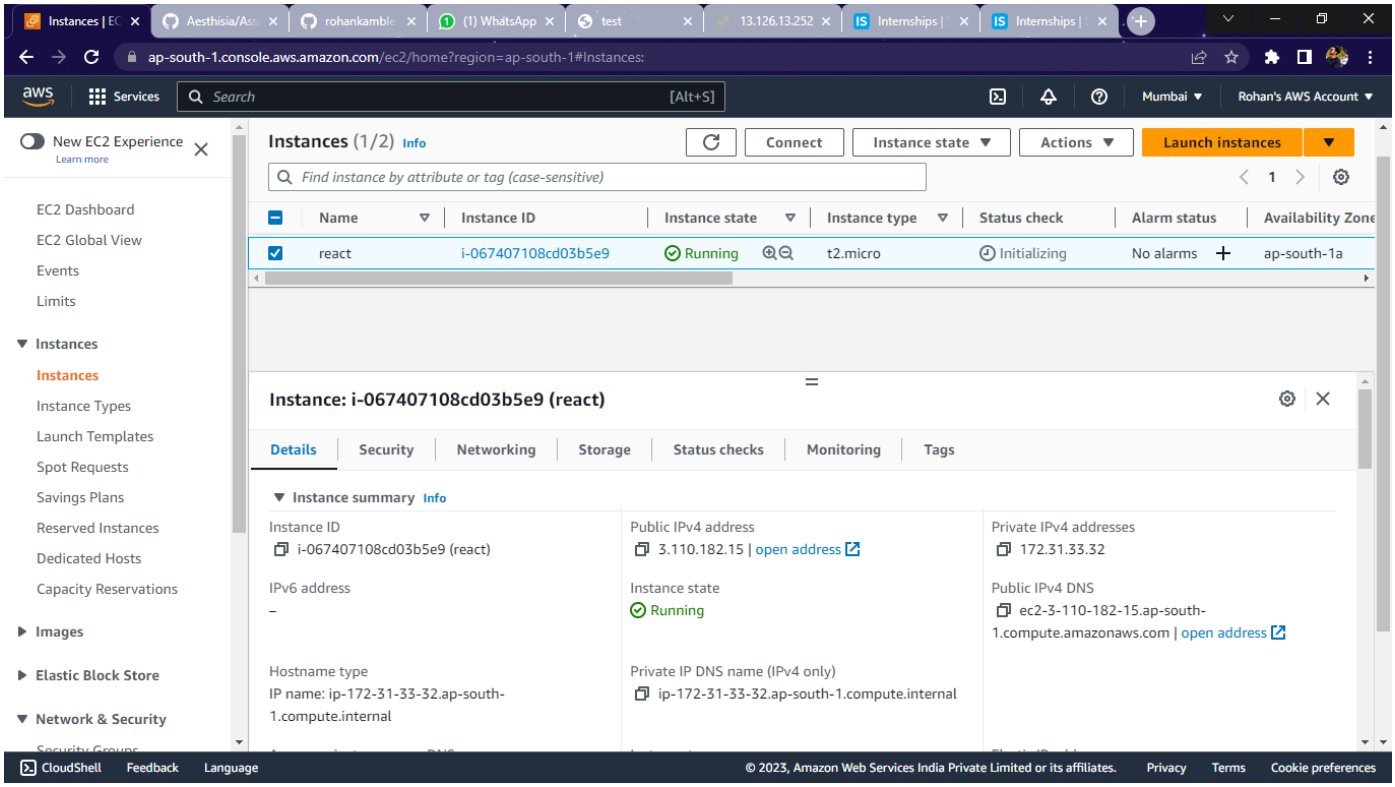
Project No 2

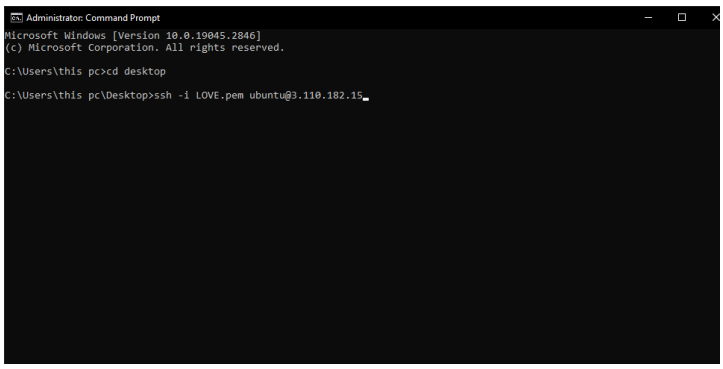
React based application build and create the image of the app using docker and deploy the container using docker-compose

