

# React angular and Node express application for production build

Chandan Gowda H S

To achieve production build

Step 1 :

we need to clone project it will contain React and node

Example: git clone <https://github.com/bbachi/react-nodejs-example.git>

```
chandanshankar@chandanshankar-Inspiron-5535: ~/Desktop/delete
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete$ git clone https://github.com/bbachi/react-nodejs-example.git
Cloning into 'react-nodejs-example'...
remote: Enumerating objects: 5879, done.
remote: Counting objects: 100% (604/604), done.
remote: Compressing objects: 100% (510/510), done.
remote: Total 5879 (delta 116), reused 96 (delta 94), pack-reused 5275
Receiving objects: 100% (5879/5879), 6.34 MiB | 8.89 MiB/s, done.
Resolving deltas: 100% (1195/1195), done.
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete$
```

- In that directory we have both front and backend

```
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete$ ls
react-nodejs-example
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete$ cd react-nodejs-example/
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ ls
api Dockerfile my-app
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$
```

- In here we have Both api and my-app folder

Step 2 :

- We need to start the npm in node
- cd api
- sudo npm install

```
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete$ ls
react-nodejs-example
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete$ cd react-nodejs-example/
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ ls
api Dockerfile my-app
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ cd api/
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ ls
gulpfile.js package.json package-lock.json README.md server.bundle.js server.js webpack.config.js
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ sudo npm install
sudo: npm: command not found
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ sudo npm install
npm WARN old lockfile
npm WARN old lockfile The package-lock.json file was created with an old version of npm,
npm WARN old lockfile so supplemental metadata must be fetched from the registry.
npm WARN old lockfile
npm WARN old lockfile This is a one-time fix-up, please be patient...
npm WARN old lockfile
[ ] | idealTree:inflate:node_modules/chownr: sill inflate node_modules/chownr
```

### Step 3 :

- we need to start react app
- cd my-app
- sudo npm install
- sudo npm start

```
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ cd ..
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ ls
api  Dockerfile  my-app
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ cd my-app/
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$ ls
package.json  package-lock.json  public  README.md  src
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$ npm install
npm WARN old lockfile
npm WARN old lockfile The package-lock.json file was created with an old version of npm,
npm WARN old lockfile so supplemental metadata must be fetched from the registry.
npm WARN old lockfile
npm WARN old lockfile This is a one-time fix-up, please be patient...
npm WARN old lockfile
[ ] | idealTree:inflate:node_modules/@babel/runtime-corejs3: sill inflate node_modules/@babel/runtime-
```

- After the npm start

## Then we need to build the application(production build)

### Step 4 :

- cd my-app
- sudo npm run build

```
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$ sudo npm run build

> my-app@0.1.0 build
> react-scripts build

Creating an optimized production build...
Browserslist: caniuse-lite is outdated. Please run:
  npx browserslist@latest --update-db
Compiled with warnings.

./src/components/DisplayBoard.js
  Line 5:11:  'headerStyle' is assigned a value but never used  no-unused-vars

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.

File sizes after gzip:

 42.39 KB  build/static/js/2.67710b93.chunk.js
 22.47 KB  build/static/css/2.af3c1da9.chunk.css
  1.95 KB  build/static/js/main.b06d4aaf.chunk.js
   770 B   build/static/js/runtime-main.7c0ae09a.js
   399 B   build/static/css/main.14ff7981.chunk.css

The project was built assuming it is hosted at /.
You can control this with the homepage field in your package.json.

The build folder is ready to be deployed.
You may serve it with a static server:

  npm install -g serve
  serve -s build

Find out more about deployment here:

  bit.ly/CRA-deploy

chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$
```

- it will create optimized production build
- out put will like -- The build folder is ready to be deployed.

Then Finally we can RUN production build application

- we need to create folder and then install npm and run node file
- and also directly we can run node page
- sudo node server.js

```
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$ ^C
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$ cd ..
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ ls
api  Dockerfile  my-app
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ cd my-app/
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$ ls
build  node_modules  package.json  package-lock.json  public  README.md  src
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/my-app$ cd ..
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ ls
api  Dockerfile  my-app
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ cd api/
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ ls
gulpfile.js  node_modules  package.json  package-lock.json  README.md  server.bundle.js  server.js  webpack.config.js
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ sudo node server.
node:internal/modules/cjs/loader:936
    throw err;
    ^
Error: Cannot find module '/home/chandanshankar/Desktop/delete/react-nodejs-example/api/server.'
    at Function.Module._resolveFilename (node:internal/modules/cjs/loader:933:15)
    at Function.Module._load (node:internal/modules/cjs/loader:778:27)
    at Function.executeUserEntryPoint [as runMain] (node:internal/modules/run_main:77:12)
    at node:internal/main/run_main_module:17:47 {
  code: 'MODULE_NOT_FOUND',
  requireStack: []
}
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ sudo node server.js
Server listening on the port::80
```

