

Delivery Documentation

Prepared for Gerald Lofstead by Team UniDrive

Release Notes

- Initial release
- Known bugs/defects:
 - Right click submenu sometimes goes off screen
 - Files names in Folders do not change if opened.
 - Application relies on the user having cookies enabled
 - Google API takes long time or does not update Shared Folders
 - Application UI is not designed for mobile devices
 - On Google Chrome. When a folder is open, starring a file in that folder does not currently update the starred status in the right click menu.
 - On Safari. In the search bar, pressing enter to search does not work.

Install Guide

1. Pre-requisites

The required configuration of software and hardware that the customer must have before they can begin the installation process are a GitHub Account and Google Account.

An AWS Account is required for Hosting. For development, GIT, Node.js and Node Package Manager (NPM) are required.

2. Dependent libraries that must be installed

This section is only required if you plan on developing the application locally. We go through the steps into more details in **Section 7. Run instructions (Local Development)**.

All of the libraries that need to be installed for our software to function can be done so with the command **npm install**.

Here's a list of all the dependent libraries:

```
"dependencies": {  
  "@fortawesome/fontawesome-svg-core": "^1.2.32",  
  "@fortawesome/free-brands-svg-icons": "^5.15.1",  
  "@fortawesome/free-solid-svg-icons": "^5.15.1",  
  "@fortawesome/react-fontawesome": "^0.1.11",
```

```

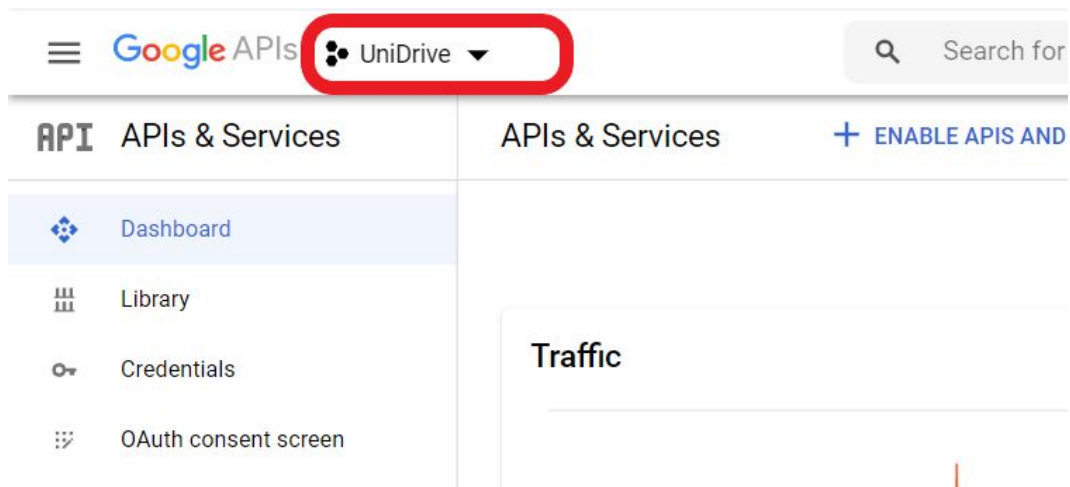
"@testing-library/jest-dom": "^4.2.4",
"@testing-library/react": "^9.5.0",
"@testing-library/user-event": "^7.2.1",
"prop-types": "^15.7.2",
"react": "^16.13.1",
"react-contextmenu": "^2.14.0",
"react-datepicker": "^3.3.0",
"react-dom": "^16.13.1",
"react-scripts": "3.4.3",
"react-select": "^3.1.0",
"universal-cookie": "^4.0.4"
},

```

3. Google Drive API Setup:

To gain access to the project, the costumer/users must following the upcoming steps:

- i. Creating the application in Google Console
 1. Go to <https://console.developers.google.com/>
- ii. Click the dropdown icon. It is outlined in red below. It won't say UniDrive, yet.



- iii. Now, click **New Project** in the top right.



- iv. Enter a project name and select a location. Then, click **Create**.

New Project



You have 11 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)

[MANAGE QUOTAS](#)

Project name *

UniDrive



Project ID:

. It cannot be changed later. [EDIT](#)

Location *

No organization

[BROWSE](#)

Parent organization or folder

[CREATE](#)

[CANCEL](#)

v. Now, click **ENABLE APIS AND SERVICES**.

Google APIs UniDrive Search for APIs and Services

APIs & Services **+ ENABLE APIS AND SERVICES**

APIs & Services

Dashboard

Library

Credentials

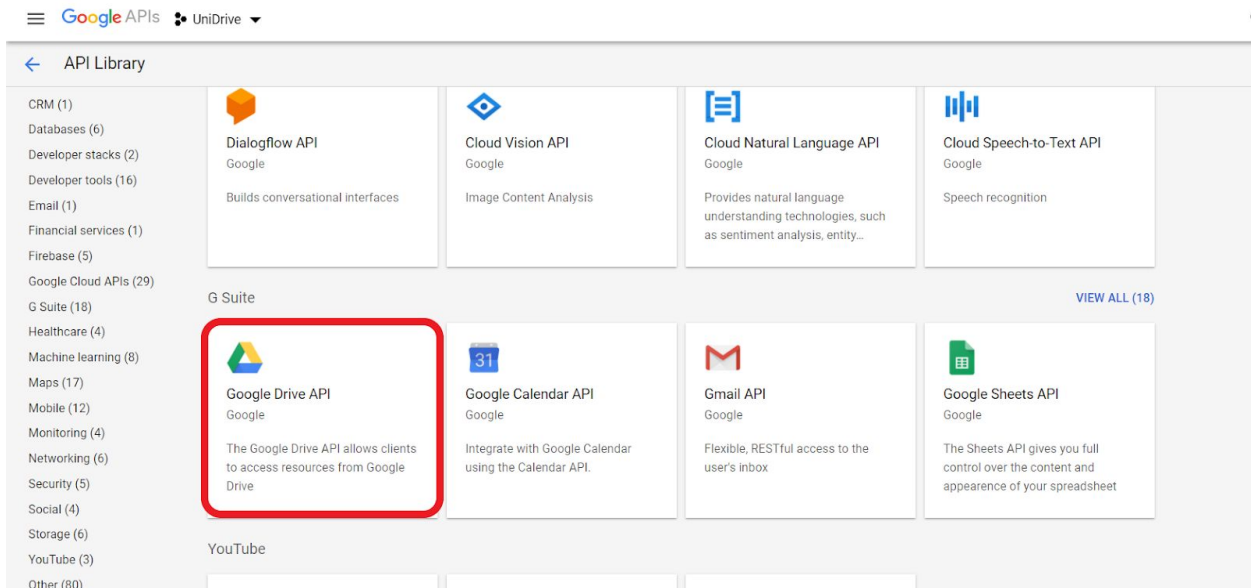
OAuth consent screen

Domain verification

Page usage agreements

i You don't have any APIs available to use yet. To get started, click "Enable APIs and Services"

vi. Scroll down or search for **Google Drive API**. Then, click **Google Drive API**.



