**FAI – CENTRO DE ENSINO SUPERIOR EM GESTÃO, TECNOLOGIA E EDUCAÇÃO**

**CURSO DE SISTEMAS DE INFORMAÇÃO**

**ALEXANDRE DA SILVA RIBEIRO**

**MANUAL DE IMPLANTAÇÃO DE SOFTWARE**

**SANTA RITA DO SAPUCAÍ - MG**

**2020**

Sumário

[1 IMPLANTAÇÃO *FRONTEND* NA VERCEL 3](#_Toc57121276)

[1.1 Criar a conta/fazer autenticação no Vercel 3](#_Toc57121277)

[1.2 Importar o projeto 4](#_Toc57121278)

[1.3 Configurar projeto 4](#_Toc57121279)

[2 IMPLANTAÇÃO *BACKEND* NA HEROKU 8](#_Toc57121280)

[2.1 Deploy da aplicação NodeJs 8](#_Toc57121281)

[2.1 Criando um add-on PostgreSQL no Heroku 11](#_Toc57121282)

[2.2 Configurando as variáveis do projeto 12](#_Toc57121283)

# 1 IMPLANTAÇÃO *FRONTEND* NA VERCEL

Implantação é a fase do ciclo de vida de um software, no contexto de um Sistema de Informação, que corresponde à passagem do software de um ambiente de desenvolvimento para o ambiente final do usuário.

Este manual demonstra o processo de implantação do *EasyJob* em servidores em nuvem Vercel e Heroku.

# 1.1 Criar a conta/fazer autenticação no Vercel

A Vercel é uma plataforma em *Cloud* onde é possível fazer o *deploy* de aplicações *frontend* de forma rápida e simples.

Primeiro, deve-se acessar o site [https://vercel.com](https://vercel.com/), escolher a opção “Sign Up” para o fazer o cadastro ou já fazer a autenticação.

A Vercel permite o login com o Github, Gitlab ou Bitbucket. No caso, o login será feito com o Github. Assim que finalizar, a Figura 1 mostra a tela inicial que deve aparecer.

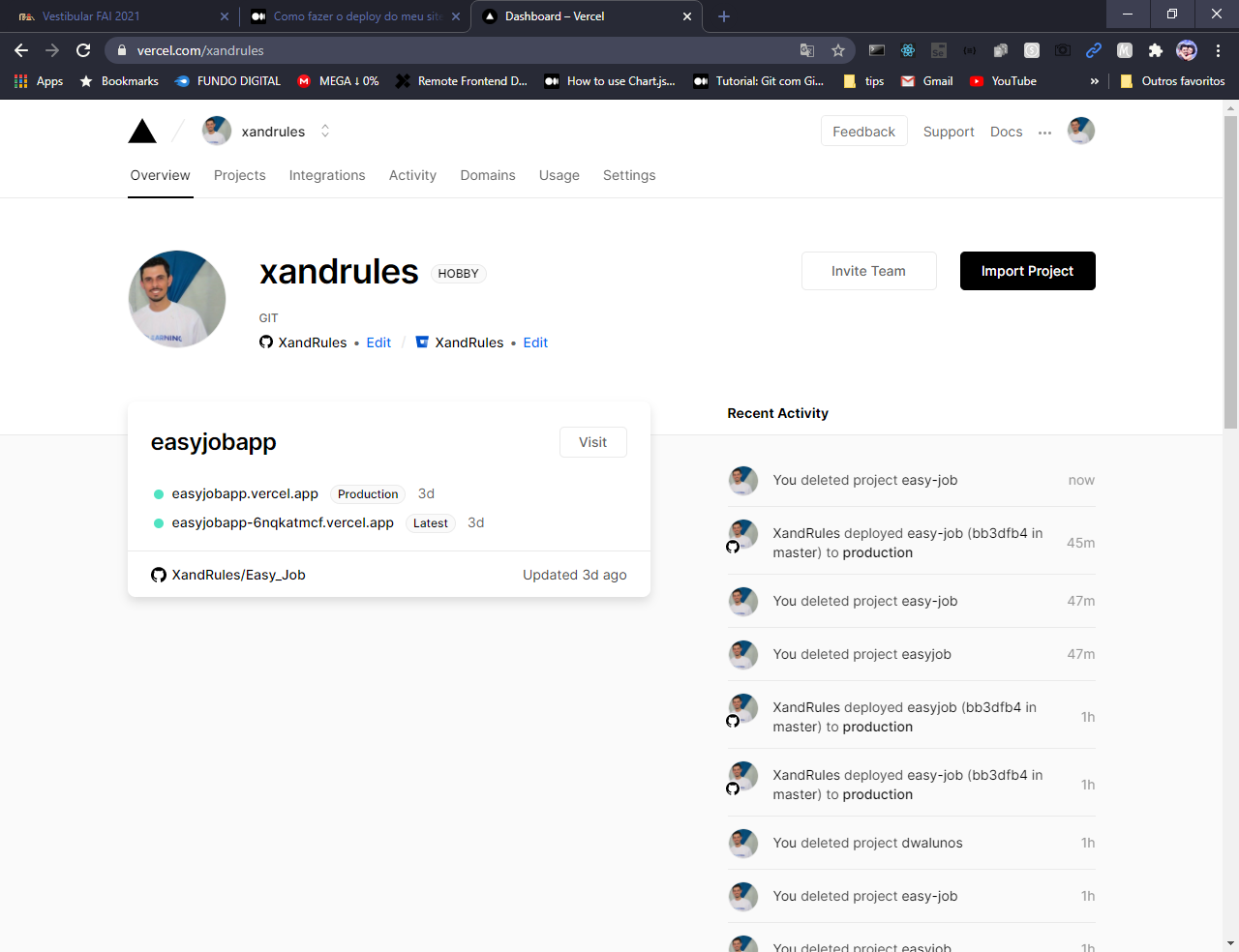


FIGURA 1 - Tela inicial da Vercel

# 1.2 Importar o projeto

Assim que a autenticação for realizada, é preciso escolher o projeto a partir do Github para fazer o *deploy*. Para isso, deve-se escolher “Import Project”. Em seguida, a opção “Continue” para importar um repositório do Git.

