Compiled successfully!

You can now view **allax** in the browser.

  http://localhost:**3000**/

Note that the development build is not optimized.

To create a production build, use npm run build.

npm run build

^[[A^[[A

^Cloop-depaulsecure-230-38:allax victorshollenberger$ npm runbuild

Usage: npm <command>

where <command> is one of:

    access, adduser, audit, bin, bugs, c, cache, ci, cit,

    completion, config, create, ddp, dedupe, deprecate,

    dist-tag, docs, doctor, edit, explore, get, help,

    help-search, hook, i, init, install, install-test, it, link,

    list, ln, login, logout, ls, outdated, owner, pack, ping,

    prefix, profile, prune, publish, rb, rebuild, repo, restart,

    root, run, run-script, s, se, search, set, shrinkwrap, star,

    stars, start, stop, t, team, test, token, tst, un,

    uninstall, unpublish, unstar, up, update, v, version, view,

    whoami

npm <command> -h     quick help on <command>

npm -l           display full usage info

npm help <term>  search for help on <term>

npm help npm     involved overview

Specify configs in the ini-formatted file:

    /Users/victorshollenberger/.npmrc

or on the command line via: npm <command> --key value

Config info can be viewed via: npm help config

npm@6.0.1 /usr/local/lib/node\_modules/npm

Did you mean this?

    rebuild

loop-depaulsecure-230-38:allax victorshollenberger$ npm run build

> allax@0.1.0 build /Users/victorshollenberger/Desktop/allax

> react-scripts build

Creating an optimized production build...

Compiled successfully.

File sizes after gzip:

  38.06 KB  build/static/js/main.a3b22bcc.js

  299 B     build/static/css/main.c17080f1.css

The project was built assuming it is hosted at the server root.

You can control this with the homepage field in your package.json.

For example, add this to build it for GitHub Pages:

  "homepage" : "http://myname.github.io/myapp",

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:

  http://bit.ly/2vY88Kr

loop-depaulsecure-230-38:allax victorshollenberger$ ls

README.md package-lock.json src

build package.json

node\_modules public

loop-depaulsecure-230-38:allax victorshollenberger$ firebase init

**🔥🔥🔥🔥🔥🔥🔥🔥 🔥🔥🔥🔥 🔥🔥🔥🔥🔥🔥🔥🔥  🔥🔥🔥🔥🔥🔥🔥🔥 🔥🔥🔥🔥🔥🔥🔥🔥     🔥🔥🔥     🔥🔥🔥🔥🔥🔥  🔥🔥🔥🔥🔥🔥🔥🔥**

**🔥🔥        🔥🔥  🔥🔥     🔥🔥 🔥🔥       🔥🔥     🔥🔥  🔥🔥   🔥🔥  🔥🔥       🔥🔥**

**🔥🔥🔥🔥🔥🔥    🔥🔥  🔥🔥🔥🔥🔥🔥🔥🔥  🔥🔥🔥🔥🔥🔥   🔥🔥🔥🔥🔥🔥🔥🔥  🔥🔥🔥🔥🔥🔥🔥🔥🔥  🔥🔥🔥🔥🔥🔥  🔥🔥🔥🔥🔥🔥**

**🔥🔥        🔥🔥  🔥🔥    🔥🔥  🔥🔥       🔥🔥     🔥🔥 🔥🔥     🔥🔥       🔥🔥 🔥🔥**

**🔥🔥       🔥🔥🔥🔥 🔥🔥     🔥🔥 🔥🔥🔥🔥🔥🔥🔥🔥 🔥🔥🔥🔥🔥🔥🔥🔥  🔥🔥     🔥🔥  🔥🔥🔥🔥🔥🔥  🔥🔥🔥🔥🔥🔥🔥🔥**

You're about to initialize a Firebase project in this directory:

**/Users/victorshollenberger/Desktop/allax**

? **Which Firebase CLI features do you want to setup for this folder? Press Space**

**to select features, then Enter to confirm your choices.** Database: Deploy Firebas

e Realtime Database Rules, Hosting: Configure and deploy Firebase Hosting sites

**=== Project Setup**

First, let's associate this project directory with a Firebase project.

You can create multiple project aliases by running **firebase use --add**,

but for now we'll just set up a default project.

