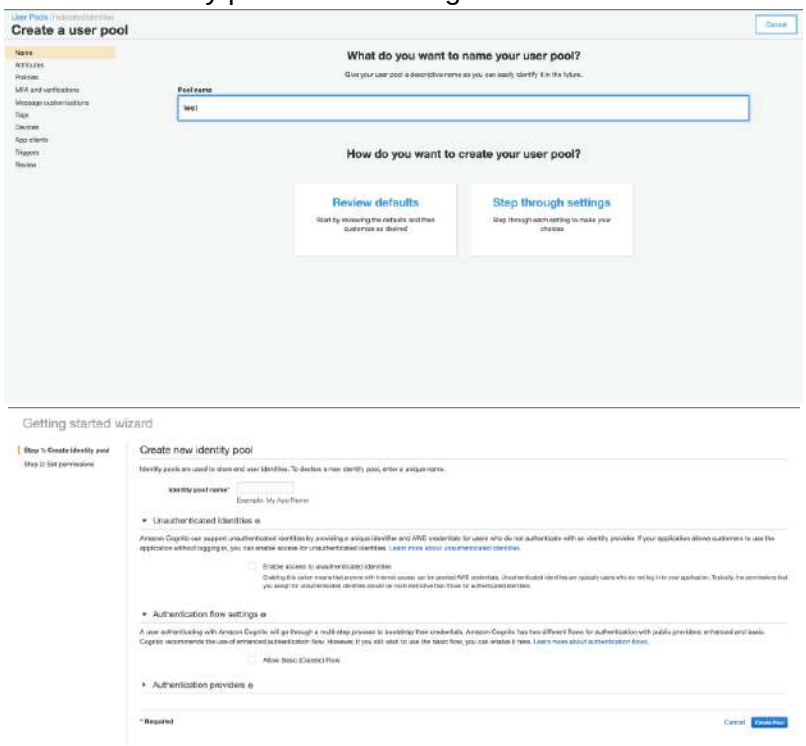
Using Amplify with Cognito is efficient because we can delegate tasks such as token refreshing and authentication to Amplify. Amplify also offers a CLI that automatically configures your project (for Yarn users who are encountering problems with dependencies, take a look at this issue: https://github.com/aws-amplify/amplify-cli/issues/2746)

Amplify provides multiple packages (aws-amplify-react / @aws-amplify/ui-react) which include built-in components for authentication (simple components, higher-order components a.k.a. HOCs, etc.). Amplify also allows customization of fields and appearance.

Note: If you are using custom signUpFields to customize the username field, then you need to make sure that either the label of that field is the same value you set in usernameAttributes or the key of the field is username.

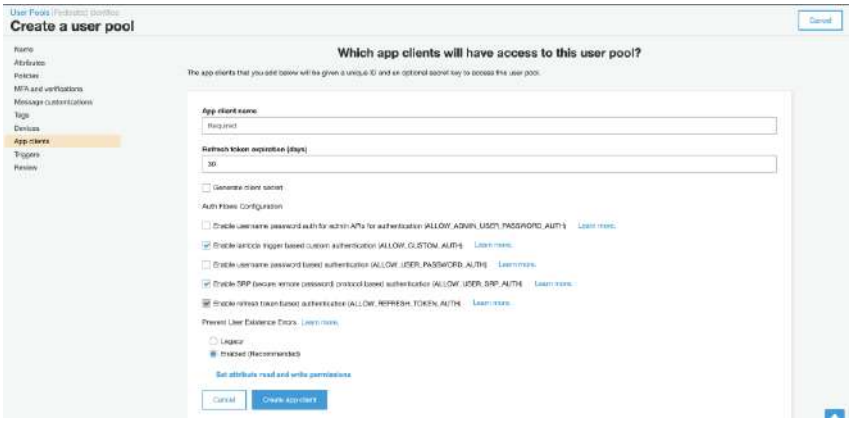
How to Implement Amplify and Cognito without the Amplify CLI