vii. Click **ENABLE** to enable the Google Drive API



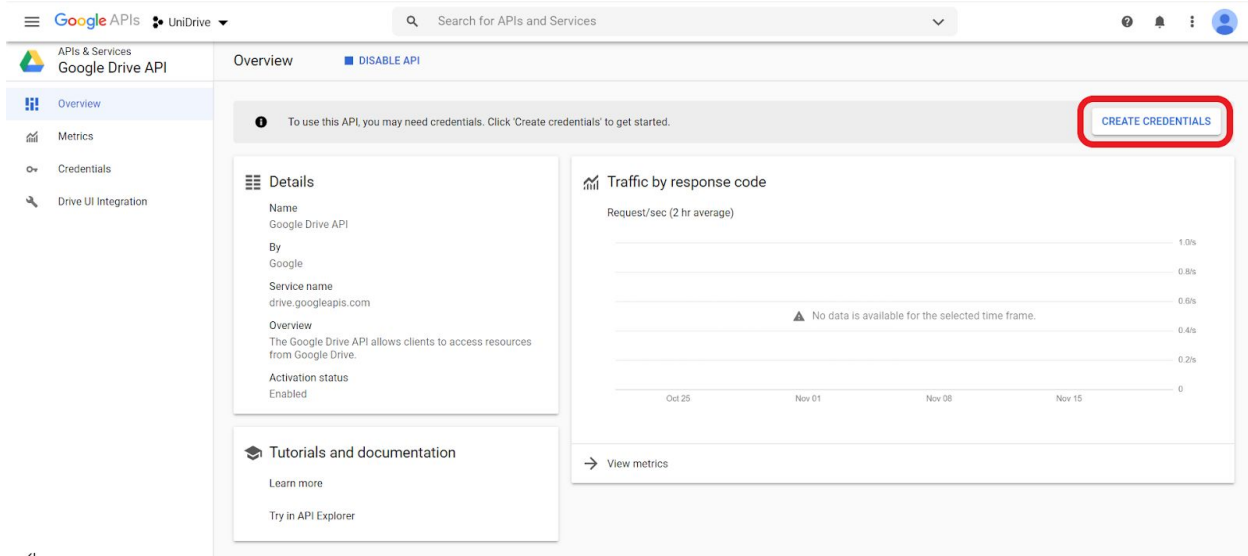
Google Drive API

Google

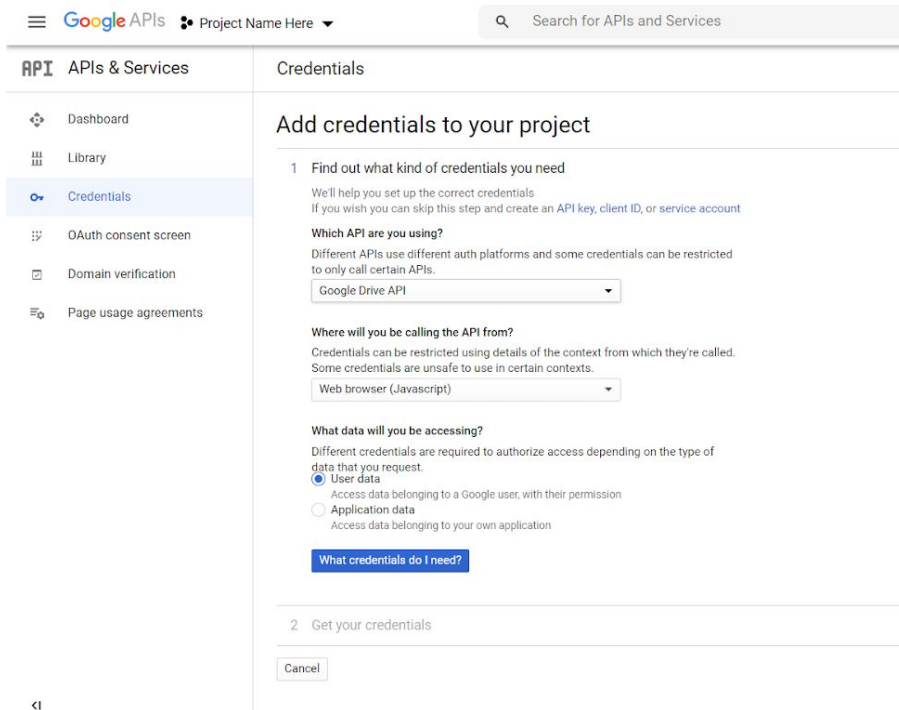
The Google Drive API allows clients to access resources from Google Drive



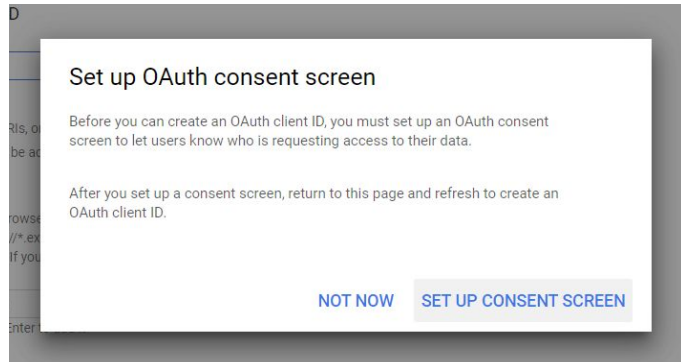
viii. Click **CREATE CREDENTIALS** to the top right side of the screen.



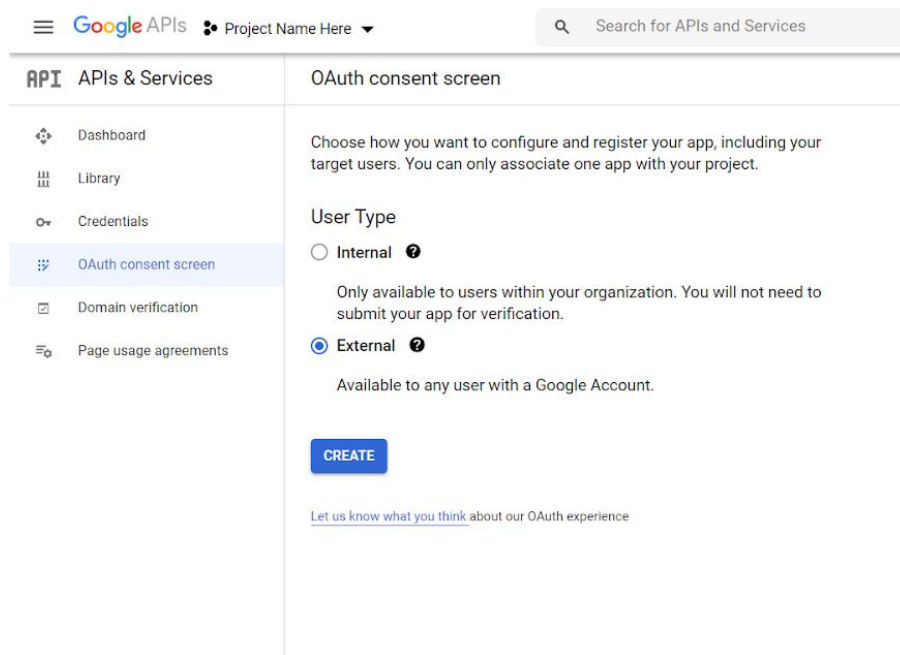
- ix. Under the **Add credentials to your project** screen
1. **Which API are you using?**
 - a. Select **Google Drive API**
 2. **Where will you be calling the API from?**
 - a. Select **Web browser (JavaScript)**
 3. **What data will you be accessing?**
 - a. Select **User data**
 4. Then, select the **What credentials do I need?** button.



