



FPT POLYTECHNIC



THỰC HỌC – THỰC NGHIỆP



Conceive Design Implement Operate

LẬP TRÌNH FRONT-END FRAMEWORK 2

FORM & EVENTS

API REQUEST VỚI REACT

- ① Form & events
- ① API request với react





PHẦN 1

FORM & EVENTS

- ❑ Thêm form vào react giống như thêm bất kỳ phần tử nào. Ví dụ

```
import React from 'react';

class SearchBar extends React.Component {
  render() {
    return (
      <div>
        <form>
          <input type="text" />
        </form>
      </div>
    );
  }
}

export default SearchBar;
```

XỬ LÝ SỰ KIỆN (EVENTS)

Người dùng click (nhấp chuột)

onClick

Người dùng thay đổi nội dung
trong input

onChange

Người dùng gửi nội dung của form
(submit a form)

onSubmit

```
import React from 'react';

class SearchBar extends React.Component {
  onChange(event) {
    console.log(event.target.value);
  }

  render() {
    return (
      <div className="ui segment">
        <form className="ui form">
          <div className="field">
            <label>Image Search</label>
            <input type="text" onChange={this.onChange} />
          </div>
        </form>
      </div>
    );
  }
}

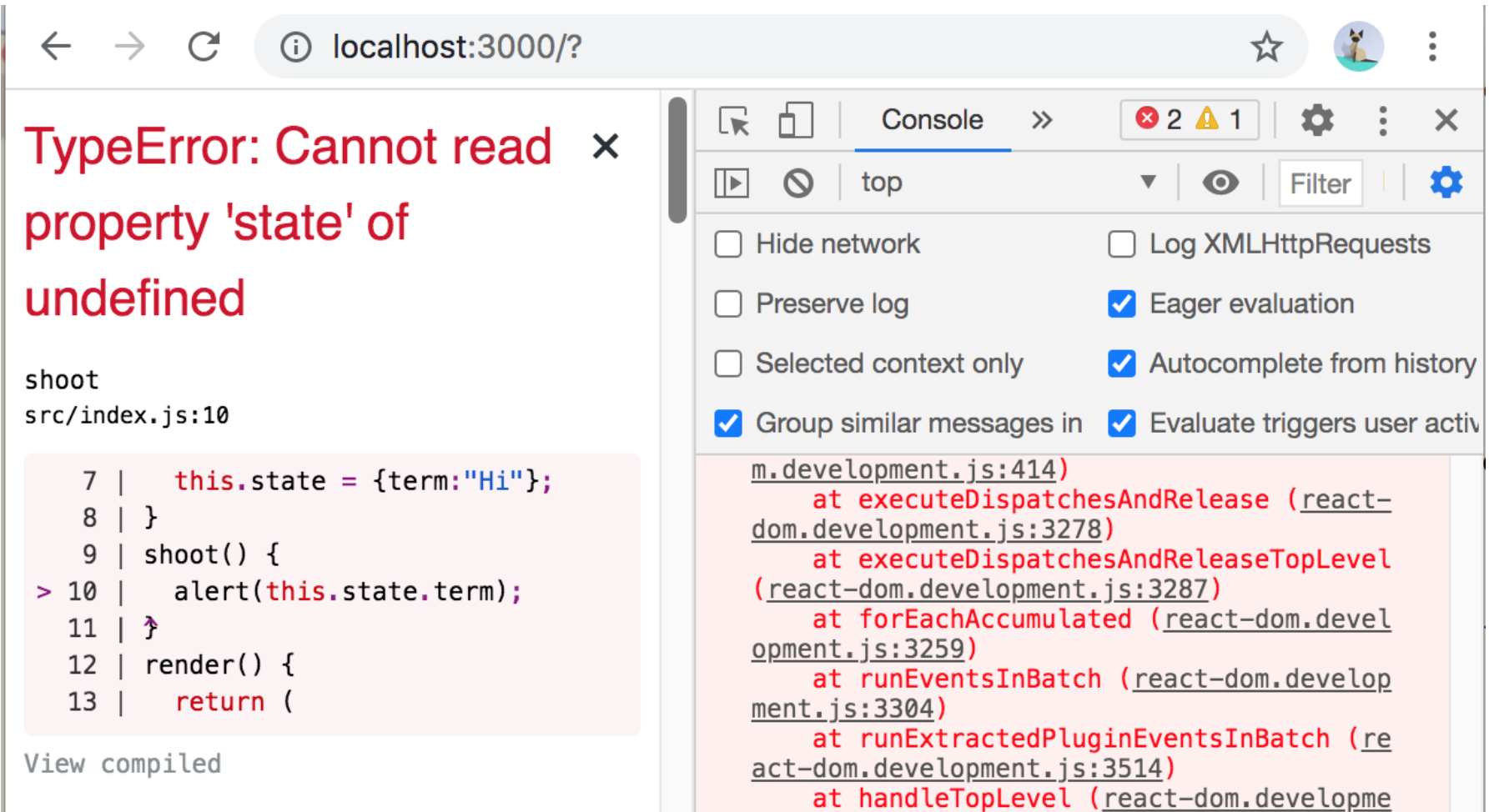
export default SearchBar;
```