? **Select a default Firebase project for this directory:** ALLAX (allax-d8a97)

**=== Database Setup**

Firebase Realtime Database Rules allow you to define how your data should be

structured and when your data can be read from and written to.

? **What file should be used for Database Rules?** database.rules.json

✔  Database Rules for **allax-d8a97** have been downloaded to **database.rules.json**.

Future modifications to **database.rules.json** will update Database Rules when you run

**firebase deploy**.

**=== Hosting Setup**

Your **public** directory is the folder (relative to your project directory) that

will contain Hosting assets to be uploaded with **firebase deploy**. If you

have a build process for your assets, use your build's output directory.

? **What do you want to use as your public directory?** public

? **Configure as a single-page app (rewrite all urls to /index.html)?** Yes

? **File public/index.html already exists. Overwrite?** No

**i** Skipping write of **public/index.html**

**i** Writing configuration info to **firebase.json**...

**i** Writing project information to **.firebaserc**...

✔  Firebase initialization complete!

loop-depaulsecure-230-38:allax victorshollenberger$ ls

README.md firebase.json package.json

build node\_modules public

database.rules.json package-lock.json src

loop-depaulsecure-230-38:allax victorshollenberger$

loop-depaulsecure-230-38:allax victorshollenberger$ npm run build

> allax@0.1.0 build /Users/victorshollenberger/Desktop/allax

> react-scripts build

Creating an optimized production build...

Compiled successfully.

File sizes after gzip:

  38.06 KB  build/static/js/main.a3b22bcc.js

  299 B     build/static/css/main.c17080f1.css

The project was built assuming it is hosted at the server root.

You can control this with the homepage field in your package.json.

For example, add this to build it for GitHub Pages:

  "homepage" : "http://myname.github.io/myapp",

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:

  http://bit.ly/2vY88Kr

loop-depaulsecure-230-38:allax victorshollenberger$ firebase deploy

**=== Deploying to 'allax-d8a97'...**

**i** deploying **database, hosting**

**i** **database:** checking rules syntax...

✔  **database:** rules syntax for database **allax-d8a97** is valid

**i** **hosting:** preparing **public** directory for upload...

✔  **hosting:** 3 files uploaded successfully

**i** **database:** releasing rules...

✔  **database:** rules for database **allax-d8a97** released successfully

✔  **Deploy complete!**

**Project Console:** https://console.firebase.google.com/project/allax-d8a97/overview

**Hosting URL:** https://allax-d8a97.firebaseapp.com

loop-depaulsecure-230-38:allax victorshollenberger$ sudo npm install -S firebase

Password:

npm ERR! file /Users/victorshollenberger/Desktop/allax/package.json

npm ERR! code EJSONPARSE

npm ERR! JSON.parse Failed to parse json

npm ERR! JSON.parse Unexpected string in JSON at position 340 while parsing '{

npm ERR! JSON.parse   "name": "allax",

npm ERR! JSON.parse   "version": "0.1.0'

npm ERR! JSON.parse Failed to parse package.json data.

npm ERR! JSON.parse package.json must be actual JSON, not just JavaScript.

npm ERR! A complete log of this run can be found in:

npm ERR!     /Users/victorshollenberger/.npm/\_logs/2018-05-19T00\_00\_26\_844Z-debug.log

loop-depaulsecure-230-38:allax victorshollenberger$ npm install -s firebase

loop-depaulsecure-230-38:allax victorshollenberger$ fire.js

-bash: fire.js: command not found

loop-depaulsecure-230-38:allax victorshollenberger$ cd allax

-bash: cd: allax: No such file or directory

loop-depaulsecure-230-38:allax victorshollenberger$ cd src

loop-depaulsecure-230-38:src victorshollenberger$ ls

App.css index.css registerServiceWorker.js

App.js index.js

App.test.js logo.svg

loop-depaulsecure-230-38:src victorshollenberger$ create-react-fire.js

-bash: create-react-fire.js: command not found

loop-depaulsecure-230-38:src victorshollenberger$ npm run deploy

npm ERR! file /Users/victorshollenberger/Desktop/allax/package.json

npm ERR! code EJSONPARSE

npm ERR! JSON.parse Failed to parse json

npm ERR! JSON.parse Unexpected string in JSON at position 340 while parsing '{

npm ERR! JSON.parse   "name": "allax",

npm ERR! JSON.parse   "version": "0.1.0'

npm ERR! JSON.parse Failed to parse package.json data.