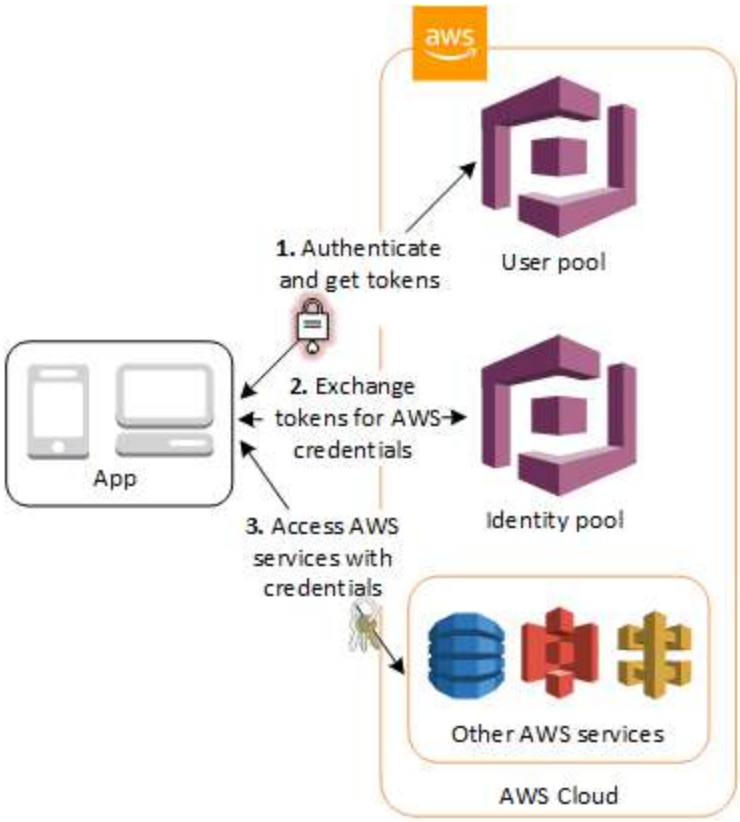
Create a user pool and an identity pool on AWS Cognito



Create an App client

While creating the App client, make sure the “Generate client secret” box is unchecked. JavaScript SDK doesn't support apps that have a client secret.





Create a .env file based on example.env with the correct IDs

```
# Amplify Auth config
REACT_APP_IDENTITY_POOL_ID=""
REACT_APP_REGION=""
REACT_APP_USER_POOL_ID=""
REACT_APP_USER_POOL_WEBCLIENT_ID=""
REACT_APP_API=""
```

The variables mentioned above are used inside an AmplifyConfig object in **`shared/amplify.config.ts`**

The Amplify configuration is based on the instructions mentioned in the following guide: https://aws-amplify.github.io/docs/js/authentication#manual-setup

```
export const amplifyConfig: AmplifyConfig = {
 Auth: {
   identityPoolId: process.env.REACT_APP_IDENTITY_POOL_ID,
   region: process.env.REACT_APP_REGION,
   userPoolId: process.env.REACT_APP_USER_POOL_ID,
   userPoolWebClientId: process.env.REACT_APP_USER_POOL_WEBCLIENT_ID,
 },
 language: "us",
};
```

We created a singleton class extending the Amplify class. This class contains an init() method that takes an AmplifyConfig object as a parameter.

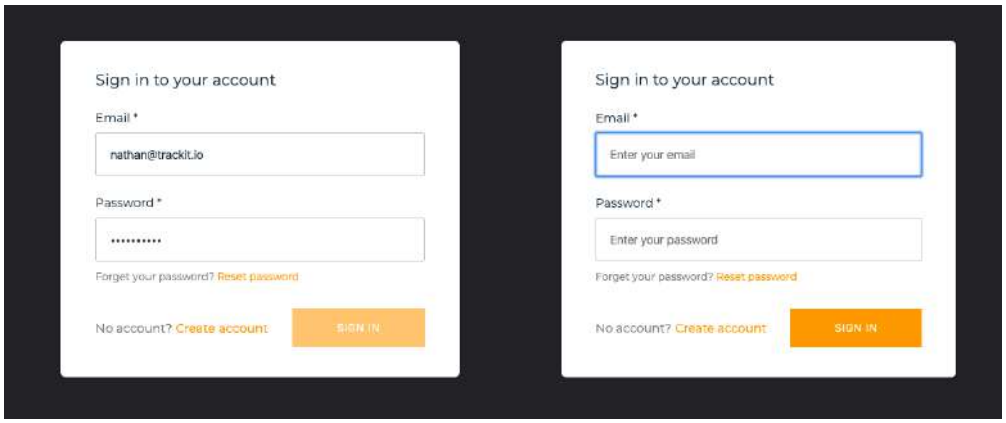
This AmplifyConfig is passed to the static configure() method of Amplify that initializes our app with our configuration file.