□ Ví dụ

```
class Football extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {term: "Hi"};  
  }  
  shoot() {  
    alert(this.state.term);  
  }  
  render() {  
    return (  
      <button onClick={this.shoot}>Take the shot!</button>  
    );  
  }  
}
```

```
ReactDOM.render(<Football />, document.getElementById('root'));
```

❑ Kết quả



The screenshot shows a web browser at `localhost:3000/?` with a red error message: **TypeError: Cannot read property 'state' of undefined**. The error occurred at `src/index.js:10` in the `shoot` function. The code snippet shows `this.state` being accessed before `render()` is called.

The Chrome DevTools console is open, showing the error stack trace:

```
m.development.js:414
    at executeDispatchesAndRelease (react-dom.development.js:3278)
    at executeDispatchesAndReleaseTopLevel (react-dom.development.js:3287)
    at forEachAccumulated (react-dom.development.js:3259)
    at runEventsInBatch (react-dom.development.js:3304)
    at runExtractedPluginEventsInBatch (react-dom.development.js:3514)
    at handleTopLevel (react-dom.development.js:3514)
```


- ❑ Trong class component, **this** không được định nghĩa mặc định, vì vậy với các function thông thường, **this** không còn là đối tượng (object) (mà phương thức được định nghĩa).
- ❑ **Giải pháp 01**: Dùng arrow function
 - ❖ Một trong những tính năng đặc biệt của arrow function là tự động bind (ràng buộc) giá trị của this với tất cả code bên trong function

```
class Football extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {term:"Hi"};  
  }  
  shoot = () => {  
    console.log(this.state.term);  
  }  
  
  render() {  
    return (  
      <button onClick={this.shoot}>Take the shot!</button>  
    );  
  }  
}
```

this tham chiếu đến component object

< > ↻ ⓘ localhost:3000 ☆ 🐱 ⋮

Take the shot!

↶ ↷ Console >> ⚠️ 1 ⚙️ ⋮ ✕

▶ ⛔ top ▼ 👁 Filter ⚙️

☐ Hide network
 ☐ Log XMLHttpRequests

☐ Preserve log
 ☒ Eager evaluation

☐ Selected context only
 ☒ Autocomplete from history

☒ Group similar messages in
 ☒ Evaluate triggers user activ

[HMR] Waiting for update signal from WDS... log.js:24

⚠️ ▶ [Deprecation] scheduler.development.js:305
 SharedArrayBuffer will require cross-origin isolation as of M91, around May 2021. See <https://developer.chrome.com/blog/enabling-shared-array-buffer/> for more details.

react-dom.development.js:24994
Download the React DevTools for a better development experience: <https://fb.me/react-devtools>

