Serving a create-react-app with Flask

Asked 2 years, 2 months ago   Active 4 months ago   Viewed 13k times

▲

45

▼

★

17

I have a flask back-end with API routes which are accessed by a React single page application, created using the create-react-app boilerplate. When using the create-react-app built-in dev server, my Flask back end works, no problem with that.

Now, I would like to serve the built (using `npm run build`) static react app from my Flask server. Building the react app leads to the following directory structure:

```
- build
  - static
    - css
        - style.[crypto].css
        - style.[crypto].css.map
    - js
        - main.[crypto].js
        - main.[crypto].js.map
  - index.html
  - service-worker.js
  - [more meta files]
```

By [crypto], I mean the randomly generated strings generated at build time.

Having recieved the `index.html` file, the browser then makes the following requests:

```
- GET /static/css/main.[crypto].css
- GET /static/css/main.[crypto].css
- GET /service-worker.js
```

My question is then: how should I go about serving these files ? I came up with this:

```python
from flask import Blueprint, send_from_directory

static = Blueprint('static', __name__)

@static.route('/')
def serve_static_index():
    return send_from_directory('../client/build/', 'index.html')

@static.route('/static/<path:path>') # serve whatever the client requested in the static
folder
def serve_static(path):
    return send_from_directory('../client/build/static/', path)

@static.route('/service-worker.js')
def serve_worker():
    return send_from_directory('../client/build/', 'service-worker.js')
```

This way, the static assets are successfully served. But it is not a very elegant solution.

On the other hand, I could incorporate this with the built-in flask static utilities. But I do not understand how to configure this.

I really do not know how to handle this, to the point that it's making me reconsider my use of create-react-app, as it is forcing me to structure my static folder in a very specific an inconvenient way: There is no way for me to change how the app requests static content from the server.

Overall: Is my solution robust enough ? Is there a way to use built in flask features to serve these assets ? Is there a better way to use create-react-app ? Any input is appreciated. I can provide more information if needed.

Thanks for reading !

python   python-3.x   reactjs   flask   create-react-app

asked May 26 '17 at 20:54

Theo
363   1   4   13

---

3   flask should know about your static folder without you having to do anything (as long as the folder is named static and is next to your flask entrypoint) ... ie `cp -rf /build/static ./static` as part of your build script ... – Joran Beasley May 26 '17 at 21:06 ✎

---

1   you can also use nginx to serve your static files, which is usually the recommended via (nginx is super good for static files) – patrick Jun 28 '17 at 15:53

---

2 Answers

50

```python
import os
from flask import Flask, send_from_directory

app = Flask(__name__, static_folder='react_app/build')

# Serve React App
@app.route('/', defaults={'path': ''})
@app.route('/<path:path>')
def serve(path):
    if path != "" and os.path.exists(app.static_folder + path):
        return send_from_directory(app.static_folder, path)
    else:
        return send_from_directory(app.static_folder, 'index.html')


if __name__ == '__main__':
    app.run(use_reloader=True, port=5000, threaded=True)
```

Thats what I ended up with. So bascially catch all routes, test if the path is a file => send file => else send the index.html. That way you can reload the react app from any route you wish and it does not break.

edited Apr 13 at 14:52

answered Aug 11 '17 at 11:51

Jodo
1,767   1   20   34

This works. Thanks Jodo. – Xinyang Li Aug 21 '17 at 3:00

2   I'm getting mime type errors with this solution :-(`The script has an unsupported MIME type ('text/html'). /service-worker.js Failed to load resource: net::ERR_INSECURE_RESPONSE registerServiceWorker.js:71 Error during service worker registration: DOMException: Failed to register a ServiceWorker: The script has an unsupported MIME type ('text/html').` – ketysek Mar 8 '18 at 22:55

@user3216673 it most likely is not an issue with flask or create-react-app but with your Browser. A guess in the wild: unregister your service workers might resolve the issue. – Jodo May 4 '18 at 6:54

1   You can use app.static_folder to keep your code DRY – Eytan Apr 13 at 12:38

thanks a lot , it works – Mehmet nuri Jul 24 at 9:43

First do `npm run build` to build the static production files as mentioned by you above

5

```python
from flask import Flask, render_template

app = Flask(__name__, static_folder="build/static", template_folder="build")

@app.route("/")
def hello():
    return render_template('index.html')

print('Starting Flask!')

app.debug=True
app.run(host='0.0.0.0')
```

Unfortunately, I don't think you can get it work with the development hot-reload.

answered Jan 11 at 3:22

Pranay Aryal
2,977   2   19   29