Agora basta informar a URL do projeto no Git e clicar em “Continue”,  novamente, conforme ilustra a Figura 2.

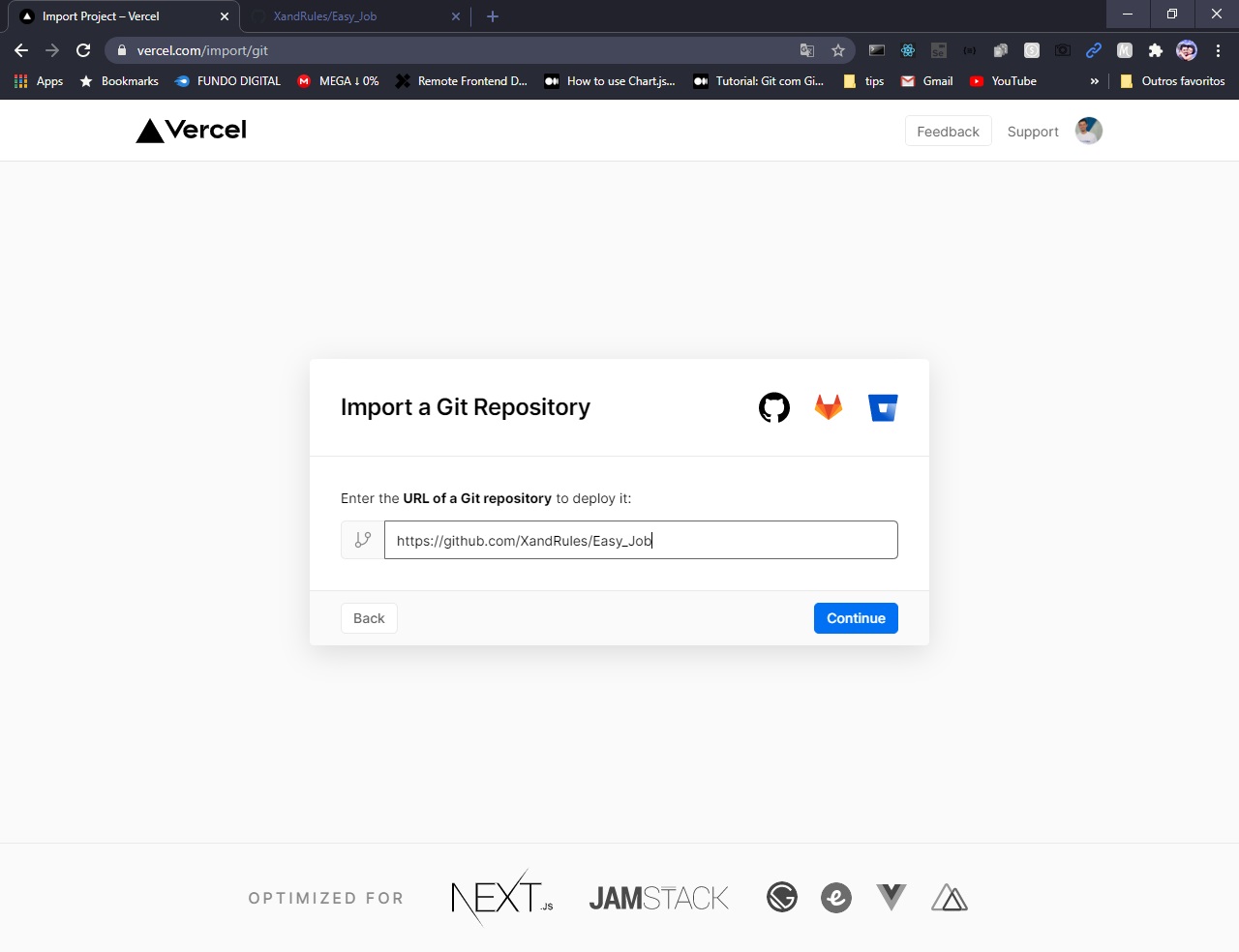


FIGURA 2 - Importar projeto do github

# 1.3 Configurar projeto

Nesta etapa, é preciso escolher o nome do projeto. Com base nele, o domínio será criado.

A Figura 3 ilustra a etapa de criação do projeto.