2 Hi index.js:10

>

❑ **Giải pháp 02:** nếu bắt buộc phải dùng function thông thường thay cho arrow function, phải liên kết **this** đến component instance bằng phương thức **bind()**

```
class Football extends React.Component {  
  constructor(props) {  
    super(props);  
    this.shoot = this.shoot.bind(this);  
    this.state = {term:"Hi"};  
  }  
  shoot() {  
    console.log(this.state.term);  
  }  
  render() {  
    return (  
      <button onClick={this.shoot}>Take the shot!</button>  
    );  
  }  
}
```

- ❑ Uncontrolled input: giống như input HTML truyền thống.
- ❑ Ví dụ:

```
class Form extends Component {  
  render() {  
    return (  
      <div>  
        <input type="text" />  
      </div>  
    );  
  }  
}
```

❑ Để viết uncontrolled component

- ❖ Không cần viết xử lý sự kiện cho mỗi lần cập nhật state(ví dụ thay đổi giá trị trong input)
- ❖ Sử dụng **ref** để truy xuất giá trị của form từ DOM.

```
class Form extends React.Component {  
  constructor(props) {  
    super(props);  
    this.shoot = this.shoot.bind(this);  
    this.state = {term: 'Hi'};  
    this.name = React.createRef();  
  }  
  
  shoot () {  
    console.log(this.state.term);  
  }  
  
  render() {  
    return (  
      <input onChange={this.shoot} ref={this.name} />  
    );  
  }  
}
```

❑ Ref trong react có 2 bước

- ❖ 1. Tạo ref và gán nó cho 1 phần tử HTML
- ❖ 2. Tạo một tham chiếu đối tượng (reference object) đến ref

```
class Form extends React.Component {  
  constructor(props) {  
    super(props);  
    this.shoot = this.shoot.bind(this);  
    this.state = {term: 'Hi'};  
    this.name = React.createRef();  
  }  
  
  shoot () {  
    console.log(this.state.term);  
  }  
  
  render() {  
    return (  
      <input onChange={this.shoot} ref={this.name} />  
    );  
  }  
}
```

- ❑ Controlled component được liên kết đến 1 giá trị và những thay đổi của nó sẽ được xử lý bằng cách sử dụng lệnh callbacks.
- ❑ Ví dụ ở đây cho thấy input form và giá trị của nó được kiểm soát bởi React


```
class Form extends React.Component {
  constructor(props) {
    super(props);
    this.state = {value: ''};
  }

  handleChange = (event) => {
    this.setState({value: event.target.value});
  }





  render() {
    return (
      <label>
        Name:
        <input type="text" value={this.state.value} onChange={this.handleChange} />
      </label>
    );
  }
}

ReactDOM.render(<Form />, document.getElementById('root'));
```

EXPLORER ...

✓ OPEN EDITORS

✕ JS SearchBar.js src/...

✓ PICS 2    

 > public


 ✓ src

 ✓ components

 JS App.js

JS SearchBar.js

 JS index.js

 .gitignore

 {} package.json

i README.md

JS SearchBar.js ✕

src > components > JS SearchBar.js > ...

```

1  import React from 'react';
2
3  class SearchBar extends React.Component {
4    state = { term: '' };
5
6    onFormSubmit = event => {
7      event.preventDefault();
8
9      console.log(this.state.term);
10   };
11
12   render() {
13     return (
14       <div className="ui segment">
15         <form onSubmit={this.onFormSubmit} className="ui form">
16           <div className="field">
17             <label>Image Search</label>
18             <input
19               type="text"
20               value={this.state.term}
21               onChange={e => this.setState({ term: e.target.value })}
22             />
23           </div>
24         </form>
25       </div>
26     );
27   }
28 }
29
30 export default SearchBar;
31

```

EXPLORER ...

×

JS App.js src/compo...

✓

PICS 2

+

+

↺

📄

>

public

✓

src

✓

components

JS App.js

JS SearchBar.js

JS index.js

🔍 .gitignore

{ } package.json

📖 README.md

JS App.js ×

src > components > JS App.js > ...