You can customize the configuration file inside **`src/shared/amplify.config.ts`**

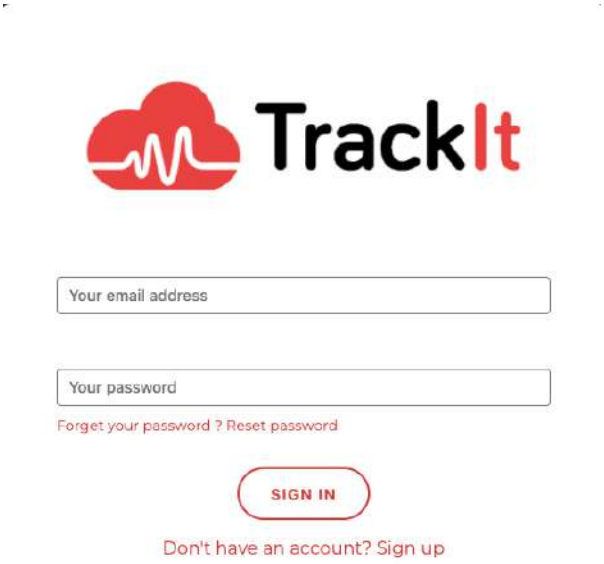
We just have to call Config.getInstance().init() at the root of the project (**`src/index.tsx`**). Amplify will handle the register/login process based on our signUpFields configuration (in shared/amplify.config.ts) and display the wrapped components if the authState changes to signedIn.

Depending on your configuration you can add more services (S3 for example). Amplify will apply these settings and will use your identity pool to grant you access to these Amazon services.