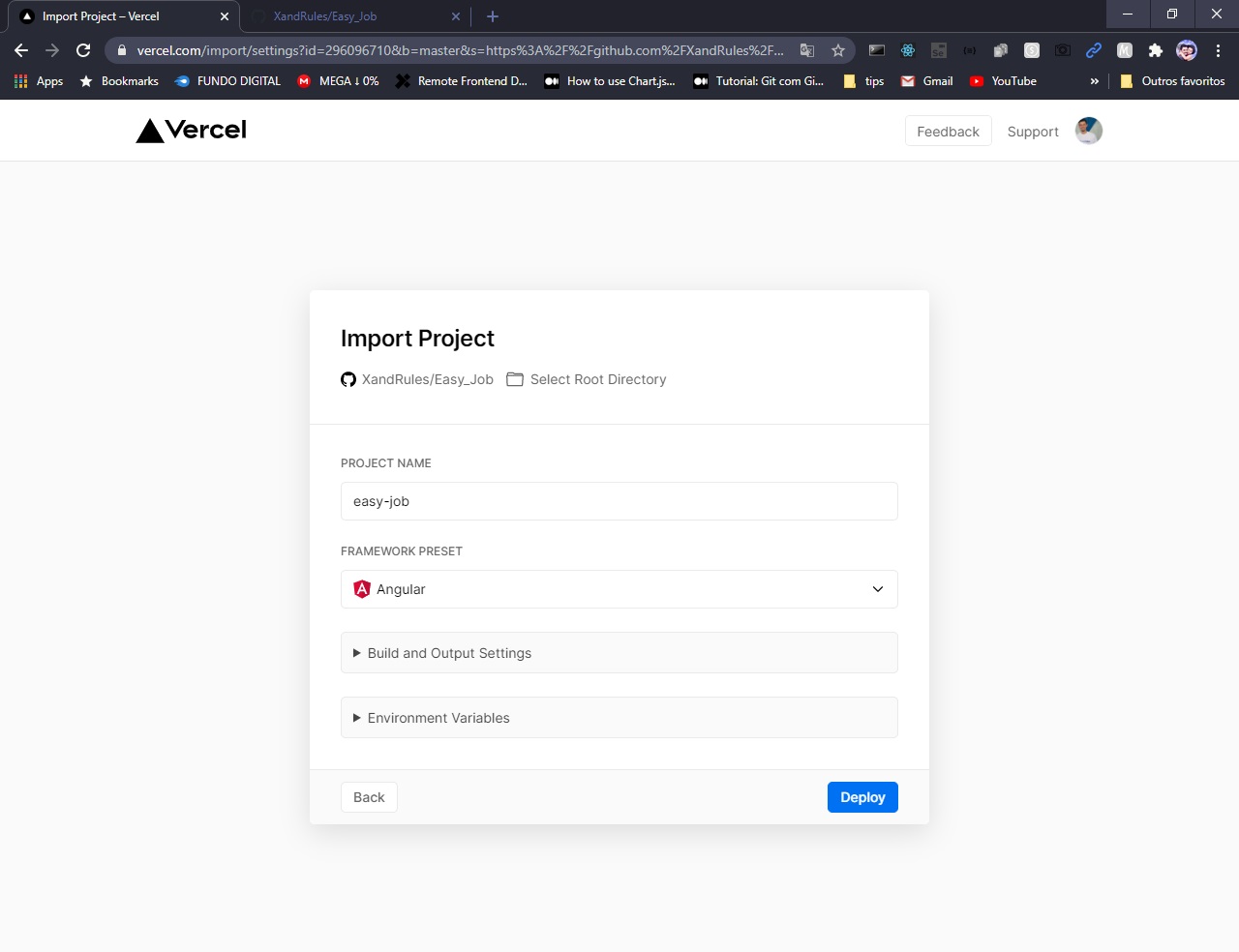


FIGURA 3 - Criação do projeto e domínio.

Após tudo configurado, deve-se escolher a opção “Deploy” e aguardar o processo de *build*. Deve aparecer uma tela como ilustra a Figura 4.