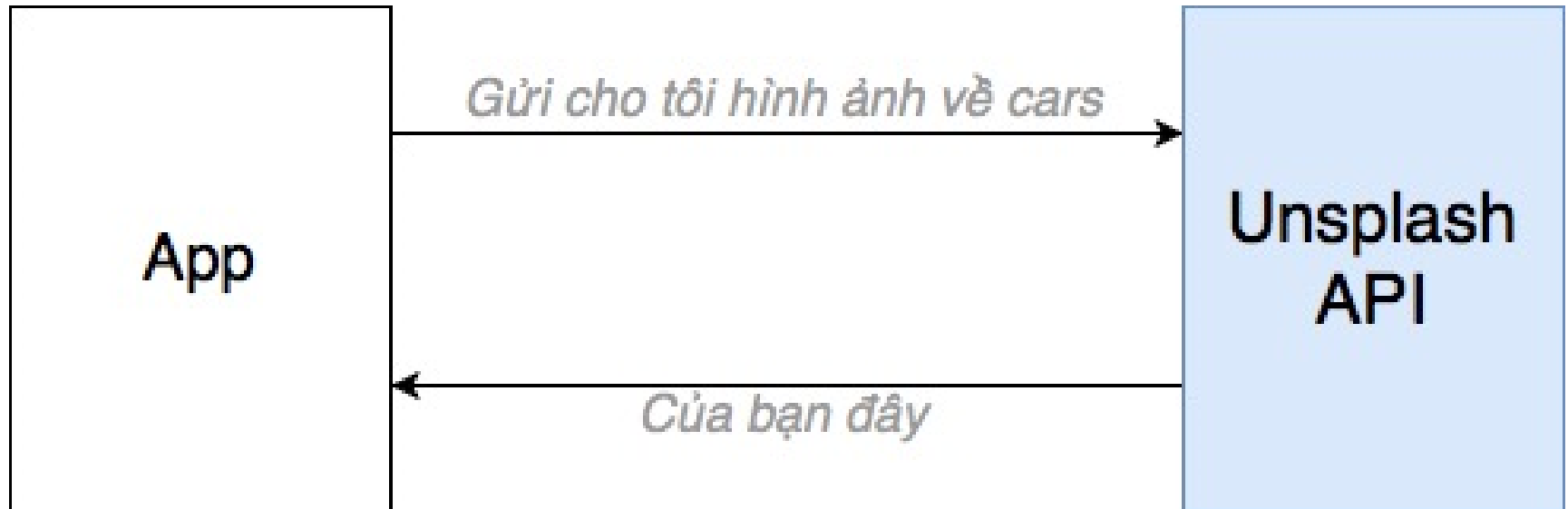
```

1  import React from 'react';
2  import SearchBar from './SearchBar';
3
4  const App = () => {
5    return (
6      <div className="ui container" style={{ marginTop: '10px' }}>
7        <SearchBar />
8      </div>
9    );
10 };
11
12 export default App;
13
```



PHẦN 2

API REQUEST VỚI REACT



- ❑ Sử dụng unsplash API: truy cập <https://unsplash.com/developers> và đăng ký với tư cách developers
- ❑ Tạo app > lưu access key và secret key
- ❑ Vào documentation


[List a user's collections](#)
[Get a user's statistics](#)

Photos

[List photos](#)
[Get a photo](#)
[Get a random photo](#)
[Get a photo's statistics](#)
[Track a photo download](#)
[Update a photo](#)
[Like a photo](#)
[Unlike a photo](#)

Search

[Search photos](#)
[Search collections](#)
[Search users](#)

Collections

[List collections](#)
[Get a collection](#)
[Get a collection's photos](#)
[List a collection's related collections](#)
[Create a new collection](#)
[Update an existing collection](#)