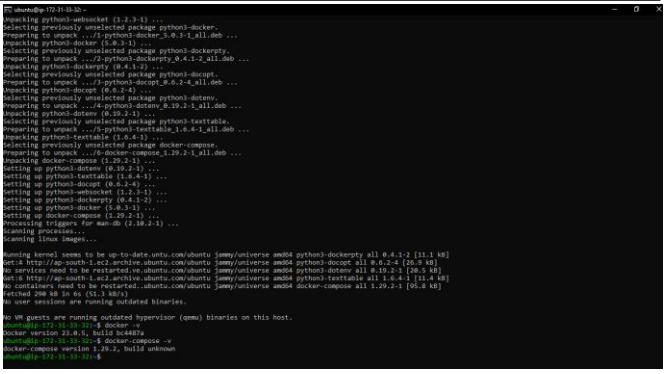
Name: Rohan Kamble

1. Firstly I am configuring a AWS server

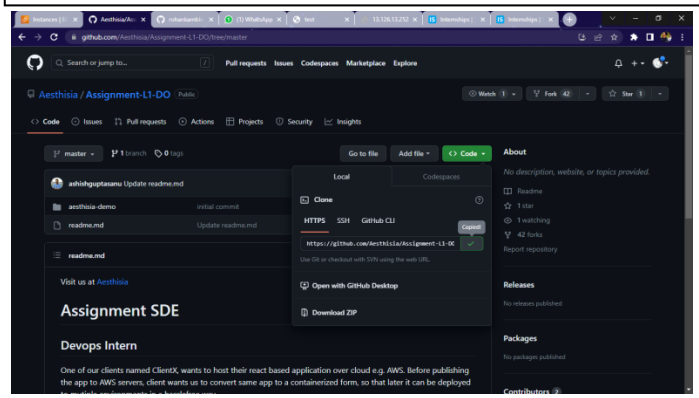


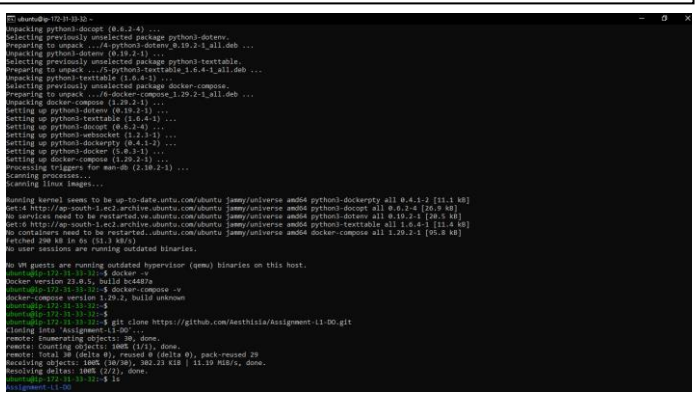
2. Then I am doing ssh in the server and then Installing Docker and Docker-compose on the server.





3. Then I am cloning the repository





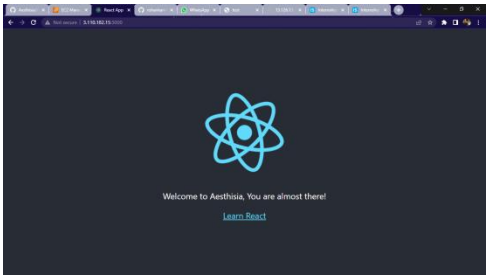
4. I am installing the necessary dependencies using npm but npm is not installed in the server so I am installing npm by command “sudo apt install npm”.