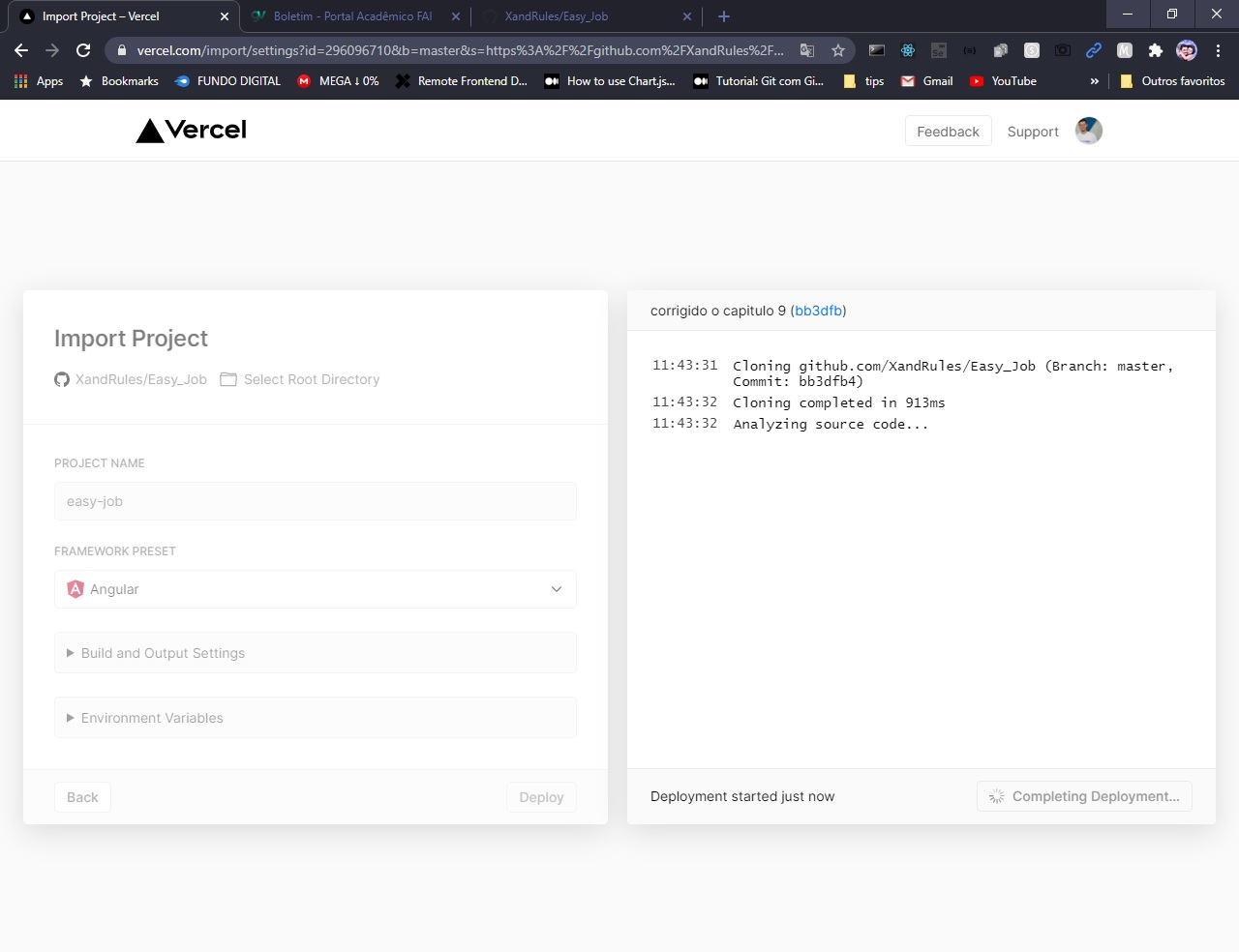


FIGURA 4 – *Deploy.*

Após finalizada essa etapa, aparecerá uma tela exibindo uma prévia do projeto que foi submetido, como ilustra a Figura 5.

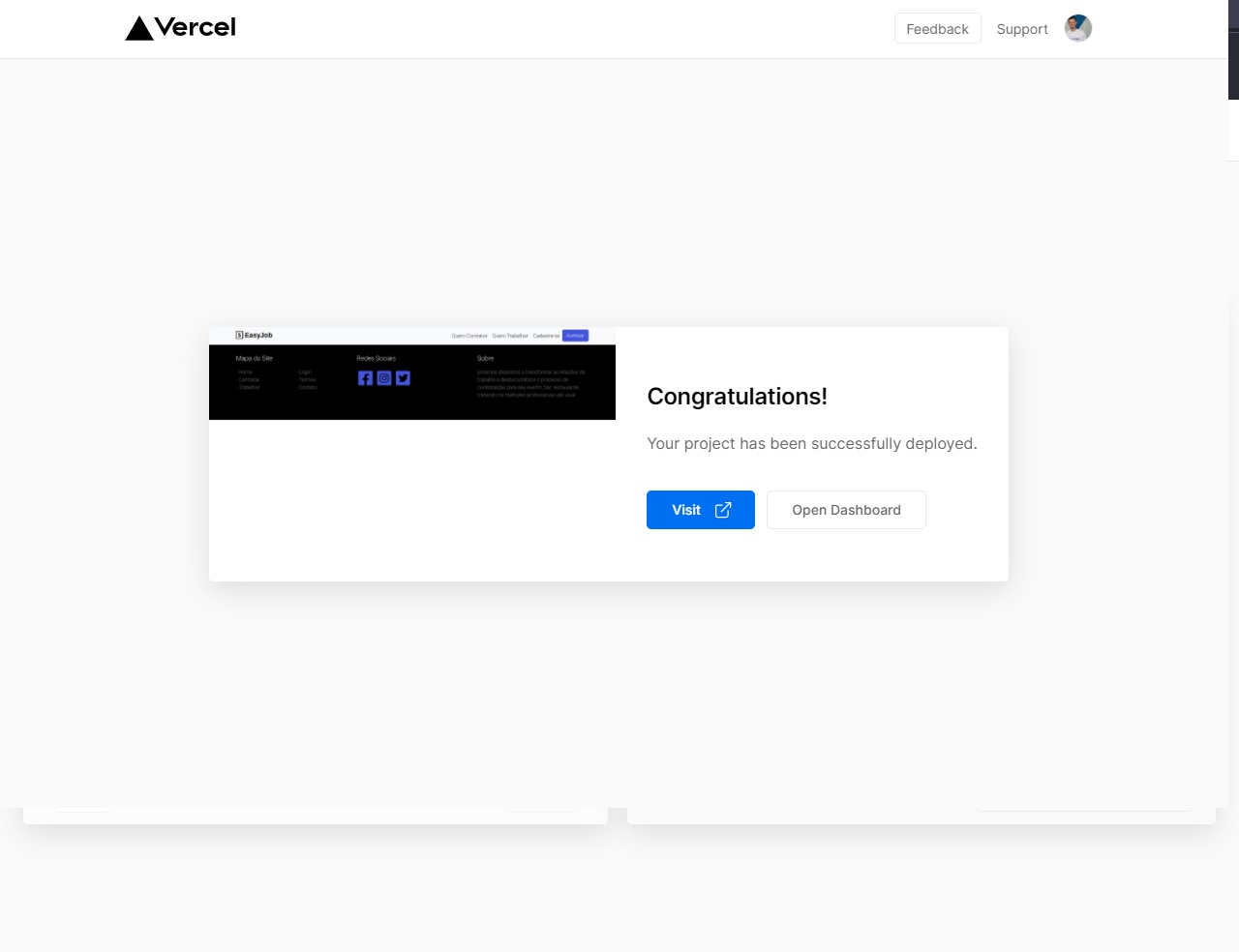


FIGURA 5 - Prévia do projeto.