Search photos

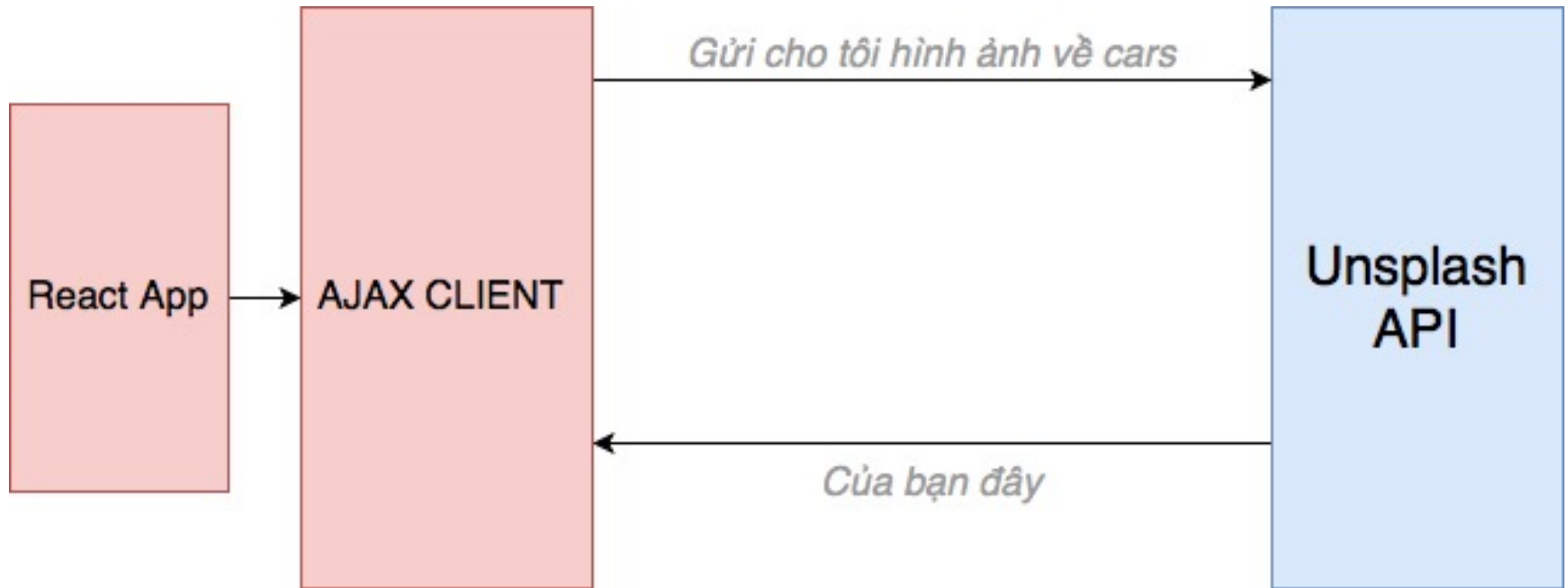
Get a single page of photo results for a query.

```
GET /search/photos
```

Note: See the note on [hotlinking](#).

Parameters

param	Description
query	Search terms.
page	Page number to retrieve. (Optional; default: 1)
per_page	Number of items per page. (Optional; default: 10)
order_by	How to sort the photos. (Optional; default: <code>relevant</code>). Valid values are <code>latest</code> and <code>relevant</code> .
collections	Collection ID('s) to narrow search. Optional. If multiple, comma-separated.
content_filter	Limit results by content safety . (Optional; default: <code>low</code>). Valid values are <code>low</code> and <code>high</code> .
color	Filter results by color. Optional. Valid values are: <code>black_and_white</code> , <code>black</code> , <code>white</code> , <code>yellow</code> , <code>orange</code> , <code>red</code> , <code>purple</code> , <code>magenta</code> , <code>green</code> , <code>teal</code> , and <code>blue</code> .
orientation	Filter by photo orientation. Optional. (Valid values: <code>landscape</code> , <code>portrait</code> , <code>squarish</code>)



