useFlask()

useFlask()

…or how to use a ⚛ React frontend for your Flask app

ABOUT ME

Adrian Mönnich **@ThiefMaster**

- Lead Developer of the Indico project at CERN
- One of the Pallets maintainers
- Python enthusiast since almost 10 years
- Moved on from jQuery to React

STATUS QUO

Based on what people ask questions about on IRC

- WTForms
- jQuery
- AJAX
- JavaScript

WTFORMS

Easy for simple cases 🎉

WTFORMS

Easy for simple cases 🎉

Complex custom widgets? Welcome to WTF-Forms 🐒

- Value round-trip even if validation fails!
- Garbage-in, garbage-out? 🗑

JQUERY

Do YOU still want to use it? 😜

JQUERY

Do YOU still want to use it? 😜

- Spaghetti code
- Classes/IDs all over the code
- Barely maintainable

AJAX

XMLHttpRequest API is horrible

fetch() isn't bad

Are you sending back large chunks of HTML?

Duplicating code for your API?

JAVASCRIPT

Rarely transpiled (sometimes minified)

Code written for old browsers 🄮 🧑

Why are you still writing the same kind of JS you wrote 10 years ago?

HOW TO IMPROVE IT?

- Build asset bundle with Webpack 🔷
- Use modern JavaScript (ES6+, ES201x) ✨
- Make the frontend reactive ⚛️

> *"Configuring webpack is too hard"*

No, it's not. Unless you do weird things.
Been there, done that.

But it *is* boilerplate code.

Can't we do something like…

`from webpack import config`

Almost!

It's called create-react-app

Create React App is an officially supported way to create single-page React applications. It offers a modern build setup with no configuration.

CREATE-REACT-APP

Contains the whole build config

Auto-rebuild 🏗 & hot reload 🔥

Includes its own dev server

LET'S CODE!
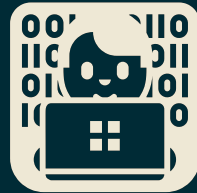
OUR FLASK APP

```python
app = Flask(__name__)


@app.route('/api/time')
def time():
    return jsonify(now=datetime.now().isoformat())



@app.route('/api/greet/<name>')
@app.route('/api/greet-stranger/', defaults={'name': 'mysterious person'})
def greeting(name):
    msg = 'Welcome, ' + name
    return jsonify(greeting=msg)
```

GENERATE THE CLIENT

```
$ npx create-react-app client
Creating a new React app in …/ep2019-flask-react/testapp/client.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts...

…

We suggest that you begin by typing:

  cd client
  npm start

Happy hacking!
```

EXPORT ROUTES TO JS

Hardcoded URLs are ugly! 👎

Especially if they are dynamic 😠

Install some more packages

```
$ pip install flask-url-map-serializer
$ npm install --save-dev flask-urls.macro
$ npm install --save flask-urls
```

Hook up the build scripts with Flask

```js
// client/.babel-plugin-macrosrc.js
const {execSync} = require('child_process');
const urlMap = JSON.parse(execSync('flask url_map_to_json'));

module.exports = {
  flaskURLs: {
    urlMap
  }
};
```

🔗 Link the dev servers

```javascript
// client/src/setupProxy.js
const proxy = require('http-proxy-middleware');
module.exports = app => {
  app.use(proxy('/api', {
    target: process.env.FLASK_URL || 'http://127.0.0.1:5000'
  }));
};
```

RECAP

We now have:

- Our Flask app providing an API
- Boilerplate for a React frontend
- The ability to build Flask URLs in JS
- The CRA dev server forwarding /api/* to Flask

```
 1  // client/src/App.js
 2  // …
 3  import Demo from './Demo';
 4
 5  function App() {
 6    return (
 7      {/* … */}
 8      <header classname="App-header">
 9        <img src={logo} className="App-logo" alt="logo" />
10        <Demo />
11      </header>
12      {/* … */}
13    );
14  }
```

```
1  // client/src/Demo.js
2  import React, {useState, useEffect} from 'react';
3  import flask from 'flask-urls.macro';
4
5  const timeURL = flask`time`;
6  const greetingURL = flask`greeting`;
7
8  export default function Demo() {
9    // continued on next slide…
10 }
```

```
1  const [time, setTime] = useState(null);
2  useEffect(() => {
3    (async () => {
4      const resp = await fetch(timeURL());
5      const data = await resp.json();
6      setTime(data.now);
7    })();
8  }, []);
9
10 const url = greetingURL({name: 'snake'});
11 return time && (
12   <p>It's {time} and your greeting URL is {url}</p>
13 );
```

How the hell does this all work?! 🧌

How the hell does this all work?! 🤷

How the hell does this all work?! 👺


```javascript
// Babel macro - runs at build time!
import flask from 'flask-urls.macro';
// Tagged template compiles to function
const greetingURL = flask`greeting`;
// equivalent of url_for('greeting', name='snake')
const url = greetingURL({name: 'snake'});
```

How the hell does this all work?! 🧙‍♀️


```javascript
// Babel macro - runs at build time!
import flask from 'flask-urls.macro';
// Tagged template compiles to function
const greetingURL = flask`greeting`;
// equivalent of url_for('greeting', name='snake')
const url = greetingURL({name: 'snake'});
```

How the hell does this all work?! 🧛


```javascript
// Babel macro - runs at build time!
import flask from 'flask-urls.macro';
// Tagged template compiles to function
const greetingURL = flask`greeting`;
// equivalent of url_for('greeting', name='snake')
const url = greetingURL({name: 'snake'});
```

That's the code generated by the macro

```
const greetingURL = flask_urls__WEBPACK_IMPORTED_MODULE_1___default.a.bind(
  null,
  {
    endpoint: "greeting",
    rules: [/* …building rules omitted here… */]
  },
  ""
);
```

YOU CAN CONTRIBUTE!

 indico/js-flask-urls

We could really use...

📜 Docs (only examples right now 😣)

✨ TypeScript stubs

🖥 More projects using it (and finding bugs? 🐛)

→ Brag about having contributed to a CERN project! 🎉

OTHER COOL STUFF

Non-macro Babel plugin for Flask URL building

```
import greetingURL from 'flask-url:greeting';
import otherURL from 'flask-url:blueprint.endpoint';
```

Same features as the macro version

Alternative to the macro, but requires control over the Webpack/Babel config

Server-side validation: webargs

Form handling & validation: 🏁 react-final-form

More complete example of React-Flask integration

[GitHub] ThiefMaster/flask-cra-example

Also covers how to build for production

FIND ME ON…

 @ThiefMaster

ThiefMaster @ Freenode (#pocoo, #indico)

FIND ME ON…

 @ThiefMaster

\# ThiefMaster @ Freenode (#pocoo, #indico)

…or right here!

I have hexagonal stickers!

…and my team at CERN is looking for a student intern!

indico
GETINDICO.IO   MADE @ CERN