# 2 IMPLANTAÇÃO *BACKEND* NA HEROKU

O Heroku é uma plataforma em nuvem, onde é possível fazer o *deploy* de aplicações e ele faz todo o gerenciamento.

Existem planos disponíveis assim como *plugins* (pagos e gratuitos) que são possíveis integrar a aplicação. No caso do *EasyJob,* o plano gratuito irá atender, apesar de suas limitações de espaço e disponibilidade, são mais que o suficiente para satisfazer as necessidades de demonstração da implantação.

# 2.1 Deploy da aplicação NodeJs

Primeiro, deve-se acessar o site <https://heroku.com>, escolher a opção “Sign Up” para o fazer o cadastro ou já fazer a autenticação de usuário.

Assim que realizar o login aparecerá uma tela como a figura a seguir.

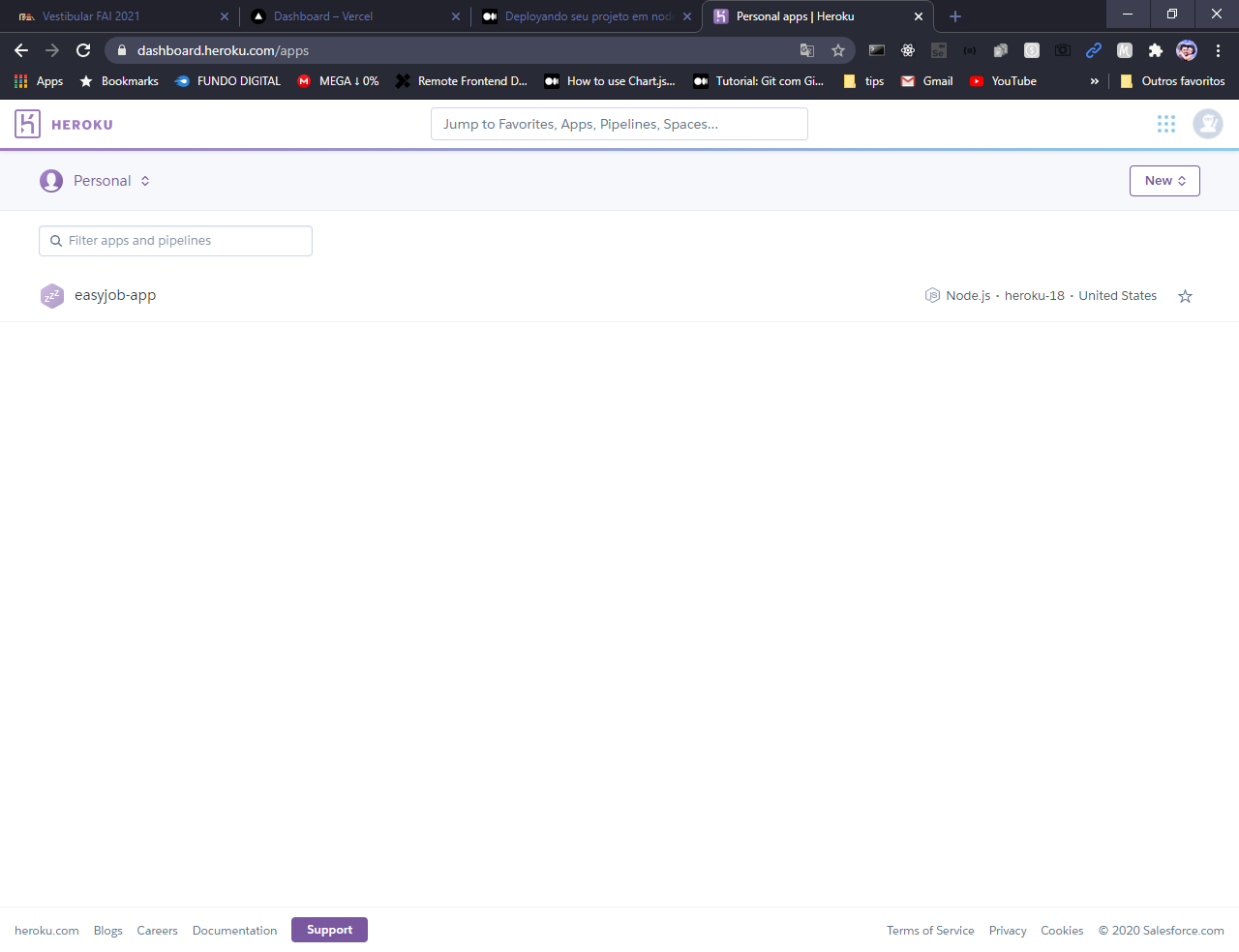


FIGURA 6 – Tela da dashboard do Heroku

Nessa tela, deve-se escolher “*New”*, em seguida a opção “*create new app*”, como ilustra a Figura 7.

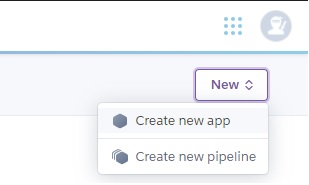


FIGURA 7 - Criar novo app.