Axios

Gói thư viện bên thứ 3

Fetch

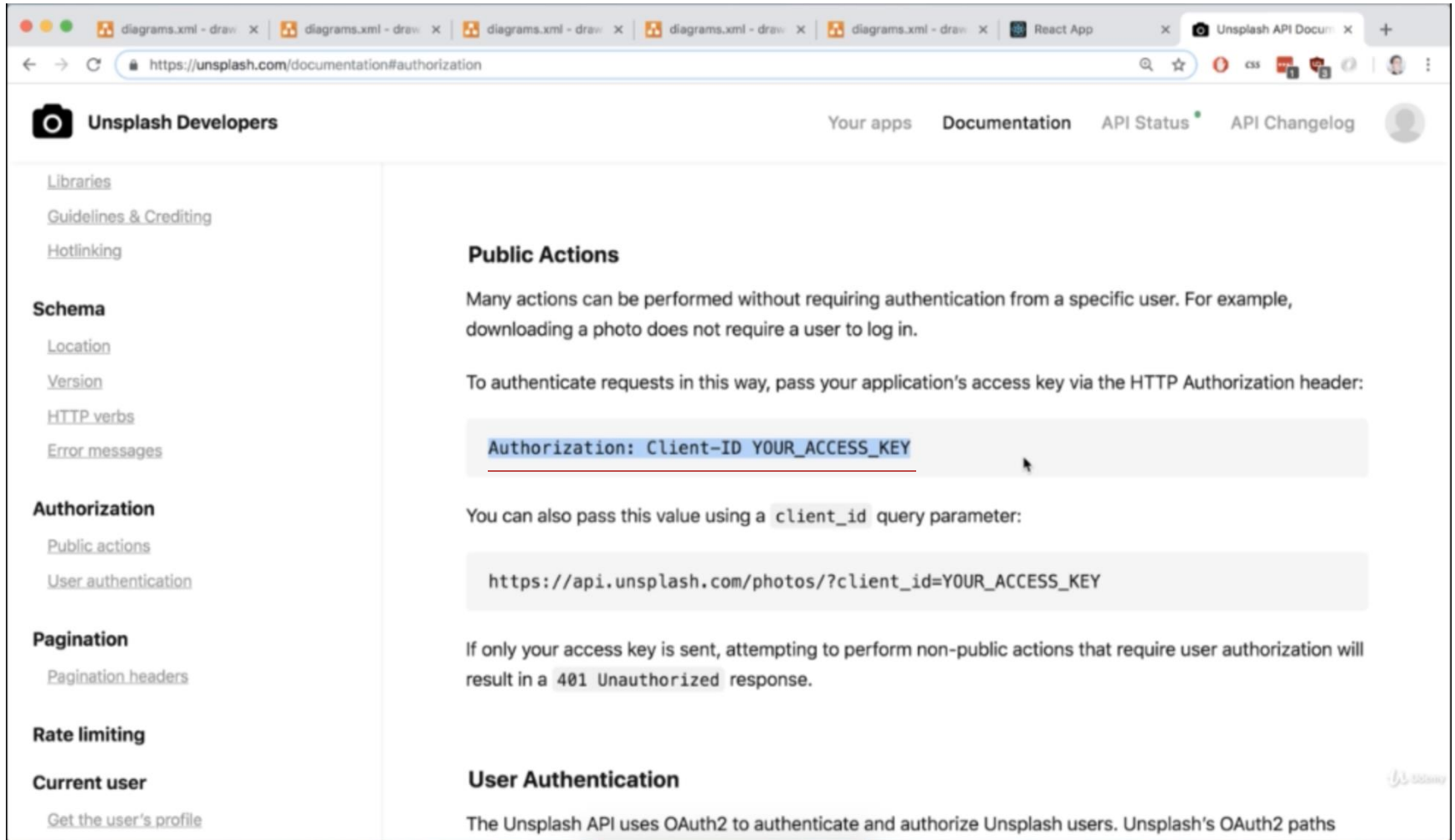
Hàm được tích hợp
sẵn trong browser

❑ Cài đặt

```
→ pics git:(092-invoke) npm install --save axios
npm WARN ajv-keywords@3.2.0 requires a peer of ajv^6.0.0 but none is
installed. You must install peer dependencies yourself.

+ axios@0.18.0
added 1 package from 1 contributor and audited 35654 packages in 8.5
41s
found 0 vulnerabilities

→ pics git:(092-invoke) x █
```



The screenshot shows a web browser window with the URL `https://unsplash.com/documentation#authorization`. The page is titled "Unsplash Developers" and has a navigation bar with links: "Your apps", "Documentation", "API Status", and "API Changelog". A sidebar on the left contains links for "Libraries", "Guidelines & Crediting", "Hotlinking", "Schema" (with sub-links for "Location", "Version", "HTTP verbs", and "Error messages"), "Authorization" (with sub-links for "Public actions" and "User authentication"), "Pagination" (with sub-link "Pagination headers"), "Rate limiting", and "Current user" (with sub-link "Get the user's profile").