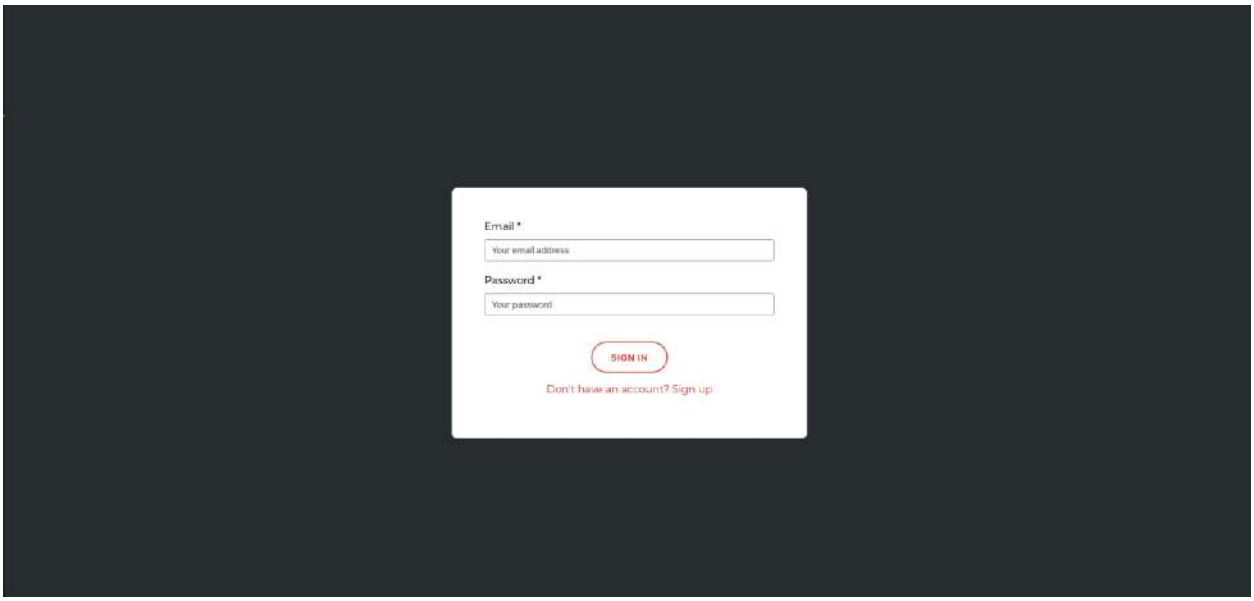
Authentication Flow Visuals



UI Customization



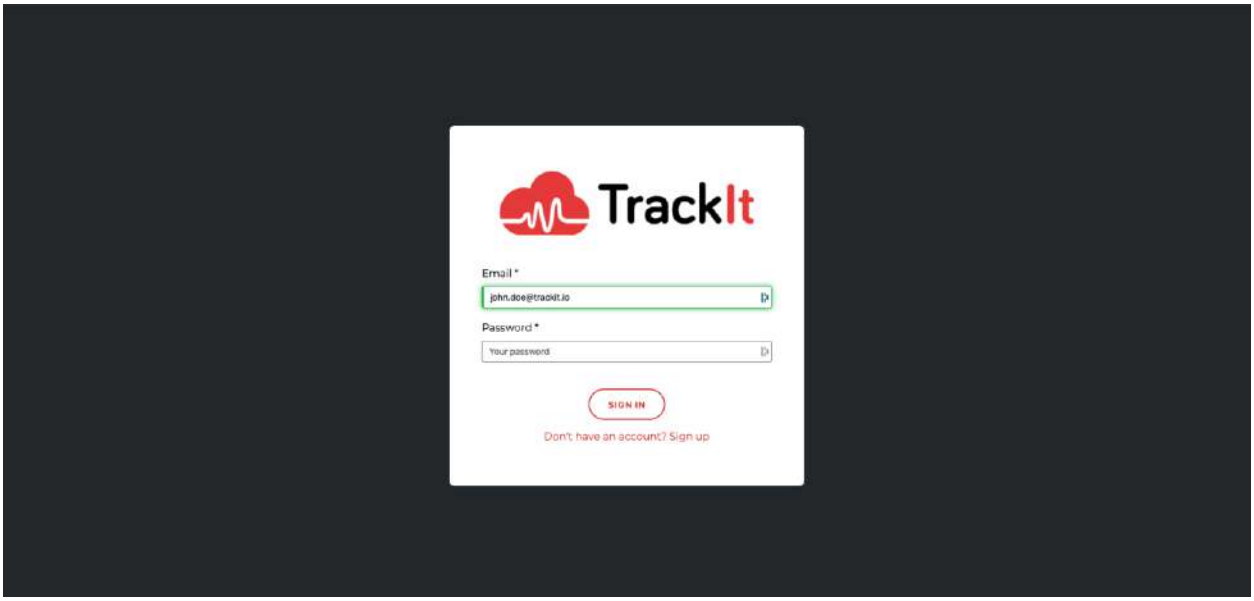
To customize the components and reuse the render logic along with the built-in methods, we need to extend the original components provided by Amplify using React inheritance.



Original components

Here's an example of how you can customize your UI's logo using React inheritance.

To add your logo, you need to display a logo on the Auth components. Often, this logo comes from a URL and thus you need to have a logoUrl property of type string. Since you cannot add new properties, you have to pass a component as a child of our component as a workaround.

