1. yarn add sucrase nodemon -D

* Sucrase para poder usar import ao invés de require

1. docker run --name database -e POSTGRES\_PASSWORD=docker -p 5432:5432 -d postgres

* Criando um container: Passa o nome, senha, porta servidor e porta do docker e o nome da imagem.
* Ele instala as paradas e retorna um id único do nosso container
* Com o comando run, várias aplicações podem usar esse container. Mas para frente vamos ver como criamos um container exclusivo para uma aplicação.

1. docker ps

* Vai listar todos os containers que estão em execução.

1. docker ps -a

* Vai listar todos os containers que tem na máquina

1. docker start nome

* Toda vez que reiniciar a máquina o container para de executar.

1. docker log nome

* Para verificar o log

1. yarn add eslint

* Ferramenta para ajudar a padronizar o código

1. yarn eslint --init

* Para configurar o eslint
* eslint + prettier;
* Tive que fazer um mix do tutorial abaixo com o do bootcamp para poder funcionar

<https://medium.com/@jeancabral/gatsbyjs-configurando-eslint-e-prettier-no-visual-studio-code-bba4a25a216>

1. yarn eslint --fix src --ext .js

* Para ajeitar todos os arquivos js da pasta

1. yarn add sequelize

* instalando a dependência do sequelize

1. yarn add sequelize-cli -D

* Interface de linha de comandos que vai nos ajudar

1. yarn add pg pg-hstore

* Precisa para o dialect do sequelize usando postgres. Olhar documentação.
* Não lembro muito bem, revisar!!!

1. yarn sequelize migration:create --name=create-users

* comando do sequelize-cli para criar migration

1. yarn sequelize db:migrate

* comando para o sequelize criar a tabela na base de dados

1. yarn sequelize db:migrate:undo:all

* undo para desfazer a última migrations e all para desfazer todas