If you're not getting `id` and `token` from `req.query`, there might be a few reasons for this issue. Here are some things you can check:

1. \*\*URL Parameters in the Route Definition:\*\*

Ensure that the route definition in your Express application includes these parameters. For example, in your `routes.js` or wherever you define your routes, make sure you have something like this:

```javascript

// routes.js

const express = require('express');

const router = express.Router();

const emailVerificationController = require('./controllers/emailVerificationController');

// Define routes

router.get('/email/verify/:id/:token', emailVerificationController);

module.exports = router;

```

With this definition, you should be able to access `req.params.id` and `req.params.token` in your controller.

2. \*\*Check the URL Format:\*\*

Ensure that the URL you are using to make the request has the correct format with the `id` and `token` parameters. For example:

```

http://yourdomain.com/email/verify/123456789abcdef/yourtoken

```

3. \*\*Query Parameters Instead of Route Parameters:\*\*

If you intend to use query parameters (e.g., `/email/verify?id=123&token=yourtoken`), then update your route definition and access them using `req.query`:

```javascript

// routes.js

router.get('/email/verify', emailVerificationController);

```

In your controller:

```javascript

const { id, token } = req.query;

```

Adjust the route definition based on how you want to structure your URLs.

Double-check these aspects to ensure that the route parameters or query parameters are set up correctly and that you're using the correct URL format when making the request.

To extract an ID from a URL in a React application, you can use the `react-router-dom` library, which is commonly used for handling routing in React applications. Assuming you have a URL like `https://example.com/items/123`, where `123` is the ID you want to extract, here's an example of how you can achieve this:

1. First, make sure you have `react-router-dom` installed:

```bash

npm install react-router-dom

```

2. Use `BrowserRouter` and `Route` components from `react-router-dom` in your main application file (e.g., `App.js`):

```jsx

// App.js

import React from 'react';

import { BrowserRouter as Router, Route, Switch } from 'react-router-dom';

import ItemComponent from './ItemComponent';

function App() {

return (

<Router>

<Switch>

<Route path="/items/:id" component={ItemComponent} />

{/\* Other routes go here \*/}

</Switch>

</Router>

);

}

export default App;

```

3. Create the `ItemComponent` where you want to use the extracted ID:

```jsx

// ItemComponent.js

import React from 'react';

import { useParams } from 'react-router-dom';

function ItemComponent() {

const { id } = useParams();

return (

<div>

<h2>Item ID: {id}</h2>

{/\* Your component logic goes here \*/}

</div>

);

}

export default ItemComponent;

```

Now, when you navigate to a URL like `https://example.com/items/123`, the `ItemComponent` will receive the `id` as a parameter and you can use it as needed within that component.