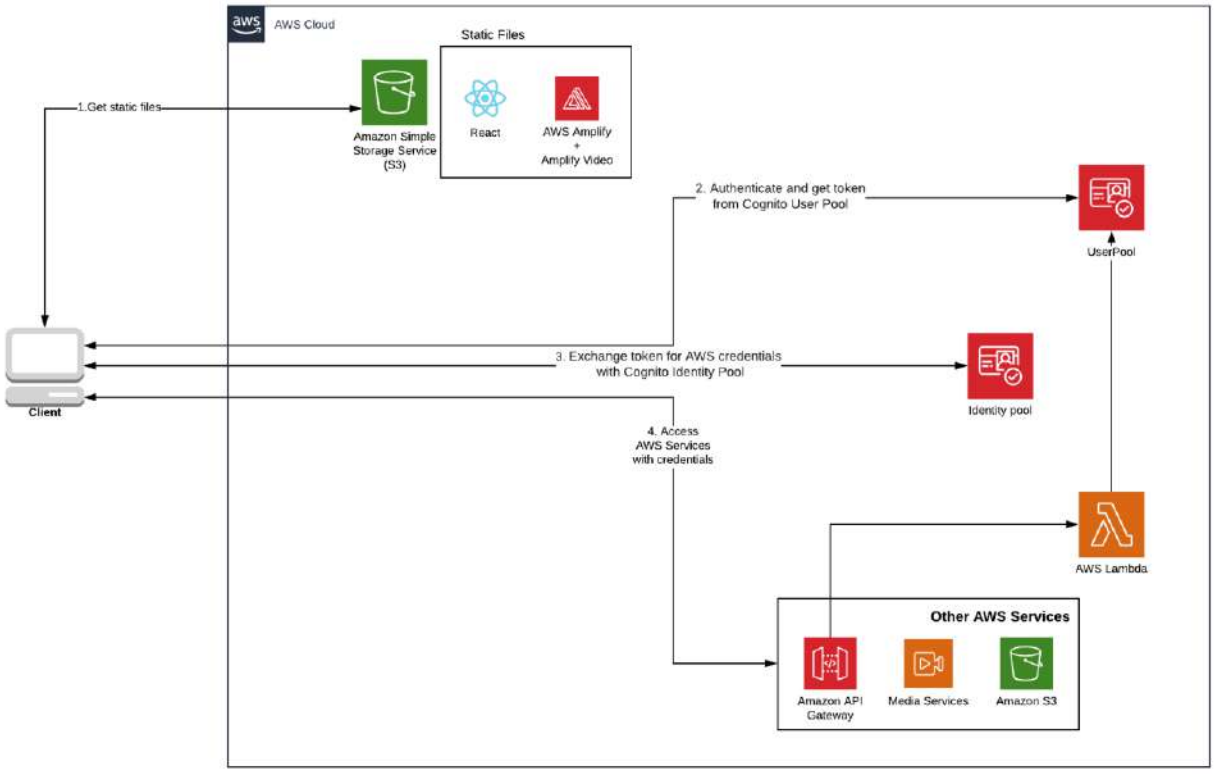


The same process can be used for other UI components as well.

Now your boilerplate is ready to use inside your new or existing projects that require an authentication flow with, or without, custom components.

Here is a link that provides you with the necessary details to customize all your components: https://github.com/trackit/aws-cognito-boilerplate/tree/master/src/shared/components

Sample Use Case: Amplify Video



Here's a sample use case of how to create a React app with role-based user access controls using Cognito groups to allow specific users to watch a live stream.

We added the following features to our original boilerplate:

- Amplify-video - that provides the live stream
- Admin queries - to manage users
- Cognito groups - for role-based user access control using

What follows is a detailed walk through that demonstrates how to implement this live streaming solution with user authentication & management.

Run the following command in the repository:

```
git checkout example/aws-cognito-amplify-video
```

```
yarn
```

or

```
npm i
```

You also have to install Amplify-CLI

```
yarn global add @aws-amplify/cli
```

Or

```
npm i -g @aws-amplify/cli
```

Then run:

```
amplify init
```

```
→  aws-cognito-boilerplate git:(example/aws-cognito-amplify-video) ✗ amplify init
Note: It is recommended to run this command from the root of your app directory
? Enter a name for the project awscognitoboilerplat
? Enter a name for the environment dev
? Choose your default editor: Visual Studio Code
? Choose the type of app that you're building javascript
Please tell us about your project
? What javascript framework are you using react
? Source Directory Path:  src
? Distribution Directory Path: build
? Build Command:  npm run-script build
? Start Command: npm run-script start
Using default provider  awscloudformation


For more information on AWS Profiles, see:
https://docs.aws.amazon.com/cli/latest/userguide/cli-multiple-profiles.html

? Do you want to use an AWS profile? Yes
? Please choose the profile you want to use default
```