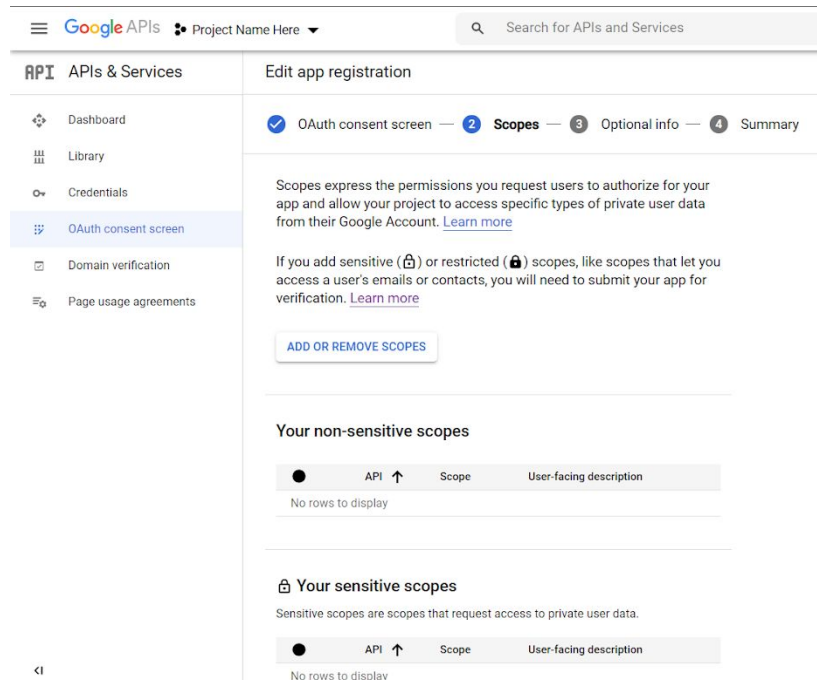
- x. Select **SETUP CONSENT SCREEN**.



- xi. Under the **OAuth consent screen**,
1. Select **External** as the user type
 2. Then, click **CREATE**.



- xii. Enter the **App name** and **User support email**.



- xv. Under the scope selection screen,
1. Select the following required scopes.
 - a. **.../auth/drive**
 - b. **.../auth/drive.appdata**
 - c. **.../auth/drive.file**
 - d. **.../auth/drive.metadata.readonly**
 2. Double check to ensure you've selected the correct scopes. Then, click **UPDATE** at the bottom of the screen.

Filter table

	API	Scope	User-facing description
<input type="checkbox"/>		.../auth/userinfo.email	View your email address
<input type="checkbox"/>		.../auth/userinfo.profile	See your personal info, including any personal info you've made publicly available
<input type="checkbox"/>		openid	Associate you with your personal info on Google
<input type="checkbox"/>	Google Drive API	.../auth/docs	See, edit, create, and delete all of your Google Drive files
<input checked="" type="checkbox"/>	Google Drive API	.../auth/drive	See, edit, create, and delete all of your Google Drive files
<input checked="" type="checkbox"/>	Google Drive API	.../auth/drive.appdata	View and manage its own configuration data in your Google Drive
<input checked="" type="checkbox"/>	Google Drive API	.../auth/drive.file	View and manage Google Drive files and folders that you have opened or created with this app
<input type="checkbox"/>	Google Drive API	.../auth/drive.metadata	View and manage metadata of files in your Google Drive
<input checked="" type="checkbox"/>	Google Drive API	.../auth/drive.metadata.readonly	View metadata for files in your Google Drive
<input type="checkbox"/>	Google Drive API	.../auth/drive.photos.readonly	View the photos, videos and albums in your Google Photos

Rows per page: 10 1 - 10 of 19

Manually add scopes

If the scopes you would like to add do not appear in the table above, you can enter them here. Each scope should be on a new line or separated by commas. Please provide the full scope string (beginning with 'https://'). When you are finished, click 'Add to table'.

ADD TO TABLE

UPDATE

xvi. Click **SAVE AND CONTINUE**. This screen contains optional info you do not need to fill.

Google APIs Project Name Here

Search for APIs and Services

APIs & Services

- Dashboard
- Library
- Credentials
- OAuth consent screen**
- Domain verification
- Page usage agreements

Edit app registration

OAuth consent screen — Scopes — **Optional info** — Summary

You can speed up the verification process by providing Google reviewers with more, helpful details about your app.

Optional info

Share email addresses of any Google contacts you've had in the past

Share any final details about your app. Include any information that will help us with verification, like the Project IDs of any other projects that use OAuth.

0 / 1000

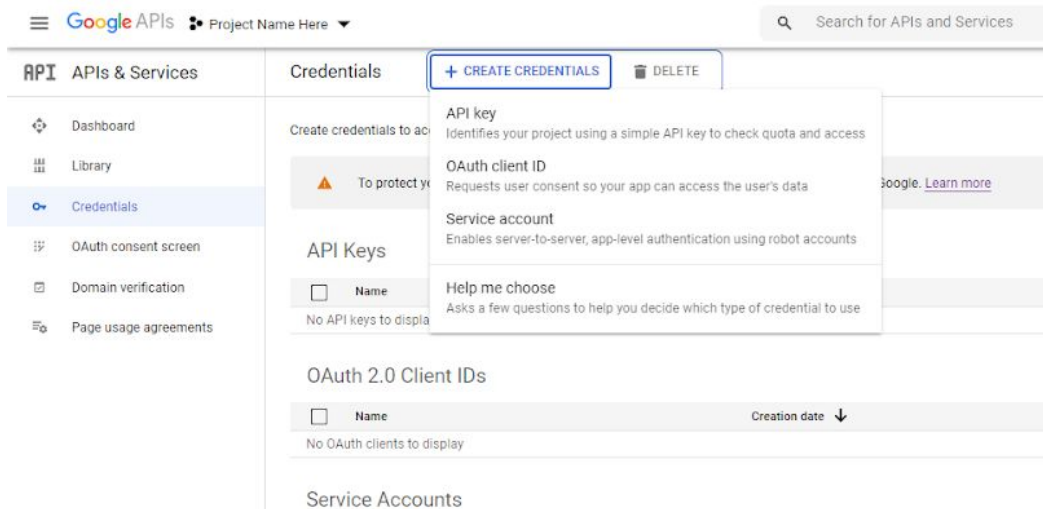
Provide up to 3 more links to any relevant documentation

SAVE AND CONTINUE **CANCEL**

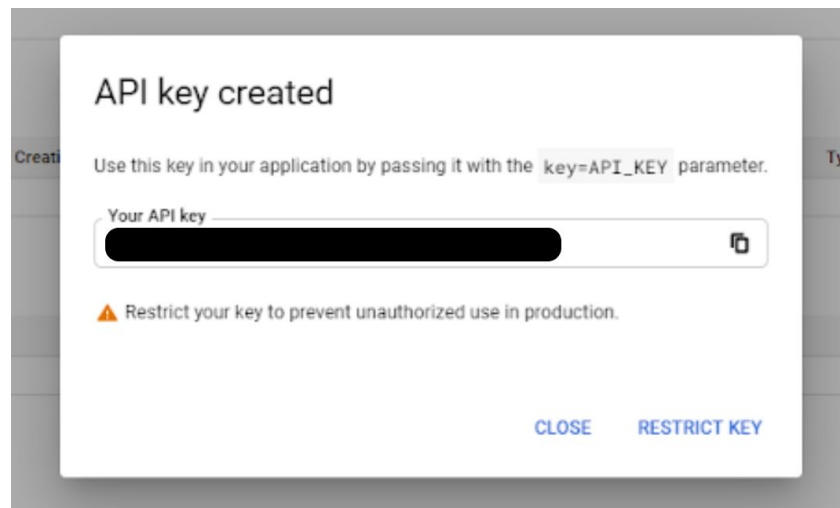
4. API and Client ID Setup:

a. Setting up your API Key

- i. In the **Google API Console**,
 1. Click the **Credentials** tab to the left
 2. Then, click **CREATE CREDENTIALS**
 3. Select **API key**.