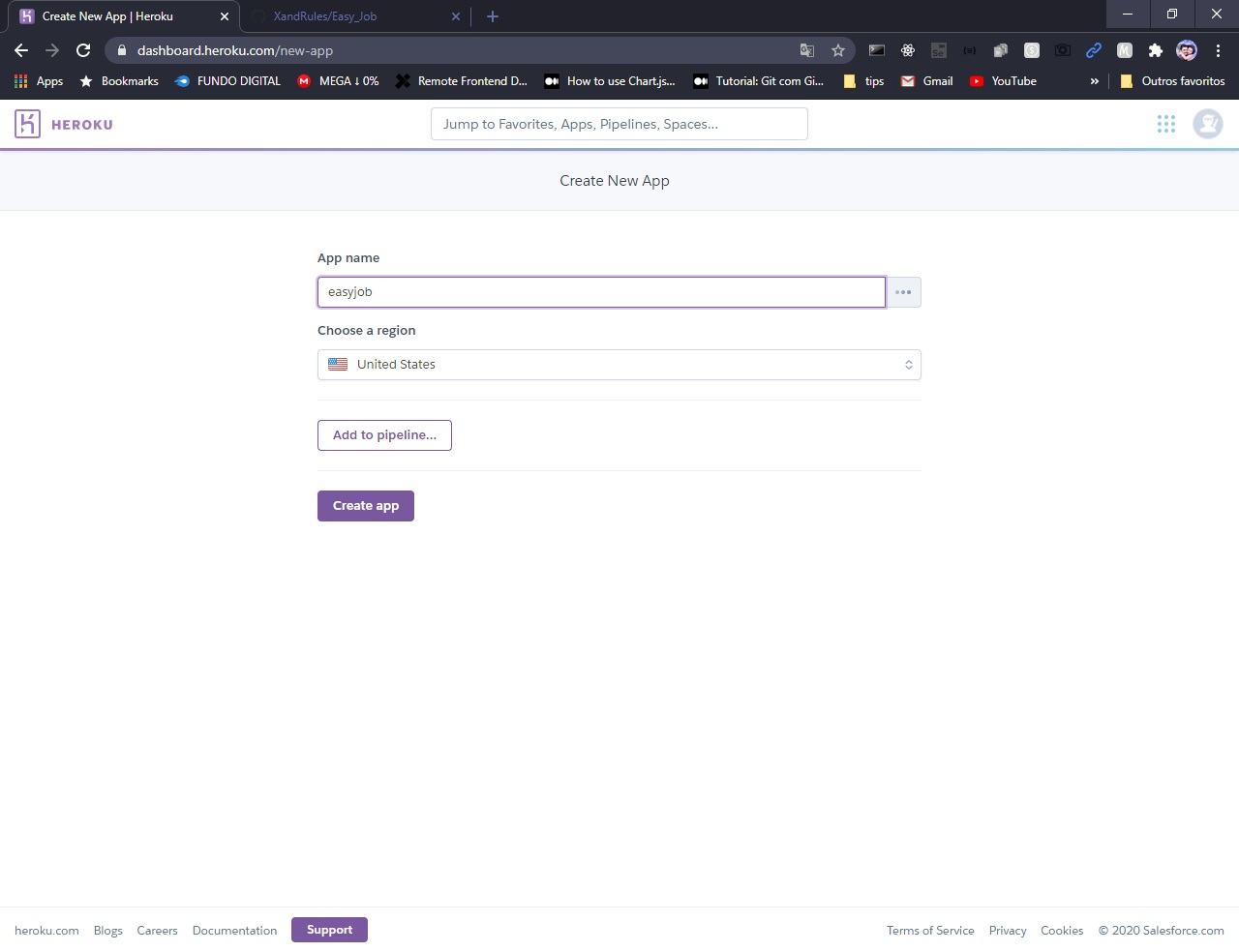
Assim que se escolhe a opção “create new app***”,*** deverá ser informado o nome do projeto e a região.  
  


FIGURA 8 - Criando projeto no servidor Heroku.

Após criar um novo projeto será necessário conectar com o repositório Github do *EasyJob*, informar o projeto que irá subir e em seguida clicar em “connect”, como ilustra a Figura 9.