Next, add auth and admin queries :

```
amplify add auth
```

```
→  aws-cognito-boilerplate git:(example/aws-cognito-amplify-video) ✗ amplify add auth
Using service: Cognito, provided by: awscloudformation

 The current configured provider is Amazon Cognito.

 Do you want to use the default authentication and security configuration? Default configuration
 Warning: you will not be able to edit these selections.
 How do you want users to be able to sign in? Email
 Do you want to configure advanced settings? No, I am done.
Successfully added resource awscognitoboilerplat30f44007 locally

Some next steps:
"amplify push" will build all your local backend resources and provision it in the cloud
"amplify publish" will build all your local backend and frontend resources (if you have hosting category added) and pr
ovision it in the cloud
```

```
amplify update auth
```

```
→  aws-cognito-boilerplate git:(example/aws-cognito-amplify-video) ✗ amplify update auth
Please note that certain attributes may not be overwritten if you choose to use defaults settings.

You have configured resources that might depend on this Cognito resource.  Updating this Cognito resource could have u
nintended side effects.

Using service: Cognito, provided by: awscloudformation
 What do you want to do? Create or update Cognito user pool groups
? Provide a name for your user pool group: admin
? Do you want to add another User Pool Group Yes
? Provide a name for your user pool group: users
? Do you want to add another User Pool Group No
✔ Sort the user pool groups in order of preference · admin, users
Successfully updated resource awscognitoboilerplat30f44007 locally

Some next steps:
"amplify push" will build all your local backend resources and provision it in the cloud
"amplify publish" will build all your local backend and frontend resources (if you have hosting category added) and pr
ovision it in the cloud
```

```
→  aws-cognito-boilerplate git:(example/aws-cognito-amplify-video) ✗ amplify update auth
Please note that certain attributes may not be overwritten if you choose to use defaults settings.
Using service: Cognito, provided by: awscloudformation
 What do you want to do? Create or update Admin queries API
? Do you want to restrict access to the admin queries API to a specific Group Yes
? Select the group to restrict access with: admin
Successfully added AdminQueriesbbc913ff function locally
Successfully added AdminQueries API locally
Successfully updated resource awscognitoboilerplat30f44007 locally

Some next steps:
"amplify push" will build all your local backend resources and provision it in the cloud
"amplify publish" will build all your local backend and frontend resources (if you have hosting category added) and pr
ovision it in the cloud
```

```
amplify push
```

Copy the .sample.env file and fill REACT_APP_API="AdminQueries"

Next, add amplify-video

```
yarn global add amplify-category-video
```

Or

```
npm i -g amplify-category-video
```

Then:

```
amplify add video
```

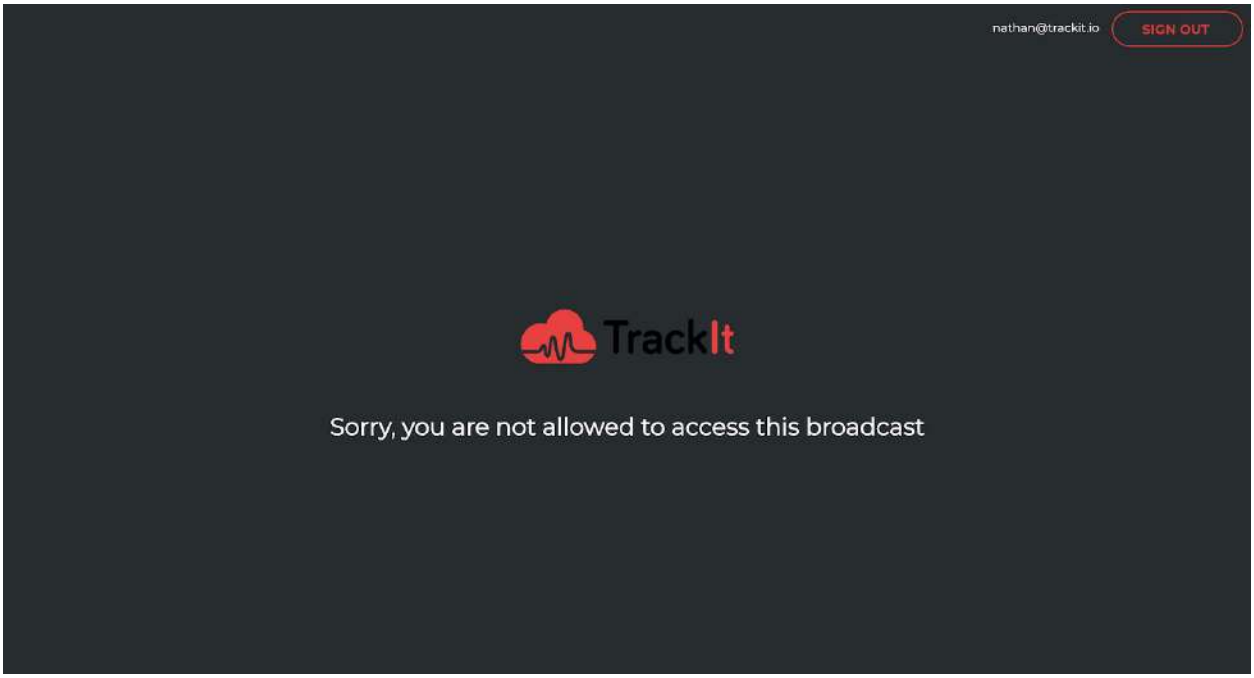
```
→  aws-cognito-boilerplate git:(example/aws-cognito-amplify-video) ✗ amplify add video
? Please select from one of the below mentioned services: Livestream
? Provide a friendly name for your resource to be used as a label for this category in the project: mylivestream
? Do you want to modify any advanced video encoding parameters? No
? Input Security Group:  0.0.0.0/0
? MediaLive ingest type:  RTMP_PUSH
? Encoding Profile:  FULL (6 renditions)
? Auto Start:  Yes
? Where do you want to stream to? MediaStore
Successfully configured mylivestream
```

```
yarn start
```