- ii. Note the API key listed. We will be referring to this key later in the AWS section. Select **RESTRICT KEY**.



- iii. Under the **API restrictions** towards the lower half of the page,
 1. Select **Restrict key**.
 2. Select the **Google Drive API** in the dropdown select item.
 3. Note the API key listed in the top right. We will be referring to this key later in the AWS section.
 4. Then, click **SAVE**.

Google APIs Project Name Here

Search for API

APIs & Services

Dashboard

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

Restrict and rename API key

REGENERATE KEY

DELETE

Name *

API key 1

Key restrictions

This key is unrestricted. Restrictions help prevent unauthorized use and quota theft. [Learn more](#)

Application restrictions

An application restriction controls which websites, IP addresses, or applications can use your API key. You can set one application restriction per key.

☒ None
 ☐ HTTP referrers (web sites)
 ☐ IP addresses (web servers, cron jobs, etc.)
 ☐ Android apps
 ☐ iOS apps

API restrictions

API restrictions specify the enabled APIs that this key can call

☐ Don't restrict key
This key can call any API
 ☒ Restrict key

1 API

Selected APIs:

Google Drive API

Note: It may take up to 5 minutes for settings to take effect

SAVE CANCEL

b. Setting up your API Key

- i. In the **Google API Console**,
 1. Click the **Credentials** tab to the left
 2. Then, click **CREATE CREDENTIALS**
 3. Select **OAuth client ID**.

Google APIs Project Name Here

Search for APIs and Services

APIs & Services

Dashboard

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

Credentials

+ CREATE CREDENTIALS

DELETE

Create credentials to access your project's APIs

To protect your project, you must create credentials to access your project's APIs.

API Keys

☐ Name
No API keys to display

OAuth 2.0 Client IDs

☐ Name
Creation date
No OAuth clients to display

API key
Identifies your project using a simple API key to check quota and access

OAuth client ID
Requests user consent so your app can access the user's data

Service account
Enables server-to-server, app-level authentication using robot accounts

Help me choose
Asks a few questions to help you decide which type of credential to use

- ii. Fill out the **Application type** and **Name**. Then, click **CREATE**.
 - Application Type is Web Application

Google APIs Project Name Here

APIs & Services

Dashboard

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information.

Application type *

Web application

[Learn more](#) about OAuth client types

Name *

Web client 1

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

Authorized JavaScript origins

For use with requests from a browser

+ ADD URI

Authorized redirect URIs

For use with requests from a web server

+ ADD URI

CREATE CANCEL

- iii. Note the **Client ID** displayed. We will be referring to this client ID in the AWS section later on. Click **OK**.

Note: the client ID is the entire value listed including the .app.googleusercontent.com

OAuth client created

The client ID and secret can always be accessed from Credentials in APIs & Services

OAuth is limited to 100 [sensitive scope logins](#) until the [OAuth consent screen](#) is verified. This may require a verification process that can take several days.

Your Client ID

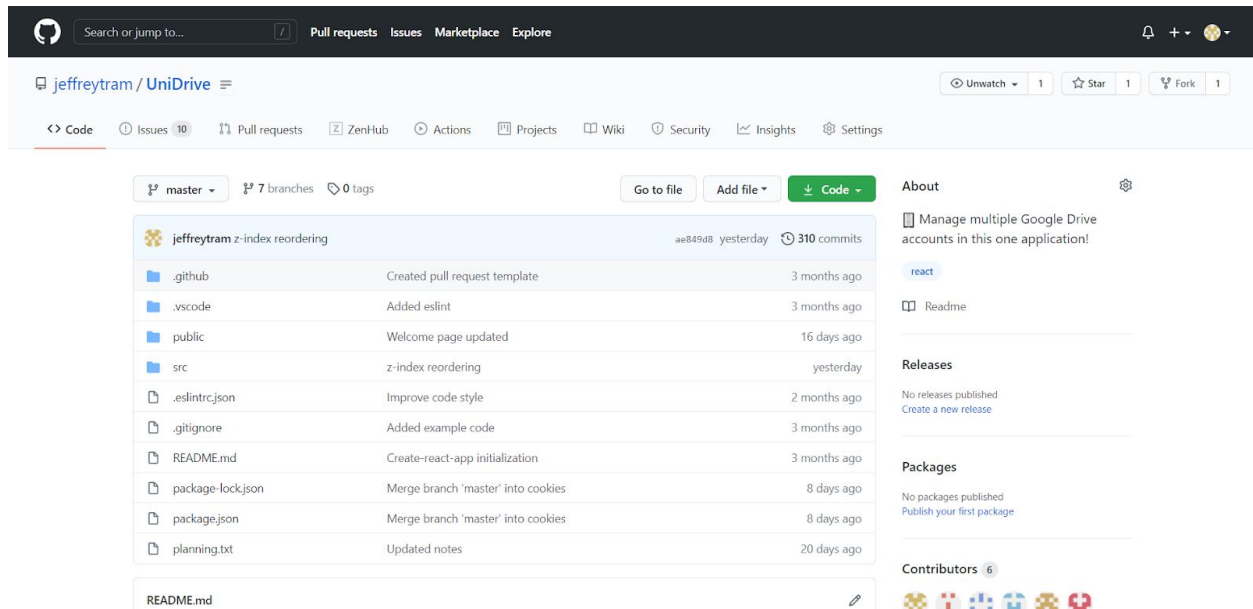
Your Client Secret

OK

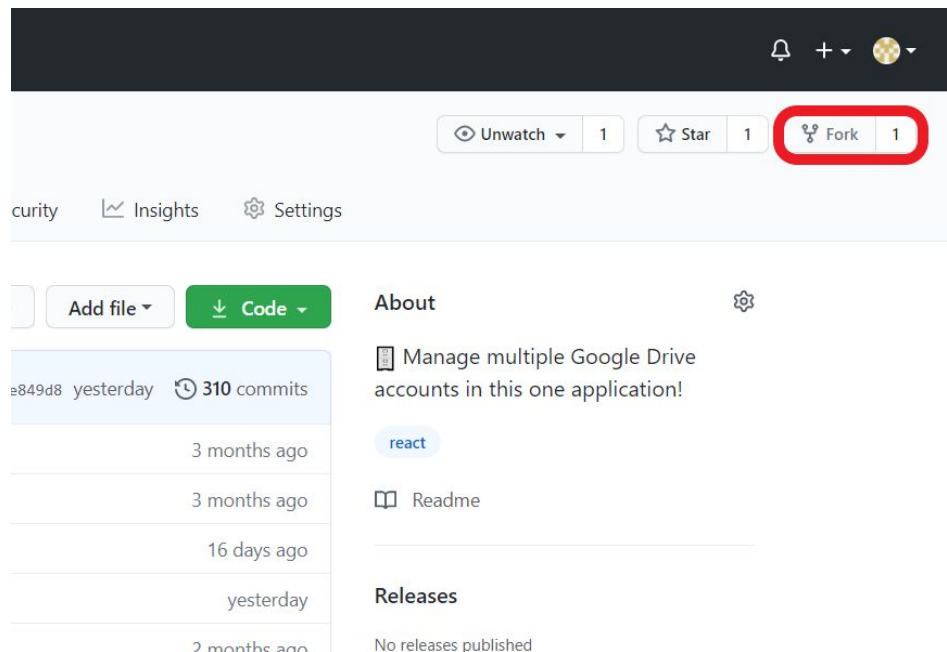
5. Build instructions (Amazon Web Services - Amplify):

a. GitHub Setup:

- i. Go to the following link to the UniDrive GitHub repository
<https://github.com/jeffreytram/UniDrive>