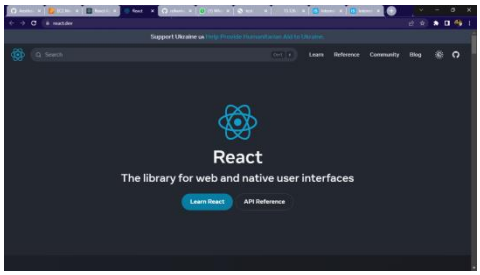
```
ubuntu@ip-172-31-33-32: ~/Assignment-L1-DO
ubuntu@ip-172-31-33-32:~/Assignment-L1-DO$ sudo apt install npm
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adwaita-icon-theme at-spi2-core build-essential bzip2 cpp cpp-11 dconf-gsettings-backend dconf-service dpkg-dev fakeroot fontconfig fontconfig-config
  fonts-dejavu-core g++ g++-11 gcc gcc-11 gcc-11-base gsettings-desktop-schemas gtk-update-icon-cache gyp hicolor-icon-theme humanity-icon-theme javascript-common
  libalgorithm-diff-perl libalgorithm-diff-xs-perl libalgorithm-merge-perl libasan6 libatk-bridge2.0-0 libatk1.0-0 libatk1.0-data libatomic1 libatspi2.0-0
  libauthen-sasl-perl libavahi-client3 libavahi-common-data libavahi-common3 libc-ares2 libc-dev-bin libc-devtools libc6-dev libcairo-gobject2 libcairo2 libcc1-0
  libclone-perl libcolord2 libcrypt-dev libcups2 libdata-dump-perl libdatrie1 libdconf1 libdeflate0 libdpkg-perl libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2
  libdrm-radeon1 libencode-locale-perl libepoxy0 libfakeroot libfile-basedir-perl libfile-desktopentry-perl libfile-fcntllock-perl libfile-listing-perl
  libfile-mimeinfo-perl libfont-afm-perl libfontconfig1 libfontenc1 libgcc-11-dev libgd3 libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgl1
  libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgomp1 libgraphite2-3 libgtk-3-0 libgtk-3-bin libgtk-3-common libgtkd-3-0
  libharfbuzz0b libhtml-form-perl libhtml-format-perl libhtml-parser-perl libhtml-tagset-perl libhtml-tree-perl libhttp-cookies-perl libhttp-daemon-perl
  libhttp-date-perl libhttp-message-perl libhttp-negotiate-perl libice6 libio-html-perl libio-socket-ssl-perl libio-stringy-perl libipc-system-simple-perl libisl23
  libitm1 libjbig0 libjpeg-turbo8 libjpeg8 libjs-events libjs-highlight.js libjs-inherits libjs-is-typedarray libjs-psl libjs-source-map libjs-sprintf-js
  libjs-typedarray-to-buffer liblcms2-2 libllvm11 libllvm15 liblsan0 liblwp-mediatypes-perl liblwp-protocol-https-perl libmailtools-perl libmpc3 libnet-dbus-perl
  libnet-http-perl libnet-smtp-ssl-perl libnet-ssleay-perl libnode-dev libnode72 libnotify-bin libnotify4 libnsl-dev libpango-1.0-0 libpangocairo-1.0-0
  libpangoft2-1.0-0 libpciaccess0 libphobos2-ldc-shared98 libpixman-1-0 libquadmath0 librsvg2-2 librsvg2-common libsensors-config libsensors5 libsm6 libssl-dev
  libssl3 libstdc++-11-dev libthai-data libthai0 libtie-ixhash-perl libtiff5 libtimedate-perl libtirpc-dev libtry-tiny-perl libtsan0 libubsan1 liburi-perl libuv1-dev
  libvte-2.91-0 libvte-2.91-common libvted-3-0 libwayland-client0 libwayland-cursor0 libwayland-egl1 libwebp7 libwww-perl libwww-robotrules-perl libx11-protocol-perl
  libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-render0 libxcb-shape0 libxcb-shm0 libxcb-sync1 libxcb-xfixes0 libxcomposite1
  libxcursor1 libxdamage1 libxfixes3 libxft2 libxi6 libxinerama1 libxkbcommon0 libxkbfile1 libxml-parser-perl libxml-twig-perl libxml-xpathengine-perl libxmu6 libxpm4
  libxrandr2 libxrender1 libxshmfence1 libxt6 libxtst6 libxv1 libxxf86dga1 libxxf86vm1 linux-libc-dev lto-disabled-list make manpages-dev node-abab node-abbrev
  node-agent-base node-ansi-regex node-ansi-styles node-ansistyles node-aproba node-archy node-are-we-there-yet node-argparse node-arrify node-asap node-asynckit
  node-balanced-match node-brace-expansion node-builtins node-cacache node-chalk node-chownr node-clean-yaml-object node-cli-table node-clone node-color-convert
  node-color-name node-colors node-columnify node-combined-stream node-commander node-console-control-strings node-copy-concurrently node-core-util-is node-coveralls
  node-cssom node-cssstyle node-debug node-decompress-response node-defaults node-delayed-stream node-delegates node-depd node-diff node-encoding node-end-of-stream
  node-err-code node-escape-string-regexp node-esprima node-events node-fancy-log node-fetch node-foreground-child node-form-data node-fs-write-stream-atomic
  node-fs.realpath node-function-bind node-gauge node-get-stream node-glob node-got node-graceful-fs node-growl node-gyp node-has-flag node-has-unicode
  node-hosted-git-info node-https-proxy-agent node-iconv-lite node-iferr node-imurmurhash node-indent-string node-inflight node-inherits node-ini node-ip
  node-ip-regex node-is-buffer node-is-plain-obj node-is-typedarray node-isarray node-isexe node-js-yaml node-jsdom node-json-buffer node-json-parse-better-errors
  node-jsonparse node-kind-of node-lcov-parse node-lodash-packages node-log-driver node-lowercase-keys node-lru-cache node-mime node-mime-types node-mimic-response
  node-minimatch node-minimist node-minipass node-mkdirp node-move-concurrently node-ms node-mute-stream node-negotiator node-nopt node-normalize-package-data
  node-npm-bundled node-npm-package-arg node-npmlog node-object-assign node-once node-opener node-osenv node-p-cancelable node-p-map node-path-is-absolute
  node-process-nextick-args node-promise-inflight node-promise-retry node-promzard node-psl node-pump node-punycode node-quick-lru node-read node-read-package-json
  node-readable-stream node-resolve node-retry node-rimraf node-run-queue node-safe-buffer node-semver node-set-blocking node-signal-exit node-slash node-slice-ansi
  node-source-map node-source-map-support node-spdx-correct node-spdx-exceptions node-spdx-expression-parse node-spdx-license-ids node-sprintf-js node-ssri
  node-stack-utils node-stealthy-require node-string-decoder node-string-width node-strip-ansi node-supports-color node-tap node-tap-mocha-reporter node-tap-parser
  node-tar node-text-table node-time-stamp node-tmatch node-tough-cookie node-typedarray-to-buffer node-unique-filename node-universalify node-util-deprecate
  node-validate-npm-package-license node-validate-npm-package-name node-wcwidth.js node-webidl-conversions node-whatwg-fetch node-which node-wide-align node-wrappy
  node-write-file-atomic node-ws node-yallist nodejs nodejs-doc perl-openssl-defaults rpcsvc-proto session-migration tilix tilix-common ubuntu-mono x11-common
  x11-utils x11-xserver-utils xdg-utils
Suggested packages:
  bzip2-doc cpp-doc gcc-11-locales debian-keyring g++-multilib g++-11-multilib gcc-11-doc gcc-multilib autoconf automake libtool flex bison gdb gcc-doc
```

