

How to add React JS into Django

I've been looking for a great front-end library to fit into Django for years. The problem is that UI requirements have been growing so tremendously that the conventional tech stack (Django template engine + Bootstrap + Vanilla JavaScript) does not meet the pace of modern web development anymore. After I have been researching and using different libraries in many projects, JQuery, React JS and Vue JS seems to be the top-tier choices. Among them, React JS definitely catches my eyes. Leveraging its component-based development, we can scale our UIs in a time-efficient manner with minimum maintenance effort. On top of that, the React community has also been very innovative in terms of all kinds of solutions on UI developments.

Unlike the most tutorials which use React/Node as web framework and Django as REST service, I am going to use Django as web framework and React as a pure front-end solution. In this article, we'll walk you through how to enable React component in Django by a simple example.

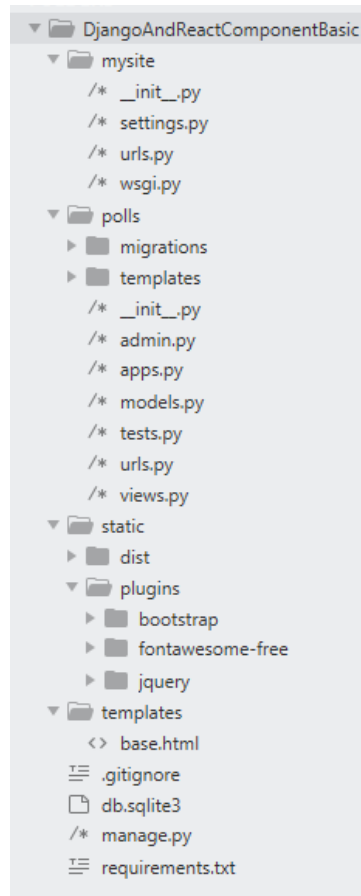
Dependencies:

1. Python 3.7
2. Django 2.2.x
3. React 17
4. AdminLTE 3.0.5 (see my previous tutorial [here](#) for this theming library)

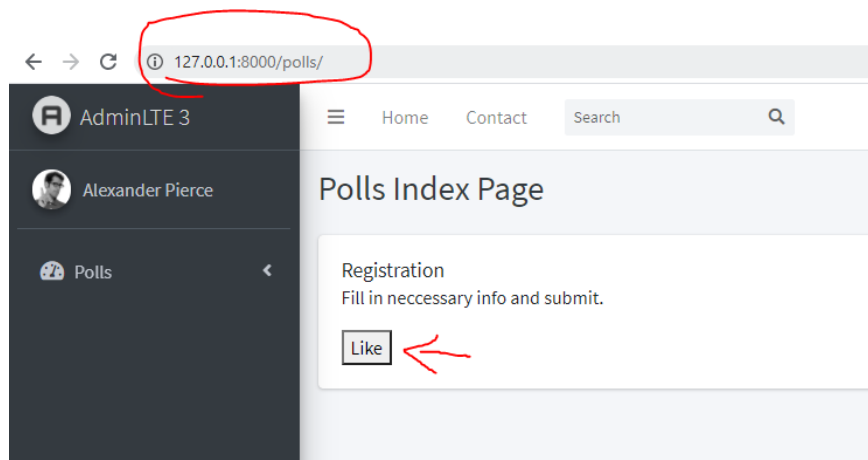
My approaches:

Step 1/5: Preview

We will start our tutorial from the project that looks like below.



In the end of tutorial, we would add a React “Like” button in page **127.0.0.1:8000/polls/** in **polls** app.

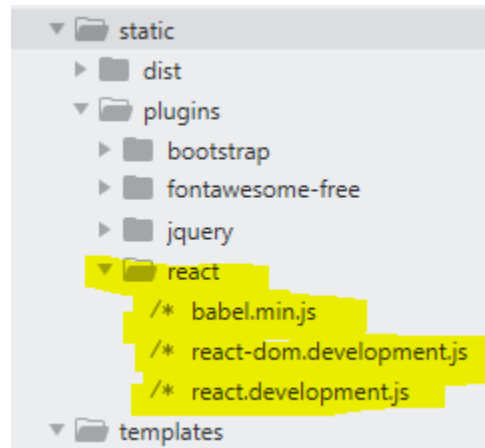


Step 2/5: Add React and JSX libraries

Create a new folder **react** in **/static/plugins/** folder. In **react** folder we just created, create three JavaScript file and paste over the content from each of the following pages.