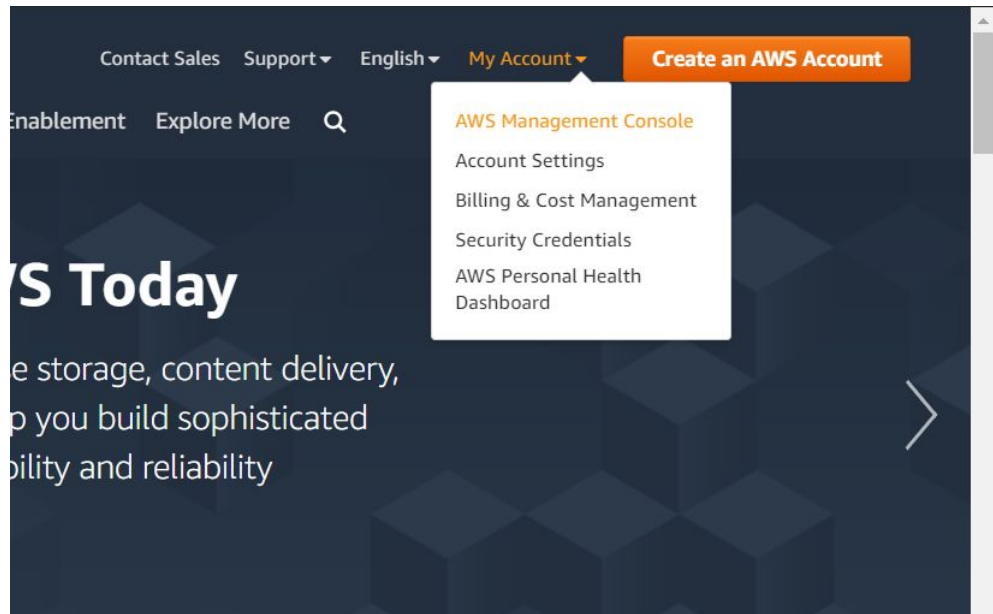


- ii. Fork the repository by clicking the **Fork** button on the top right. Forking will be required since we will be linking our GitHub account in AWS to build the application later on.



b. AWS Amplify Set Up:

- i. Go to <https://aws.amazon.com/>
- ii. At the top right, hover over **My Account**, then click **AWS Management Console**



- iii. Login to your AWS account. Create an AWS account if you do not have one. You will be redirected to the AWS Management Console upon successfully logging in.



Sign in

☒ **Root user**
Account owner that performs tasks requiring unrestricted access. [Learn more](#)

☐ **IAM user**
User within an account that performs daily tasks. [Learn more](#)

Root user email address

username@example.com

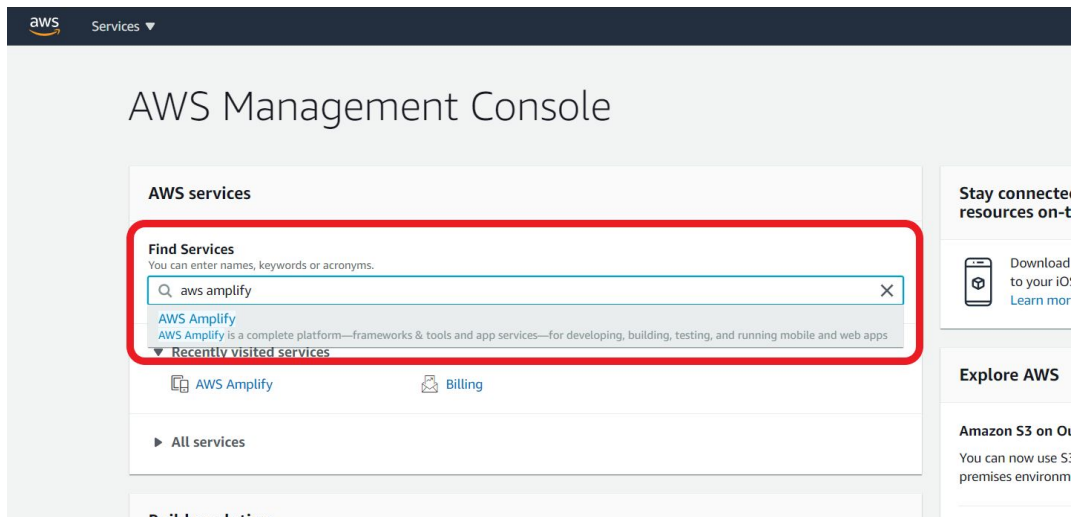
Next

New to AWS?