5. Installed dependencies by command “npm install” and after the dependencies are installed I started the development server using “npm start”

```
ubuntu@ip-172-31-33-32: ~/Assignment-L1-DO/aesthisia-demo
Compiled successfully!

You can now view aesthisia-demo in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://172.31.33.32:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

6. Now its time to check the browser with email Id of our server on port 3000 and its working





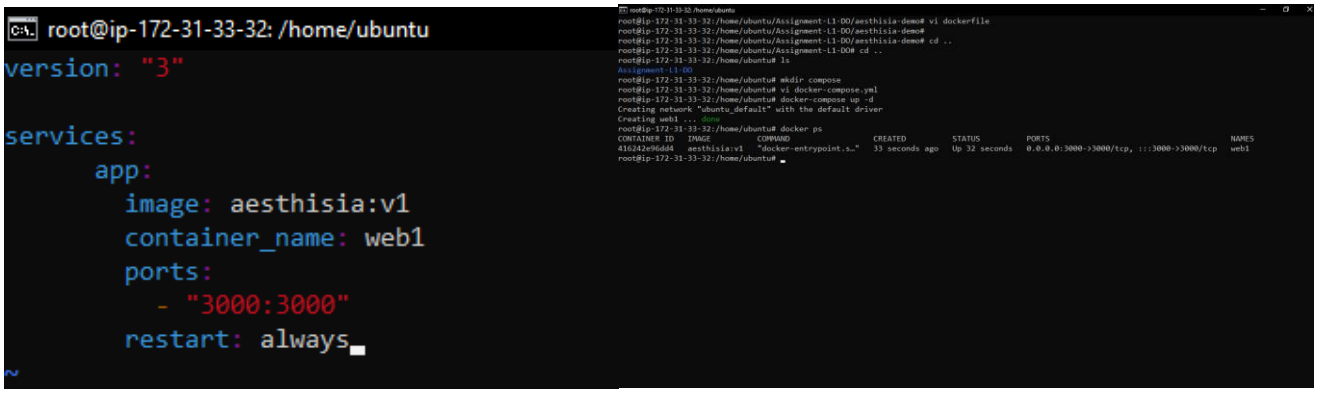
7. Now its time to create the image of the app so that we can containerize it, for that we need to create dockerfile.

