



```
{
  "urls": {
    "raw": "https://images.unsplash.com/photo-1461988320302-91bde64fc8e4?ixid=2yJhcHBfaWQiOjE9MDd9",
    "full": "https://images.unsplash.com/photo-1461988320302-91bde64fc8e4?ixid=2yJhcHBfaWQiOjE9MDd9&fm=jpg&q=80",
    "regular": "https://images.unsplash.com/photo-1461988320302-91bde64fc8e4?ixid=2yJhcHBfaWQiOjE9MDd9&fm=jpg&fit=crop&w=1080&q=80&fit=max",
    "small": "https://images.unsplash.com/photo-1461988320302-91bde64fc8e4?ixid=2yJhcHBfaWQiOjE9MDd9&&fm=jpg&w=400&fit=max",
    "thumb": "https://images.unsplash.com/photo-1461988320302-91bde64fc8e4?ixid=2yJhcHBfaWQiOjE9MDd9&fm=jpg&w=200&fit=max"
  },
  // ... other photo fields
}
```


Axios

3rd Party

Fetch

Function bawaan yang
sudah ada di javascript

<https://jsonplaceholder.typicode.com/>

ubah file src > App.js

Alternatif 1

```
class App extends React.Component {  
  state = { images : [] };  
  
  onSearchSubmit(term){  
    fetch(`https://jsonplaceholder.typicode.com/photos?id=${term}`, {  
      method: 'GET'  
    })  
    .then(res => res.json())  
    .then((data) => {  
      this.setState({images: data});  
    })  
  }  
}
```

Alternatif 2

```
class App extends React.Component {  
  state = { images : [] };  
  
  onSearchSubmit = async (term) => {  
    const response = await fetch(`https://jsonplaceholder.typicode.com/photos?id=${term}`, {  
      method: 'GET'  
    });  
  
    const data = await response.json();  
    this.setState({images: data});  
  };
```


Component render

onSearchSubmit method dipanggil

Request ke API Server

Menunggu respon ...

Request komplit

setState data yang didapat dari API

Component re-render

ubah file src > App.js

```
render() {  
  return (  
    <div className="container mt-3">  
      <SearchBar onSubmit={this.onSearchSubmit}/>  
      Found: {this.state.images.length} images  
    </div>  
  )  
};  
}  
...
```


Buat file baru src > ImageList.js

```
import React from 'react';

const ImageList = (props) => {
  const imgs = props.images.map((image) => {
    return (
      <img src={image.url} alt='No Image' />
    )
  });

  return (
    <div>
      {imgs}
    </div>
  )
}

export default ImageList;
```

Update file src > App.js

```
...
render() {
  return (
    <div className="container mt-3">
      <SearchBar onSubmit={this.onSearchSubmit}/>
      Found : {this.state.images.length}
      <ImageList images={this.state.images}/>
    </div>
  )
}
...
```


create-react-app layanan-belakang

kosongkan isi src, buat file index.js untuk render DOM, App.js untuk komponen entry point

di App.js

```
import {getUserInfo} from "../api/user";
...
state = {userInfo: null, userFullName: "", address: ""};

onGetLayananCepat = async () => {
  const response = await getUserInfo();
  const data = await response.json();
  this.setState({userInfo: data});
};

componentDidMount() {
  this.onGetLayananCepat();
}

onRenderInfo = () => {
  const {userInfo} = this.state;
  if (userInfo) {
    return (
      <div>
        {userInfo.namaLengkap} {userInfo.alamat}
      </div>
    )
  } else {
```

buat file src > api > user > index.js

```
export function getUserInfo() {
  return fetch('http://localhost:6969/user', {
    method: 'GET'
  });
}
```

Access to fetch at 'http://localhost:6969/' from origin 'http://localhost:3000' has been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the requested resource. If an opaque response serves your needs, set the request's mode to 'no-cors' to fetch the resource with CORS disabled.

```
} else {  
  return <div></div>  
}
```

```
}
```

```
render() {  
  return (  
    <div>  
      {this.onRenderInfo()}  
    </div>  
  )  
}
```



```
buat file root_project > .env.development
```

```
REACT_APP_WS_URL=http://localhost:3000
```

```
buat file src > api> user > index.js
```

```
export function getUserInfo() {  
  return fetch(process.env.REACT_APP_WS_URL + '/user', {  
    method: 'GET'  
  });  
}
```

```
ubah package.json
```

```
...  
"proxy": "http://localhost:6969"  
...
```