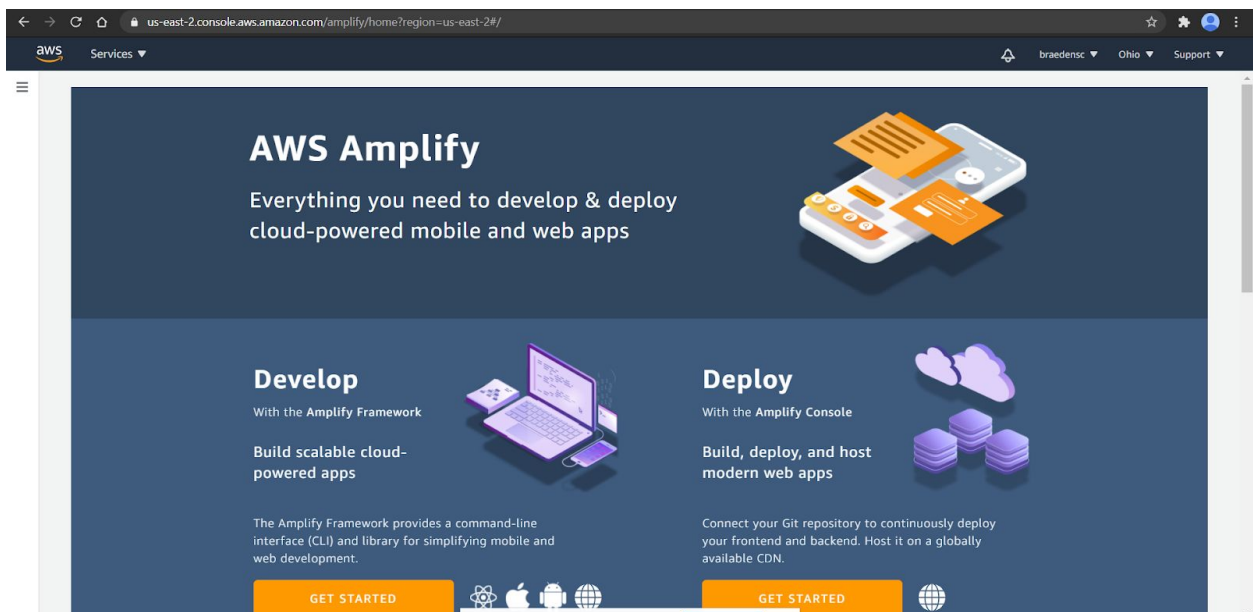
Create a new AWS account



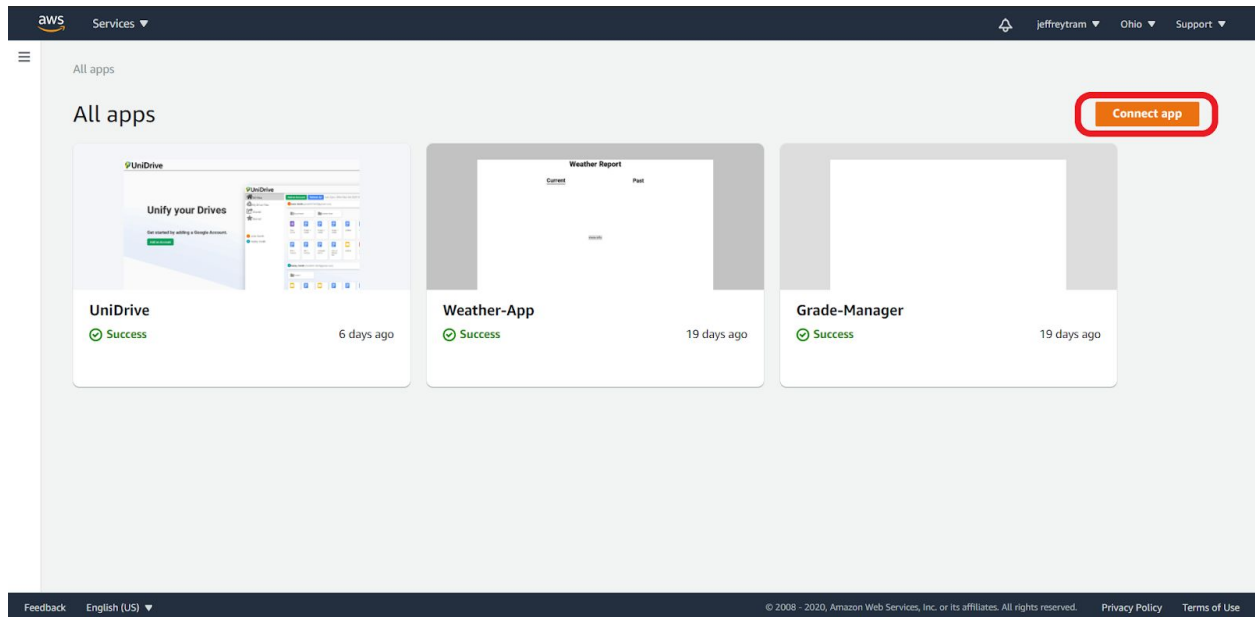
- iv. Under **Find Services**, search for **AWS Amplify**. Then, select the **AWS Amplify** search suggestion that pops up.



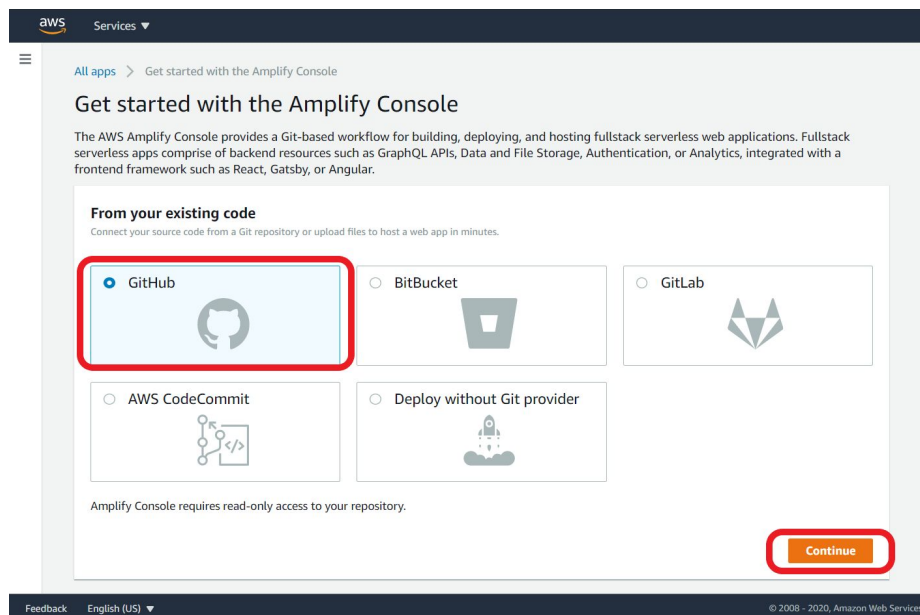
- v. If this is your first time creating a AWS Amplify application
1. Under **Deploy**, click **Get Started**



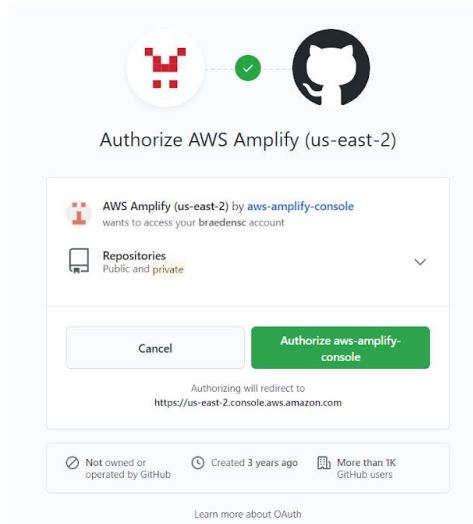
- vi. If this is not your first AWS Amplify application, click **Connect App** to the top right of the screen. Otherwise, proceed to the next step.



vii. Select **GitHub** as the location of the source code. Then, **click Continue**.



viii. **Authorize with github.**



- ix. Select the correct repository (the one with ***yourGitHubUserName/UniDrive***) from the **recently updated repository** drop-down menu and then **click Next**.

Add repository branch

GitHub


✔ GitHub authorization was successful.

Repository service provider

GitHub

Recently updated repositories

If you don't see your repository below, please push a commit and then click the refresh button.

Select a repository ▼ 

Cancel Previous **Next**

- x. Select the branch you want to build. In this case, we select the **unidrive branch** since that is our build branch. Then, **click Next**.

Add repository branch

GitHub

✔ GitHub authorization was successful.

Repository service provider

GitHub

Recently updated repositories

If you don't see your repository below, please push a commit and then click the refresh button.

braedensc/UniDrive

Branch

Select a branch from your repository.

unidrive

☐ Connecting a monorepo? Pick a folder.

Cancel Previous Next

xi. Enter the **App name**. Then, click **Next**.

Configure build settings

App build and test settings

App name

Pick a name for your app.

UniDrive

Name cannot contain periods

Build and test settings

We've auto-detected your app's build settings. Please ensure your build command and output folder (baseDirectory) are correctly detected.

```

1 version: 1
2 frontend:
3   phases:
4     preBuild:
5       commands:
6         - npm ci
7     build:
8       commands:
9         - npm run build
10  artifacts:
11    baseDirectory: build
12    files:
13      - '**/*'
14  cache:
15    paths:
16      - node_modules/**/*
17

```

Build and test settings

Download Edit

► Advanced settings

Cancel Previous Next

xii. Click **Save and Deploy**

Review

Repository details

Repository service GitHub	Branch environment Application root
Repository braedensc/UniDrive	
Branch unidrive	