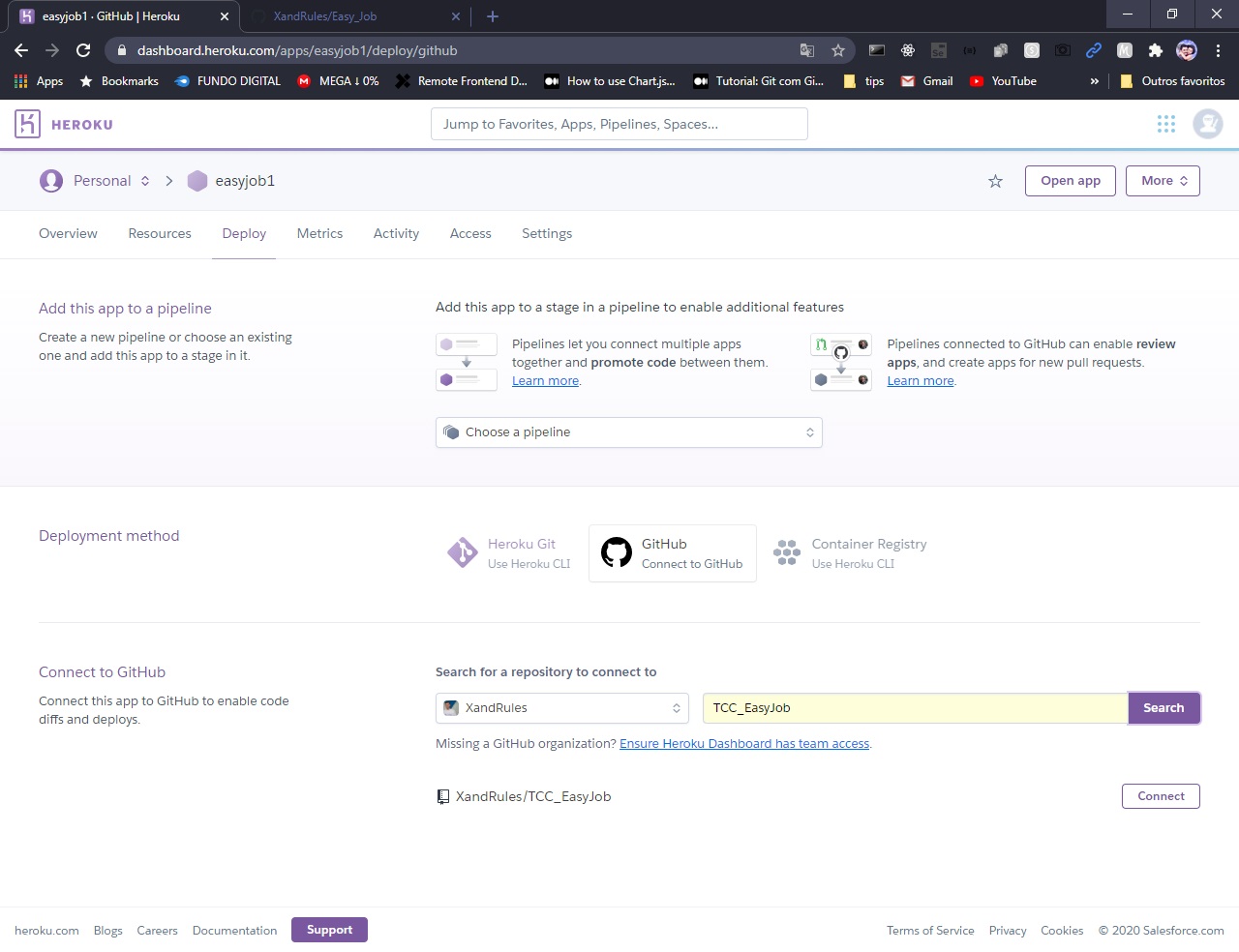


FIGURA 9 - Conectar com Github.

Após conectar o projeto no Heroku, deve-se escolher qual será a *branch* escolhida para subir o projeto. No projeto *backend* do *Easyjob* foi criado uma *branch* chamada *deploy,* onde toda alteração nessa *branch* será resultante em nova *release.*

Assim que for escolhida a *branch* , deve-se clicar em *Deploy* e aguardar o fim do processo de *build*.