The main content area is titled "Public Actions" and contains the following text:

Many actions can be performed without requiring authentication from a specific user. For example, downloading a photo does not require a user to log in.

To authenticate requests in this way, pass your application's access key via the HTTP Authorization header:

```
Authorization: Client-ID YOUR_ACCESS_KEY
```

You can also pass this value using a `client_id` query parameter:

```
https://api.unsplash.com/photos/?client_id=YOUR_ACCESS_KEY
```

If only your access key is sent, attempting to perform non-public actions that require user authorization will result in a `401 Unauthorized` response.

The section "User Authentication" is partially visible at the bottom, with the text: "The Unsplash API uses OAuth2 to authenticate and authorize Unsplash users. Unsplash's OAuth2 paths".

EXPLORER

...

JS SearchBar.js

JS App.js

JS index.js

OPEN EDITORS

JS SearchBar.js src/...

JS App.js src/compo...

JS index.js src

PICS

public

src

components

JS App.js

JS SearchBar.js

JS index.js

.gitignore

package.json

README.md

src > components > JS App.js > ...

```

1  import React from 'react';
2  import axios from 'axios';
3  import SearchBar from './SearchBar';
4
5  class App extends React.Component {
6    onSearchSubmit(term) {
7      axios.get('https://api.unsplash.com/search/photos', {
8        params: { query: term },
9        headers: {
10          Authorization:
11            'Client-ID 2b98c1afb0aed3b3d94a1866bdc3ac013d21a0c86d236a0fee32355c331c
12        },
13      });
14    }
15
16    render() {
17      return (
18        <div className="ui container" style={{ marginTop: '10px' }}>
19          <SearchBar onSubmit={this.onSearchSubmit} />
20        </div>
21      );
22    }
23  }

```

Component render 1 lần (không có danh sách hình ảnh)

onSearchSubmit được gọi

Request đến unsplash

.....chờ.....

Request hoàn thành

Thiết lập dữ liệu hình ảnh trên state của App component

App component re-render và hiển thị hình ảnh

```
async onSearchSubmit(term) {  
  const response = await axios.get('https://api.unsplash.com/search/photos', {  
    params: { query: term },  
    headers: {  
      Authorization:  
        'Client-ID 2b98c1afb0aed3b3d94a1866bdc3ac013d21a0c86d236a0fee32355c331c0296',  
    },  
  });  
  
  console.log(response.data.results);  
}
```

❑ Thiết lập state

```
class App extends React.Component {  
  state = { images: [] };  
  
  async onSearchSubmit(term) {  
    const response = await axios.get('https://api.unsplash.com/search/photos', { ...  
    });  
  
    console.log(this);  
    this.setState({ images: response.data.results });  
  }  
  
  render() {  
    return (  
      <div className="ui container" style={{ marginTop: '10px' }}>  
        <SearchBar  
          onSubmit={this.onSearchSubmit}  
          guesswhtIam="Im the propsobject"  
        />  
        Found: {this.state.images.length} images  
      </div>  
    );  
  }  
}
```

❑ Binding callback với arrow function

```
onSearchSubmit = async (term) => {  
  const response = await axios.get('https://api.unsplash.com/search/photos', {  
    params: { query: term },  
    headers: {  
      Authorization:  
        'Client-ID 2b98c1afb0aed3b3d94a1866bdc3ac013d21a0c86d236a0fee32355c331c0296',  
    },  
  });  
  
  this.setState({ images: response.data.results });  
};  
  
render() {  
  return (  
    <div className="ui container" style={{ marginTop: '10px' }}>  
      <SearchBar onSubmit={this.onSearchSubmit} />  
      Found: {this.state.images.length} images  
    </div>  
  );  
}
```



APP – SEARCH PICS

thank
you!