App settings

Edit

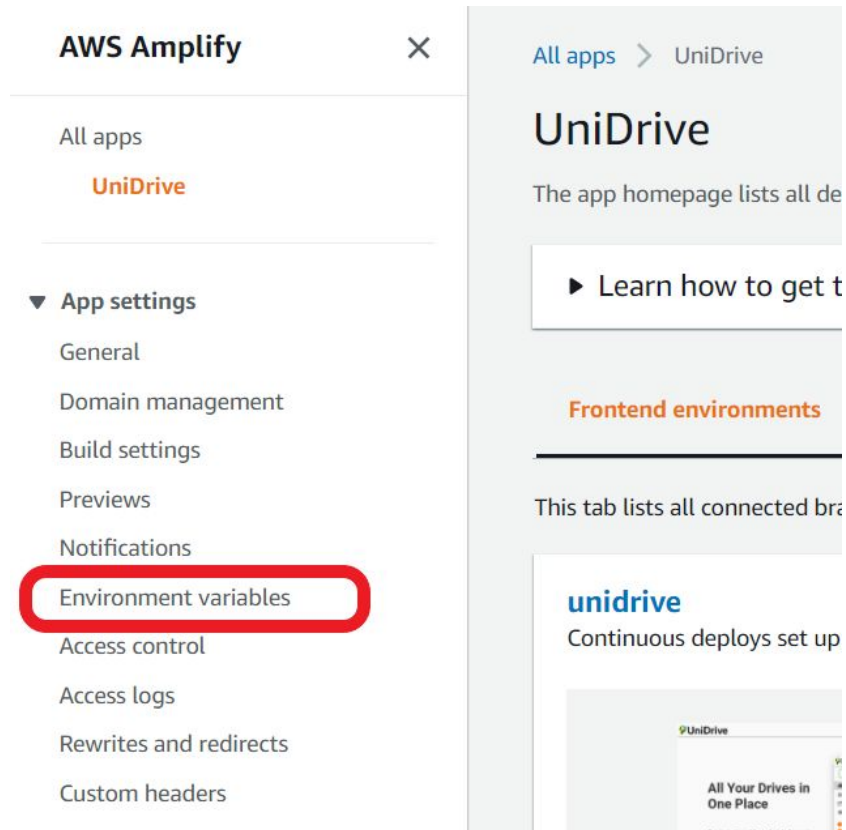
App name UniDrive	Framework React
Build image Using default image	Build settings Auto-detected settings will be used
Environment variables None	

Cancel Previous Save and deploy

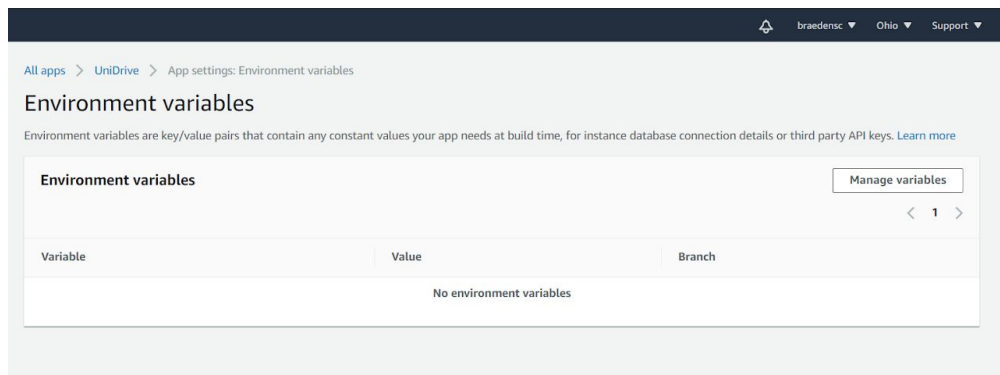
- xiii. Open the sidebar menu by **clicking the 3 horizontal lines** in the top left corner of the screen if it is not already open.

The screenshot shows the AWS Amplify console interface. At the top, there's a dark blue header with the AWS logo and a 'Services' dropdown. Below the header, a sidebar menu is visible on the left, with a red arrow pointing to the hamburger menu icon (three horizontal lines). The main content area displays the 'UniDrive' app configuration page. It includes a breadcrumb 'All apps > UniDrive', a title 'UniDrive', and a description 'App homepage lists all deployed frontend and backend environments.' Below this, there's a link to 'Learn how to get the most out of Amplify Console'. The page is divided into two tabs: 'Frontend environments' (selected) and 'Backend environments'. A message states 'This tab lists all connected branches, select a branch to view build details.' The 'unidrive' section shows 'Continuous deploys set up' and a preview of the app's homepage. At the bottom, there's a deployment status section with 'Last deployment' (11/20/2020, 3:09:56 PM) and 'Last commit' (This is an autogenerated commit from GitHub - unidrive).

- xiv. Click **Environment Variables** on the side-bar.



xv. Click **Manage Variables** at the right side of the screen.



xvi. Click **add variable**, so that there are 2 total environmental variable spots.

Environment variables

Environment variables are key/value pairs that contain any constant values your app needs at build time, for instance database connection details or third party API keys. [Learn more](#)

Manage variables

Variable	Value	Branch	Action
<input type="text" value="Enter variable here"/>	<input type="text" value="Enter value here"/>	All branches	Actions ▼
<input type="text" value="Enter variable here"/>	<input type="text" value="Enter value here"/>	All branches	Actions ▼

- xvii. Add 2 new environment variables:
- For the first environment variable:**
 - Enter **REACT_APP_API_KEY** as variable
 - Enter the **API KEY** that you set up in the Google developer page as value
 - For the second environment variable:**
 - Enter **REACT_APP_CLIENT_ID** as variable
 - Enter the **CLIENT ID** that you set up in the Google developer page as value
 - Then, **click SAVE**.

All apps > UniDrive > App settings: Environment variables

Environment variables

Environment variables are key/value pairs that contain any constant values your app needs at build time, for instance database connection details or third party API keys. [Learn more](#)