```
root@ip-172-31-33-32: /home/ubuntu/Assignment-L1-DO/aesthisia-demo
FROM node:latest

WORKDIR /home/ubuntu/Assignment-L1-DO/aesthisia-demo

COPY package.json package-lock.json ./

RUN npm install

COPY . .

RUN npm run build

ENV PORT=3000
EXPOSE 3000

CMD ["npm", "start"]
~
~
~
~
```

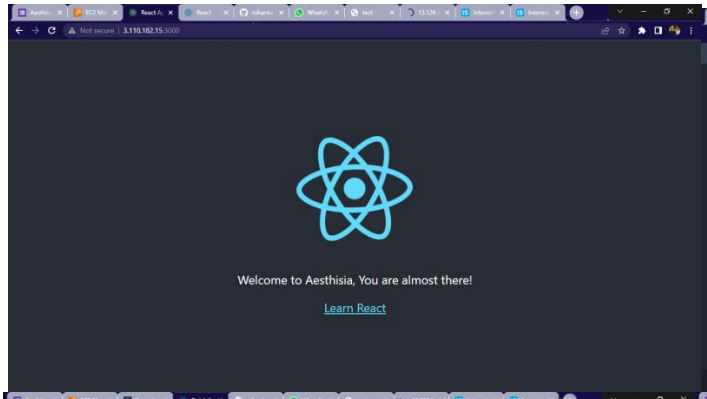
8. Now we need to build the image with the help of command “docker build -t aesthisia:v1 -f ./dockerfile .”

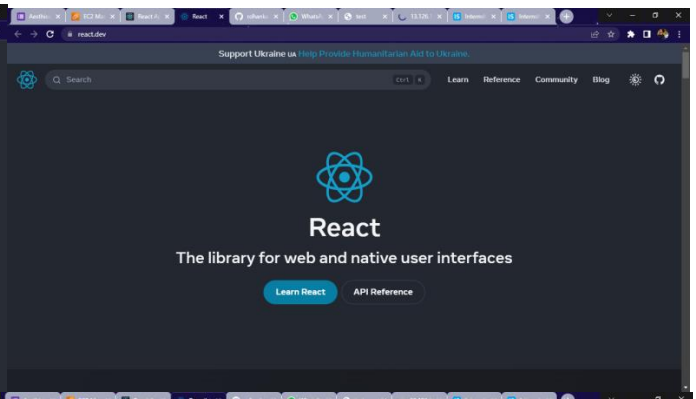
```
root@ip-172-31-33-32: /home/ubuntu/Assignment-L1-DO/aesthisia-demo
[+] Building 77.0s (9/10)
 => [internal] load build definition from dockerfile                                    0.0s
 => => transferring dockerfile: 245B                                                    0.0s
[+] Building 77.1s (9/10)
 => [internal] load build definition from dockerfile                                    0.0s
[+] Building 126.0s (11/11) FINISHED
 => [internal] load build definition from dockerfile                                    0.0s
 => => transferring dockerfile: 245B                                                    0.0s
 => [internal] load .dockerignore                                                       0.0s
 => => transferring context: 2B                                                         0.0s
 => [internal] load metadata for docker.io/library/node:latest                          0.7s
 => [1/6] FROM docker.io/library/node:latest@sha256:0efc3ef3fea2822c9d16da084c40181ed7f74b6f45141100580f9887ccc8e9a1   0.0s
 => [internal] load build context                                                      31.2s