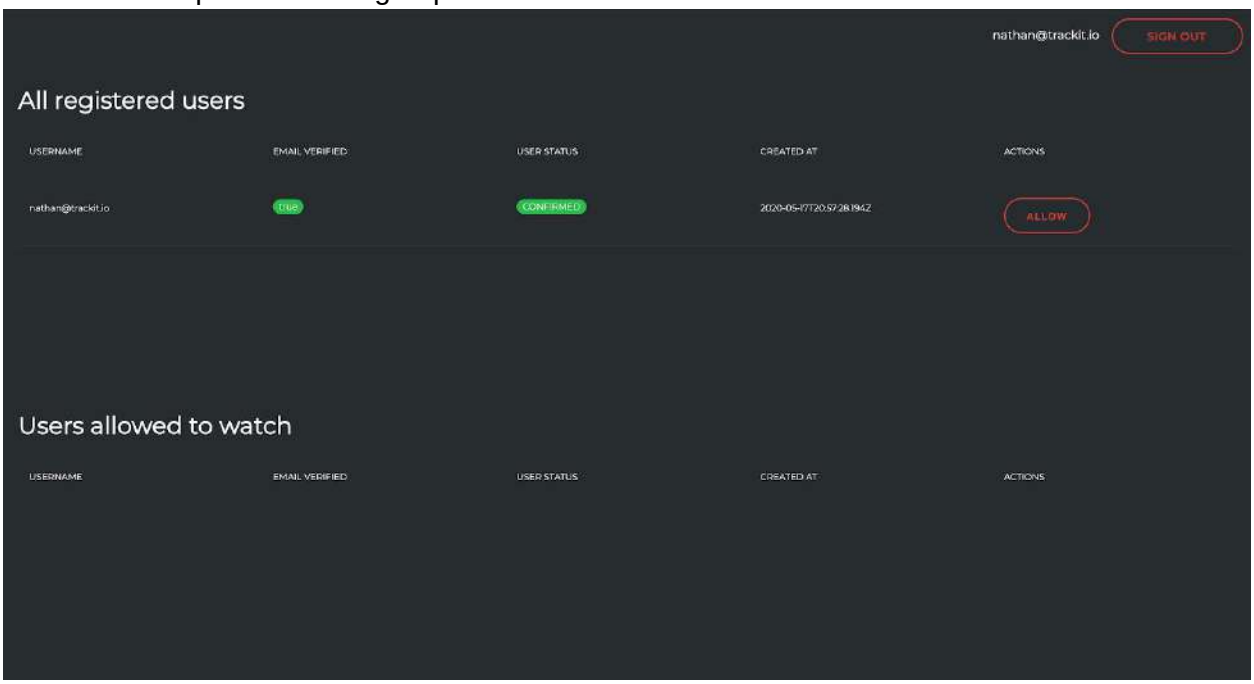
Or

```
npm start
```

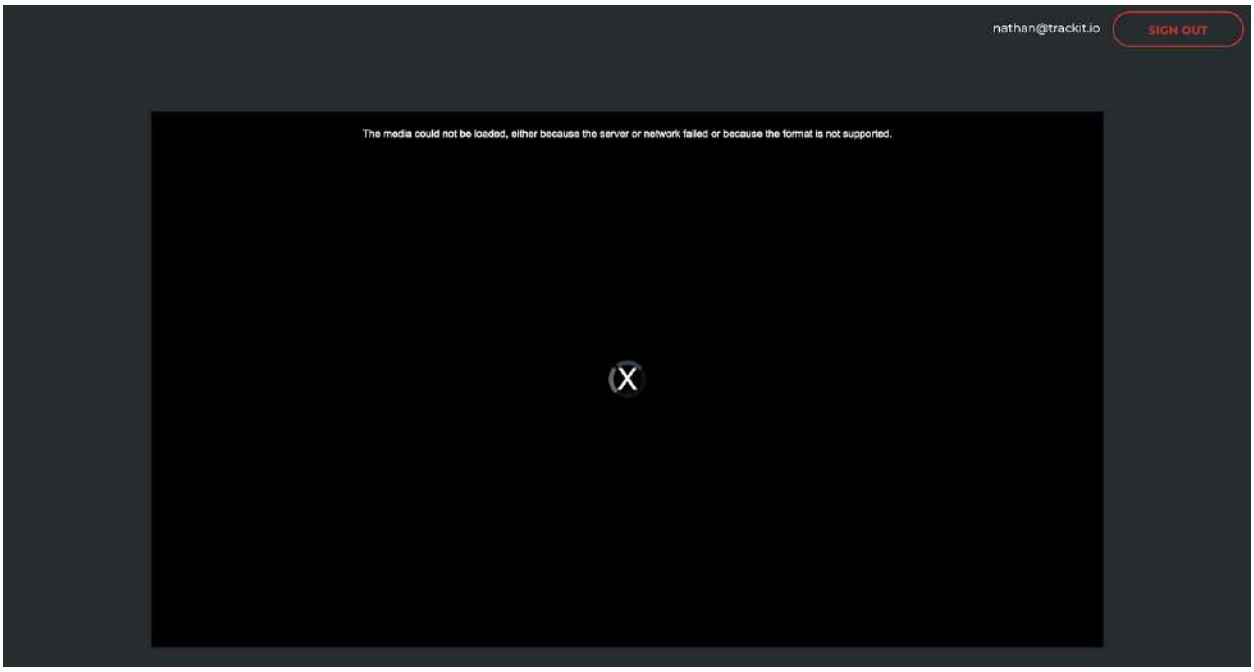
By default, because you are not part of any group, you'll see this screen:



Users that are part of admin group will see this screen:



Standard users that are part of the user group will see this screen:



“At TrackIt, we’re always trying to automate and accelerate our delivery capabilities and this boilerplate allows us to provide solutions more rapidly and repeatedly for our customers” - Ludovic Francois, CEO of TrackIt

GitHub: https://github.com/trackit/aws-cognito-boilerplate/tree/example/aws-cognito-amplify-video

About TrackIt

TrackIt is an Amazon Web Services Advanced Consulting Partner specializing in cloud management, consulting, and software development solutions based in Venice, CA.

TrackIt specializes in Modern Software Development, DevOps, Infrastructure-As-Code, Serverless, CI/CD, and Containerization with specialized expertise in Media & Entertainment workflows, High-Performance Computing environments, and data storage.

TrackIt's forté is cutting-edge software design with deep expertise in containerization, serverless architectures, and innovative pipeline development. The TrackIt team can help you architect, design, build and deploy a customized solution tailored to your exact requirements.

In addition to providing cloud management, consulting, and modern software development services, TrackIt also provides an open-source AWS cost management tool that allows users to optimize their costs and resources on AWS.