- Then it will show the server listening port is 80
- beacuse we run in localhost it sholud be localhost:80

## Step 5

- we can access the page

React With NodeJS

**Create User**

First Name:  Last Name:

Email:

**Users Created**

3

**Users**

| User Id | Firstname | Lastname | Email         |
|---------|-----------|----------|---------------|
| 1       | first1    | last1    | abc@gmail.com |
| 2       | first2    | last2    | abc@gmail.com |
| 3       | first3    | last3    | abc@gmail.com |

## Step 6 :

- To check if it connect to the backend just create user
- and we can see in user create dashboard
- when ever we create new user taht all reflect in our terminal

```
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example/api$ sudo node server.js
Server listening on the port::80
api/users called!
Adding user::::: {
  firstName: 'chadan',
  lastName: 'shankar',
  email: 'chandanshankar1999@gmail.com'
}
api/users called!
```

- we can see the new user get added shown in terminal page
- after that again api/user get called! To create an new user

## Final output we can see here

**Create User**

First Name:  Last Name:

Email:

**Users Created**

4

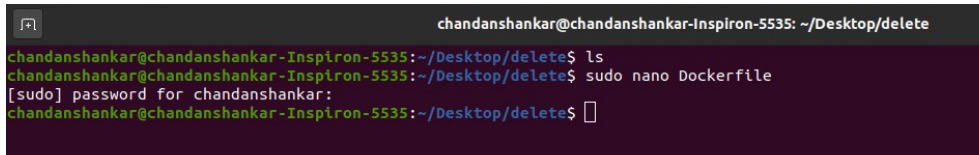
**Users**

| User Id | Firstname | Lastname | Email         |
|---------|-----------|----------|---------------|
| 1       | first1    | last1    | abc@gmail.com |
| 2       | first2    | last2    | abc@gmail.com |
| 3       | first3    | last3    | abc@gmail.com |

**From this task we can make all the process in Dockerfile**

## **Step1:**

### **Create Dockerfile and add the content**

A terminal window with a dark background. The prompt is 'chandanshankar@chandanshankar-Inspiron-5535: ~/Desktop/delete'. The user enters 'ls', then 'sudo nano Dockerfile'. A password prompt '[sudo] password for chandanshankar:' is shown, followed by the user's input. The prompt returns to 'chandanshankar@chandanshankar-Inspiron-5535: ~/Desktop/delete\$'.

- Dockerfile contains

```
FROM node:10 AS ui-build
WORKDIR /usr/src/app
COPY my-app/ ./my-app/
RUN cd my-app && npm install && npm run build
```

```
FROM node:10 AS server-build
WORKDIR /root/
COPY --from=ui-build /usr/src/app/my-app/build ./my-app/build
COPY api/package*.json ./api/
RUN cd api && npm install
COPY api/server.js ./api/
```

```
EXPOSE 3080
```

```
CMD ["node", "./api/server.js"]
```

---

```
FROM node:10 AS ui-build
// fetch node image from dockerhub
```

```
WORKDIR /usr/src/app
```



//set workind directory as /usr/src/app

COPY my-app/ ./my-app/

//copy file to insede image

RUN cd my-app && npm install && npm run build

//run those npm install command and build command

FROM node:10 AS server-build

WORKDIR /root/

COPY --from=ui-build /usr/src/app/my-app/build ./my-app/build

COPY api/package\*.json ./api/

RUN cd api && npm install

COPY api/server.js ./api/

EXPOSE 3080

//and expose in 3080 port

CMD ["node", "./api/server.js"]

//using node we can run server.js

Step2:

we need to build the image from dockerfile

- sudo docker build -t chandna .
- After we can get success output we can run this iamge

```
api Dockerfile my-app
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ sudo docker build -t chandan .
Sending build context to Docker daemon  8.371MB
Step 1/12 : FROM node:10 AS ui-build
--> 28dca6642db8
Step 2/12 : WORKDIR /usr/src/app
--> Using cache
--> 869e77ca7d39
Step 3/12 : COPY my-app/ ./my-app/
--> db3f291d9a7b
Step 4/12 : RUN cd my-app && npm install && npm run build
--> Running in 87ff8e8ff7a9

> core-js@2.6.11 postinstall /usr/src/app/my-app/node_modules/babel-runtime/node_modules/core-js
> node -e "try{require('./postinstall')}catch(e){}"

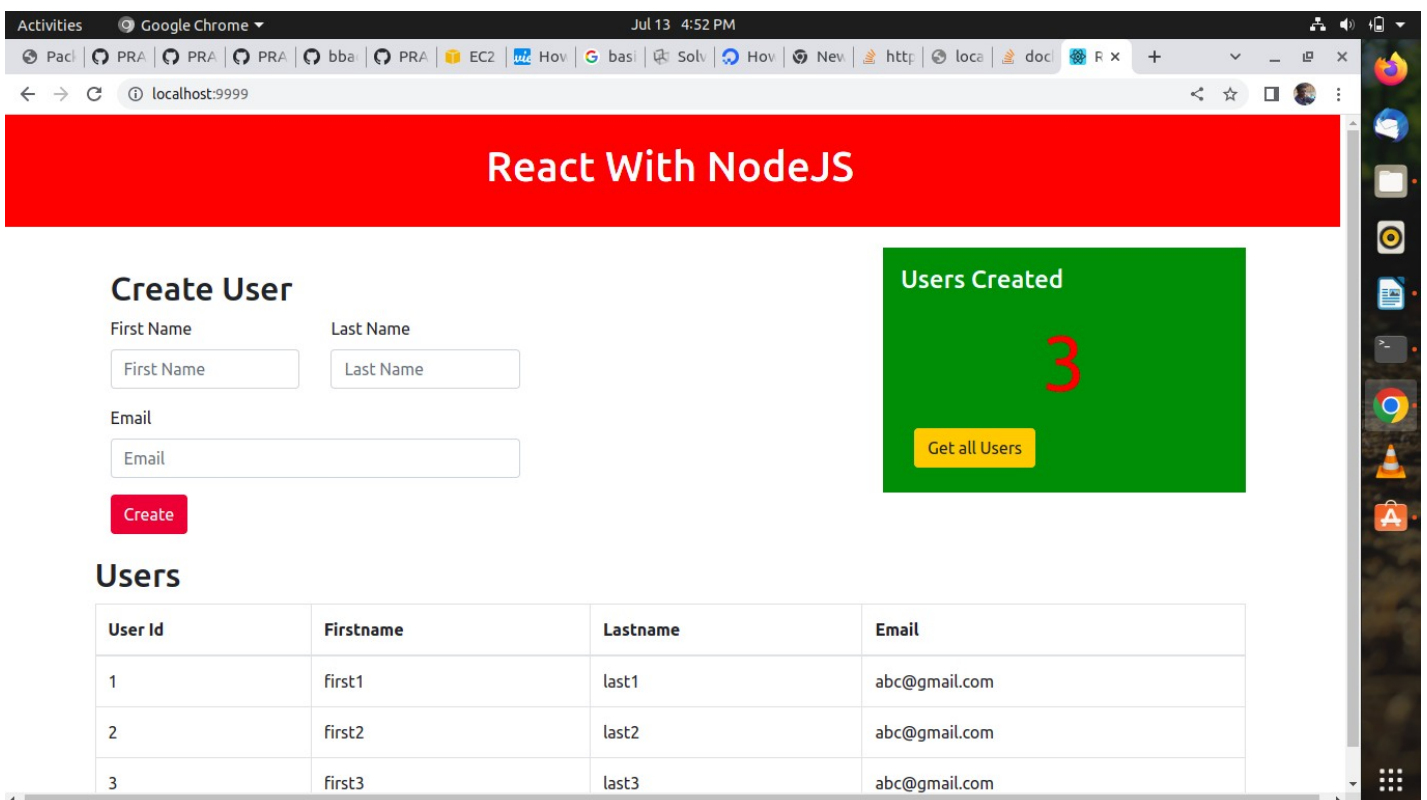
Thank you for using core-js ( https://github.com/zloirock/core-js ) for polyfilling JavaScript standard library!
```

```

Successfully built 8491a25697a3
Successfully tagged chandan:latest
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
chandan       latest    8491a25697a3   51 seconds ago 950MB
<none>        <none>    215c7847cddb   About a minute ago 1.11GB
task2         latest    4fd7a1debe30   42 hours ago   1.12GB
ubuntu        latest    27941809078c   5 weeks ago    77.8MB
node          10        28dca6642db8   15 months ago  910MB
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ sudo docker run -it -d -p 9999:80 chandan
f9d9fd58d528354f00bf021309d915dfc5bad156cf00c85d8196a276994a2521
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
f9d9fd58d528   chandan   "docker-entrypoint.s..." 9 seconds ago  Up 5 seconds  0.0.0.0:9999->80/tcp, :::9999->80/tcp  mystifying_liskov
chandanshankar@chandanshankar-Inspiron-5535:~/Desktop/delete/react-nodejs-example$

```

- we can access this application using localhost:9999



React With NodeJS

**Create User**

First Name:  Last Name:

Email:

**Users Created**

3

**Users**

| User Id | Firstname | Lastname | Email         |
|---------|-----------|----------|---------------|
| 1       | first1    | last1    | abc@gmail.com |
| 2       | first2    | last2    | abc@gmail.com |
| 3       | first3    | last3    | abc@gmail.com |