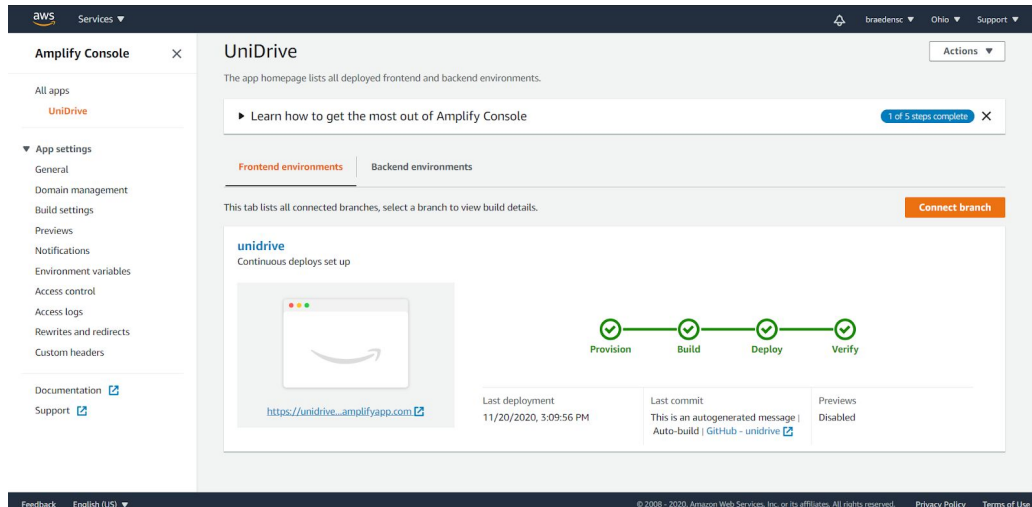
Environment variables

Manage variables

< 1 >

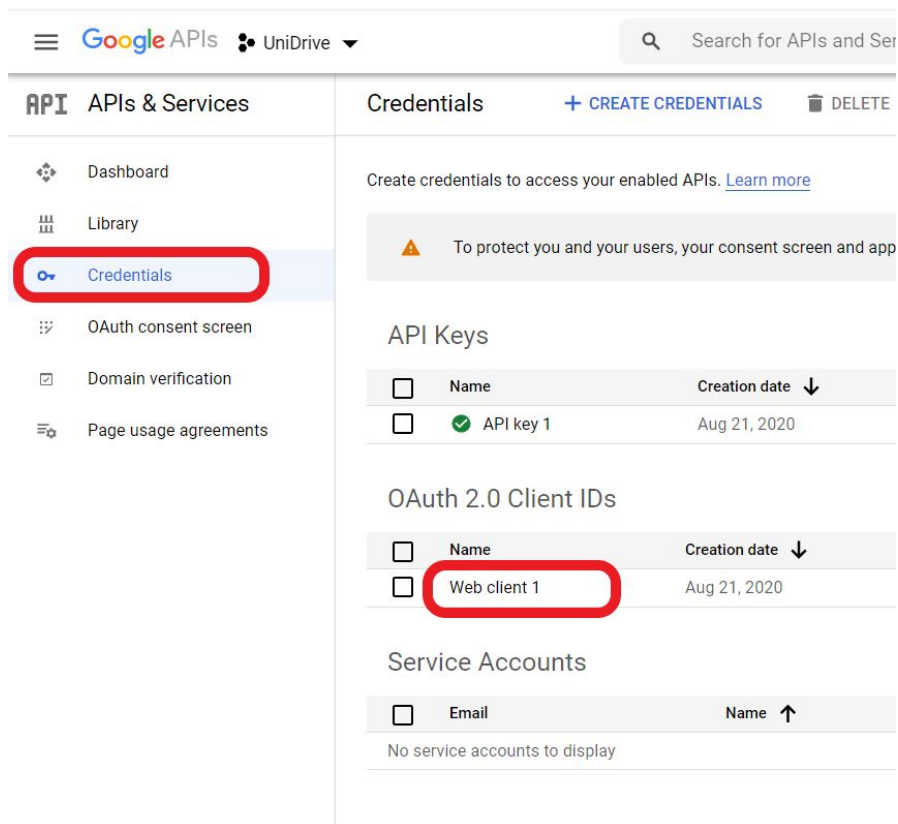
Variable	Value	Branch
REACT_APP_API_KEY	<input type="password" value="REACT_APP_API_KEY"/>	All branches
REACT_APP_CLIENT_ID	<input type="password" value="REACT_APP_CLIENT_ID.apps.googleusercontent.com"/>	All branches

- xviii. Return to the main page by **clicking UniDrive** on the side-bar.
- You can **test** to see if you've done everything correct so far **by clicking the link** at the center of the page.
 - It should take you to the UniDrive welcome screen, but there will be no functionality, yet. Simply **the welcome screen displayed indicates that you've done everything correct so far**.



c. Adding the AWS Amplify Build Link to the Google API Console

- i. Now, navigate back to the **Google developer console**. Select **credentials** from the sidebar and **click Web Client 1** (or the name you gave it). This is under the OAuth 2.0 Client IDs section.



- ii. **Click the first Add URI** (under authorized javascript origins) and now **paste the link** that was in the UniDrive Amazon Web Server Home Page (the same link you used to test it earlier). If desired, you can also add **localhost** as a link, so that the application works when developing/testing locally.

UniDrive

Search for APIs and Services

Client ID for Web application

DOWNLOAD JSON

RESET

The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

Authorized JavaScript origins

For use with requests from a browser

URIs

http://localhost:3000

https://unidrive. .amplifyapp.com

+ ADD URI

Authorized redirect URIs

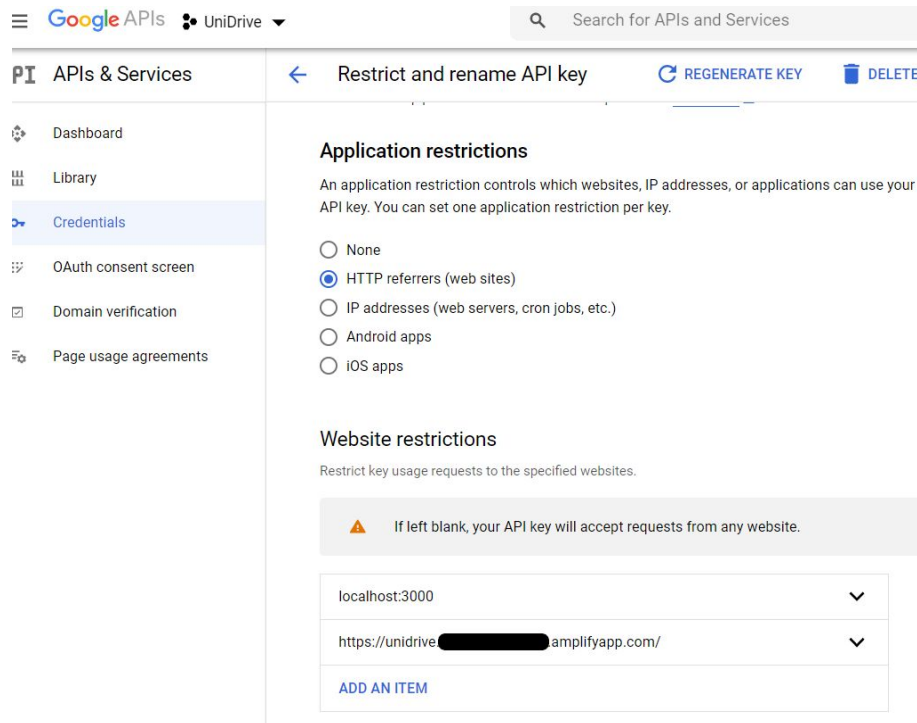
For use with requests from a web server

+ ADD URI

SAVE

CANCEL

- iii. Go back to the credentials home page, and select the API key. **Under Application Restrictions, click HTTP.** Then, **enter the same website links as before into the website restrictions.**



- iv. It may take around 5 minutes for the settings to take effect.
- v. Your application should now be set up and functioning. You should be able to see the application's main page. Clicking **add account** should now work.

6. Installation of Application

No steps need to be taken after the software is built. No directories are required for installation.

7. Run instructions (Local Development)

This is only needed if the user wants to develop the application locally.

To get the software to execute, the user/customer have to:

- a. Clone the Github Repository
- b. Go to master branch
- c. Get the API key and Client ID from <https://console.developers.google.com/>
 - i. If you don't have an API key or Client ID, follow the steps in **Section 3: Google Drive API Setup**
- d. Config file setup:
 - i. Create Config.js file in src folder
 - ii. Format the file as so. Replace the placeholder text with the necessary information. Do not remove the quotes.


```
export const config = {
  web: {
```



```
    api_key: "Paste API Key Here",  
    client_id: "Paste Client ID Here",  
  }  
}
```

- e. Running the application:
 - i. Type **npm install** in the terminal at the cloned repository directory.
 - ii. Type **npm start** to run the application in development mode.
- f. Open <http://localhost:3000> to view the application in your browser.

8. Troubleshooting

- a. AWS Troubleshooting
 - i. Incorrect environment variable names
 - 1. Ensure you are using the correct names for the variables as provided
 - a. **REACT_APP_API_KEY**
 - b. **REACT_APP_CLIENT_ID**
 - ii. Incorrect environment variable values
 - 1. Ensure you are pasting the exact value that Google lists for the API key and client ID
 - 2. For the client ID, it includes the **.app.googleusercontent.com** at the end
 - iii. Not referencing the AWS environment variable properly
 - 1. For every instance where you use values from the config file in the source code, you need to replace it with **process.env.AWS_Environment_Variable_Name**
 - 2. For an example, we use **process.env.REACT_APP_API_KEY** for the api key, and **process.env.REACT_APP_CLIENT_ID** for the client ID.
 - 3. If you are using the **unidrive branch** as listed in the AWS instructions, this is already done for you.
- b. Local Development Troubleshooting
 - i. Ensure you have installed all the dependencies with **npm install**