

# React + Material-UI + Truffle Box

Rapid Ethereum Dapp Development

This box contains everything you need to get started using smart contracts from a React app with the Material-ui library.

## Installation

First, create a new empty directory and go to it.

1. Run `theunbox`
2. command `via npx`
3. and skip to step 3. This will install all necessary dependencies. A React app is generated at the root
4. of the directory.
5. `npx`
6. `truffle`
7. `unbox`
8. `rouftom`
9. `/`
10. `truffle`
11. `-`
12. `react`
13. `-`
14. `material`
15. Alternatively, you can install Truffle globally and run `theunbox`
16. command.
17. `npm`
18. `install`
19. `-`
20. `g`
21. `truffle`
22. `truffle`
23. `unbox`
24. `rouftom`
25. `/`
26. `truffle`
27. `-`
28. `react`
29. `-`
30. `material`
31. Run the development console.
32. `truffle`
33. `develop`
34. Compile and migrate the smart contracts. Note inside the development console we don't preface commands with `truffle`
35. `.`
36. `compile`
37. `migrate`
38. If you want to migrate your smart contract with a fresh blockchain state, you can run the command below.  
`truffle migrate --reset`
39. The `--reset`
40. flag purge the blockchain state. Don't run the command with this flag in production unless you know what you are doing.
41. In the client
42. directory, we run the React app. Smart contract changes must be manually recompiled and migrated.
43. // in another terminal (i.e. not in the `truffle develop` prompt)
44. `npm`
45. `run`
46. `start`
47. Truffle can run tests written in Solidity or JavaScript against your smart contracts. Note the command varies slightly if you're in or outside of the development console.
48. // inside the development console.
49. `test`
50. // outside the development console..
51. `truffle`
52. `test`
53. For testing React components with Jest, you can install it as dev dependency. Compile your contracts before running Jest, or you may receive some file not found errors.

```

54. // ensure you are inside the client.old directory when running this
55. npm
56. install
57. --
58. save
59. -
60. dev
61. @
62. testing
63. -
64. library
65. /
66. react
67. npm
68. run
69. test
70. To build the application for production, use the build script. A production build will be in the dist/
71. folder.
72. // ensure you are inside the client.old directory when running this
73. npm
74. run
75. build

```

## FAQ

- How do I use this with the Ganache-CLI?
- It's as easy as modifying the config file
- . Depending on the port you're using, you'll need to update line 12 of `src/lib/Web3Context.js`
- . You'll also need to update the line 10 of `src/lib/MyContractContext.js`
- depending your network id.
- Where is my production build?
- The production build will be in the `dist/` folder after running `npm run build`
- in the root folder.
- Where can I find more documentation?
- This box is a marriage of [Truffle](#)
- and a React setup created with [React](#)
- . Either one would be a great place to start!
- This box is composed of [Truffle](#)
- , [React](#)
- , [Material-UI](#)
- , [Webpack 5](#)
- and [Babel 7](#)
- You can also find amazing royalty free illustrations on [Manypixels](#)

## Extra

☺ Do you like this library ? Buy me a coffee or support me with a star on Github

- Btc address: `bc1qettgagenn9nc8ks7ghntjfm96yvvkfhntk774`
- Eth address: `0xB0413d8D0336E263e289A915c383e152155881E0`