- <https://unpkg.com/react@17/umd/react.development.js>
- <https://unpkg.com/react-dom@17/umd/react-dom.development.js>
- <https://unpkg.com/babel-standalone@6/babel.min.js>

Then, we would expect the result as below.



Step 3/5: Import libraries in template

In this project, each page uses **base.html** as a base template. Thus, we will import libraries in **base.html** file to make them available to all child templates. Import them at the end. And the result is as below.

```
<script src="{% static 'plugins/react/react.development.js' %}"></script>
<script src="{% static 'plugins/react/react-dom.development.js' %}"></script>
<!-- JSX support-->
<script src="{% static 'plugins/react/babel.min.js' %}"></script>
```

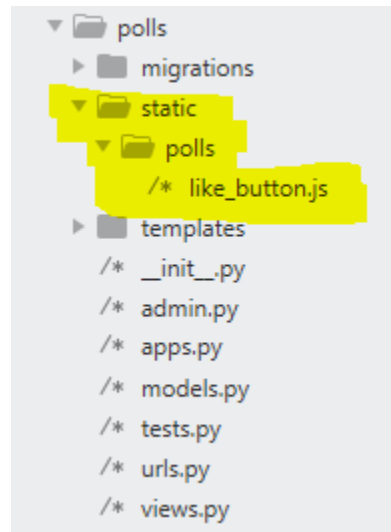
Step 4/5: Create container for “like” button

Let us add a division in **127.0.0.1:8000/polls/** page where button will be added. It is as simple as just one line of code. Make sure id is declared.

```
<div id="like_button_container"></div>
```

Step 5/5: Create React component

Create a folder **static** in **polls** app. Within **static** folder we just created, create a folder **polls**. Then we can create a file **like_button.js** in polls folder. The schema would look like below.



Copy and paste following JSX code in **like_button.js** file.

```
'use strict';

const e = React.createElement;

class LikeButton extends React.Component {
  constructor(props) {
    super(props);
    this.state = { liked: false };
  }

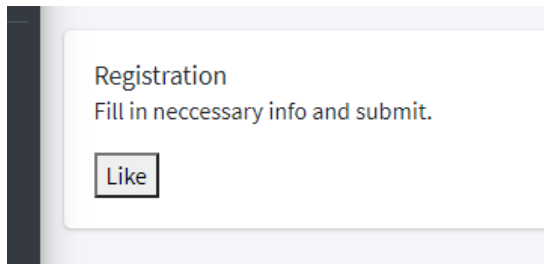
  render() {
    if (this.state.liked) {
      return 'You liked this.';
    }

    return (
      <button onClick={() => this.setState({ liked: true })}>
        Like
      </button>
    );
  }
}

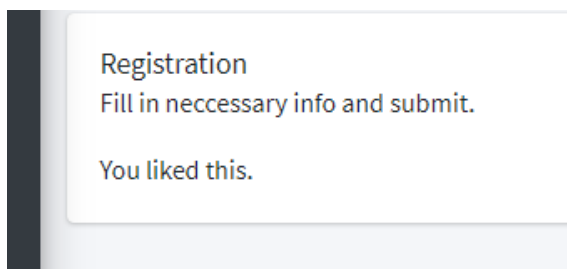
const domContainer = document.querySelector('#like_button_container');
ReactDOM.render(<LikeButton />, domContainer);
```

Let's start project and review what we have achieved. We would expect the button being rendered in "like_button_container" division. After we clicked button, it will be transformed to "You liked this." text.

Before.



After.



Conclusion

Hope this tutorial gives you a hint. Make sure you use these libraries under production environment.

- <https://unpkg.com/react@17.0.1/umd/react.production.min.js>
- <https://unpkg.com/react-dom@17.0.1/umd/react-dom.production.min.js>

In this article, I didn't get a chance to cover how you can connect React component to Django view. If you are interested, please leave your comments. I've been very busy recently. But I will definitely make a new article on it.

I've uploaded this project in git. Repo name is DjangoAndReactComponentBasic. Check out this project [here](#).

<https://github.com/slow999/DjangoAndReactComponentBasic>

I've made a video on it. Check out it [here](#).

<https://youtu.be/irTg8rNUKb8>