npm ERR! JSON.parse package.json must be actual JSON, not just JavaScript.

npm ERR! A complete log of this run can be found in:

npm ERR!     /Users/victorshollenberger/.npm/\_logs/2018-05-19T00\_11\_03\_845Z-debug.log

loop-depaulsecure-230-38:src victorshollenberger$ ls

App.css fire.js logo.svg

App.js index.css registerServiceWorker.js

App.test.js index.js

loop-depaulsecure-230-38:src victorshollenberger$ cd allax

-bash: cd: allax: No such file or directory

loop-depaulsecure-230-38:src victorshollenberger$ cd desktop

-bash: cd: desktop: No such file or directory

loop-depaulsecure-230-38:src victorshollenberger$ desktop

-bash: desktop: command not found

loop-depaulsecure-230-38:src victorshollenberger$ help

GNU bash, version 3.2.57(1)-release (x86\_64-apple-darwin15)

These shell commands are defined internally.  Type `help' to see this list.

Type `help name' to find out more about the function `name'.

Use `info bash' to find out more about the shell in general.

Use `man -k' or `info' to find out more about commands not in this list.

A star (\*) next to a name means that the command is disabled.

 JOB\_SPEC [&]                       (( expression ))

 . filename [arguments]             :

 [ arg... ]                         [[ expression ]]

 alias [-p] [name[=value] ... ]     bg [job\_spec ...]

 bind [-lpvsPVS] [-m keymap] [-f fi break [n]

 builtin [shell-builtin [arg ...]]  caller [EXPR]

 case WORD in [PATTERN [| PATTERN]. cd [-L|-P] [dir]

 command [-pVv] command [arg ...]   compgen [-abcdefgjksuv] [-o option

 complete [-abcdefgjksuv] [-pr] [-o continue [n]

 declare [-afFirtx] [-p] [name[=val dirs [-clpv] [+N] [-N]

 disown [-h] [-ar] [jobspec ...]    echo [-neE] [arg ...]

 enable [-pnds] [-a] [-f filename]  eval [arg ...]

 exec [-cl] [-a name] file [redirec exit [n]

 export [-nf] [name[=value] ...] or false

 fc [-e ename] [-nlr] [first] [last fg [job\_spec]

 for NAME [in WORDS ... ;] do COMMA for (( exp1; exp2; exp3 )); do COM

 function NAME { COMMANDS ; } or NA getopts optstring name [arg]

 hash [-lr] [-p pathname] [-dt] [na help [-s] [pattern ...]

 history [-c] [-d offset] [n] or hi if COMMANDS; then COMMANDS; [ elif

 jobs [-lnprs] [jobspec ...] or job kill [-s sigspec | -n signum | -si

 let arg [arg ...]                  local name[=value] ...

 logout                             popd [+N | -N] [-n]

 printf [-v var] format [arguments] pushd [dir | +N | -N] [-n]

 pwd [-LP]                          read [-ers] [-u fd] [-t timeout] [

 readonly [-af] [name[=value] ...]  return [n]

 select NAME [in WORDS ... ;] do CO set [--abefhkmnptuvxBCHP] [-o opti

 shift [n]                          shopt [-pqsu] [-o long-option] opt

 source filename [arguments]        suspend [-f]

 test [expr]                        time [-p] PIPELINE

 times                              trap [-lp] [arg signal\_spec ...]

 true                               type [-afptP] name [name ...]

 typeset [-afFirtx] [-p] name[=valu ulimit [-SHacdfilmnpqstuvx] [limit

 umask [-p] [-S] [mode]             unalias [-a] name [name ...]

 unset [-f] [-v] [name ...]         until COMMANDS; do COMMANDS; done

 variables - Some variable names an wait [n]

 while COMMANDS; do COMMANDS; done  { COMMANDS ; }

loop-depaulsecure-230-38:src victorshollenberger$ exit src

logout

-bash: exit: src: numeric argument required

Saving session...

...copying shared history...

...saving history...truncating history files...

...completed.

Deleting expired sessions...13 completed.

[Process completed]