 => => transferring context: 247.40MB                                                  31.0s
 => CACHED [2/6] WORKDIR /home/ubuntu/Assignment-L1-DO/aesthisia-demo                   0.0s
 => CACHED [3/6] COPY package.json package-lock.json ./                                 0.0s
 => CACHED [4/6] RUN npm install                                                        0.0s
 => [5/6] COPY . .                                                                     29.8s
 => [6/6] RUN npm run build                                                            22.8s
 => exporting to image                                                                 41.3s
 => => exporting layers                                                                41.3s
 => => writing image sha256:95a0491d252925986cb06ad45d9d5d5b39e6681d82e9b9f4a58b1aaa28cdd2cb   0.0s
 => => naming to docker.io/library/aesthisia:v1                                         0.0s
root@ip-172-31-33-32:/home/ubuntu/Assignment-L1-DO/aesthisia-demo# docker images
REPOSITORY   TAG       IMAGE ID       CREATED         SIZE
aesthisia    v1        95a0491d2529   2 minutes ago   1.52GB
root@ip-172-31-33-32:/home/ubuntu/Assignment-L1-DO/aesthisia-demo#
```

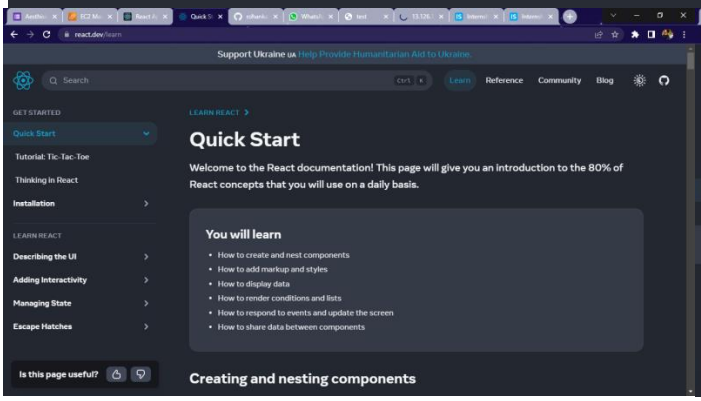
9. Now with the help of docker-compose are are launching container and for that we need to create docker-compose.yml file, Launching container by “docker-compose up –d”

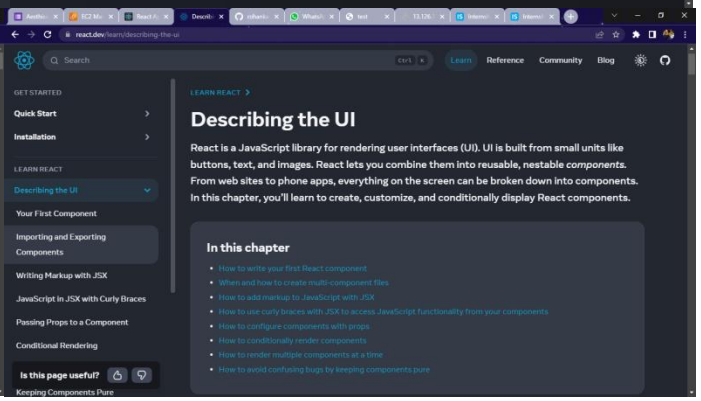


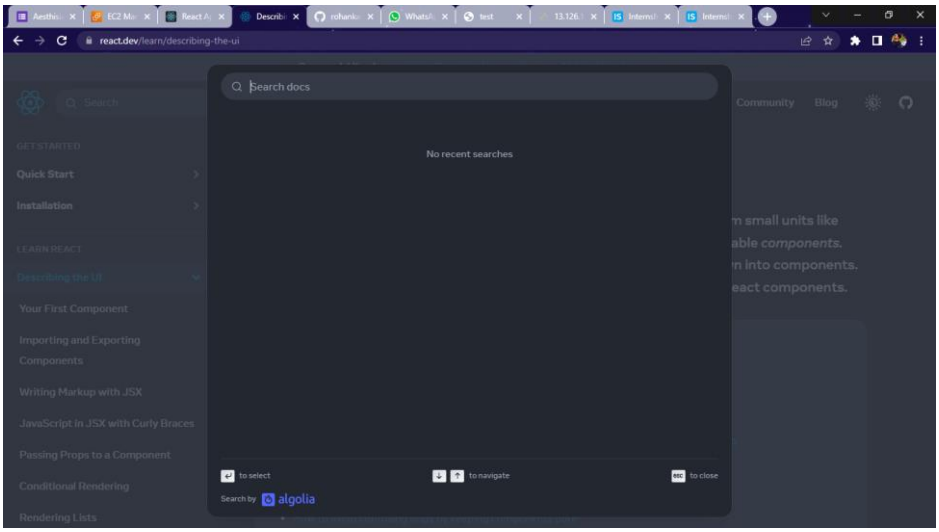
10. Now we have lauched the container using docker-compose and now we have automated it.











Regards

Rohan Kamble