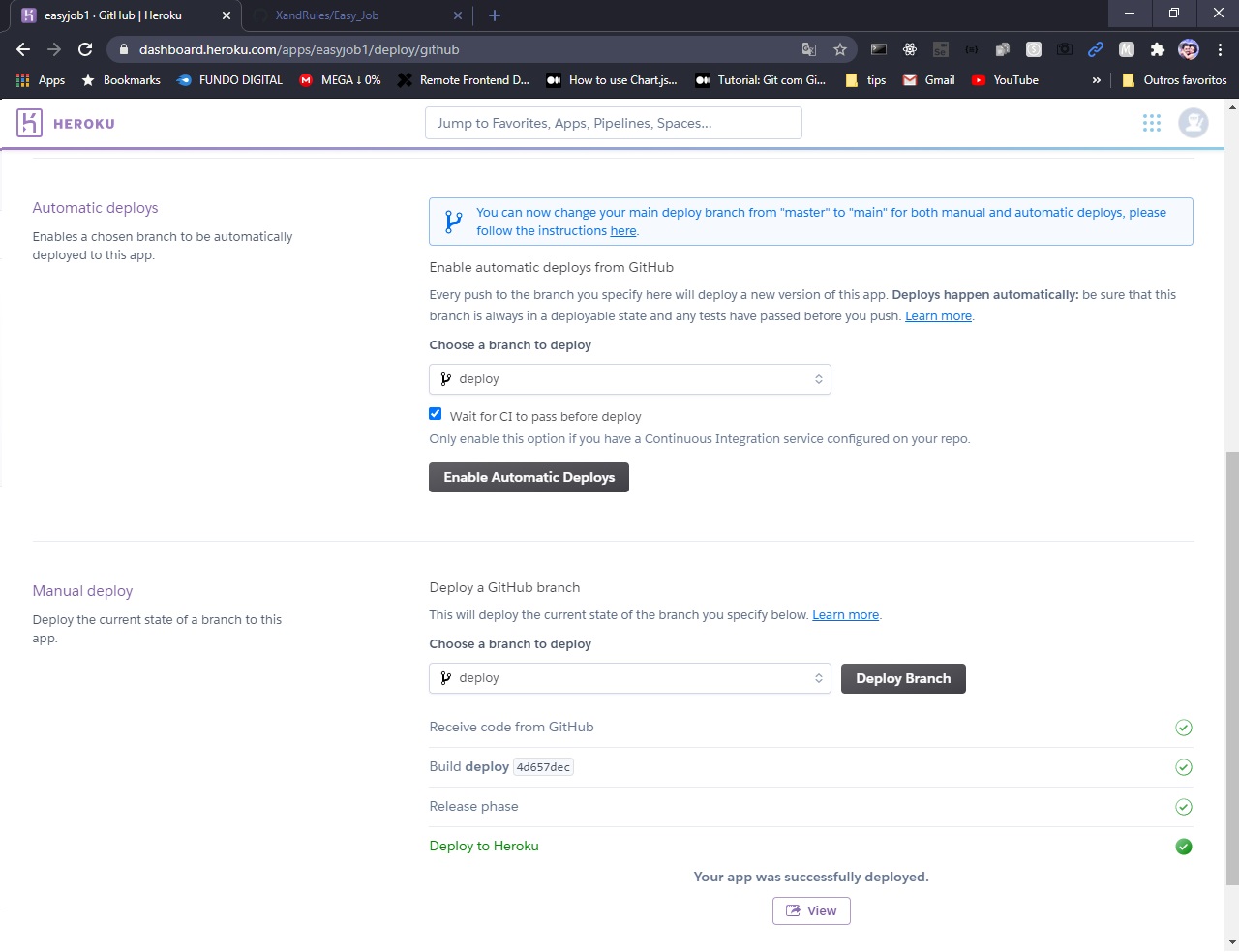


FIGURA 10 – Escolher a *branch.*

# 2.1 Criando um add-on PostgreSQL no Heroku

Após conectar o projeto ao Heroku, deve-se configurar os recursos a ser utilizados. Um deles é adicionar ao Heroku a extensão (*add-nos*) heroku-postgres.  
  
Para isso, deve-se navegar até a aba “Resources” e buscar por “postgres”, como ilustra a Figura 11.

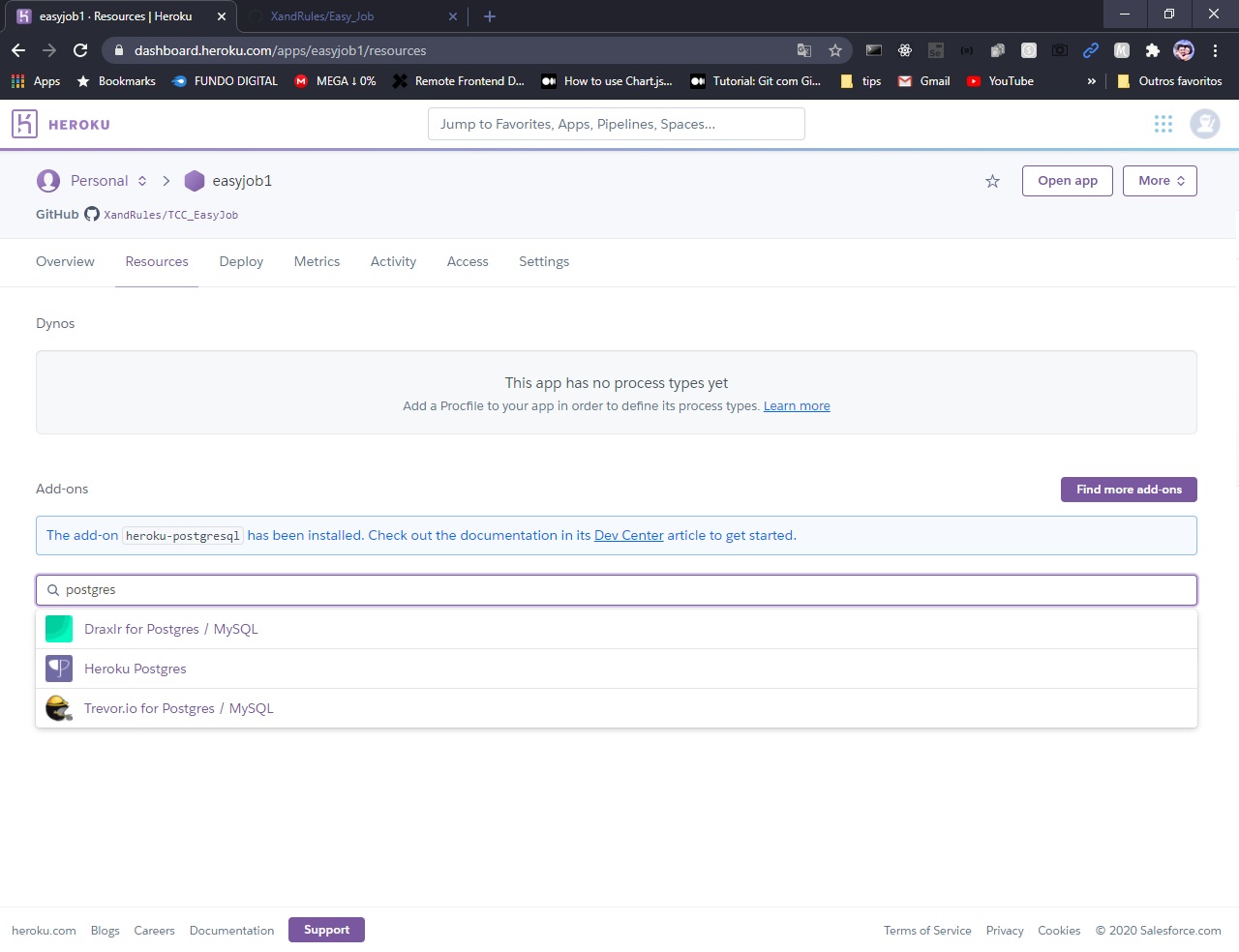


FIGURA 11 - add-ons postgres.

Depois de clicar em “*Heroku* Postgres”, basta clicar em “Submit Order Form”para adicionar ao projeto.

# 2.2 Configurando as variáveis do projeto

Depois de ter adicionado a extensão Postgres, deve-se configurar as variáveis do projeto para que seja possível a aplicação se conectar à base de dados criada.  
  
Para isso primeiro, deve-se navegar em “Resources”, depois clicar em “*Heroku postgres”*, assim que for exibido a página da extensão Heroku Postgres clicar em “*Settings”* e clicar em “*View Credentials”*.

A Figura 12 ilustra as credenciais do postgres que foi gerada.

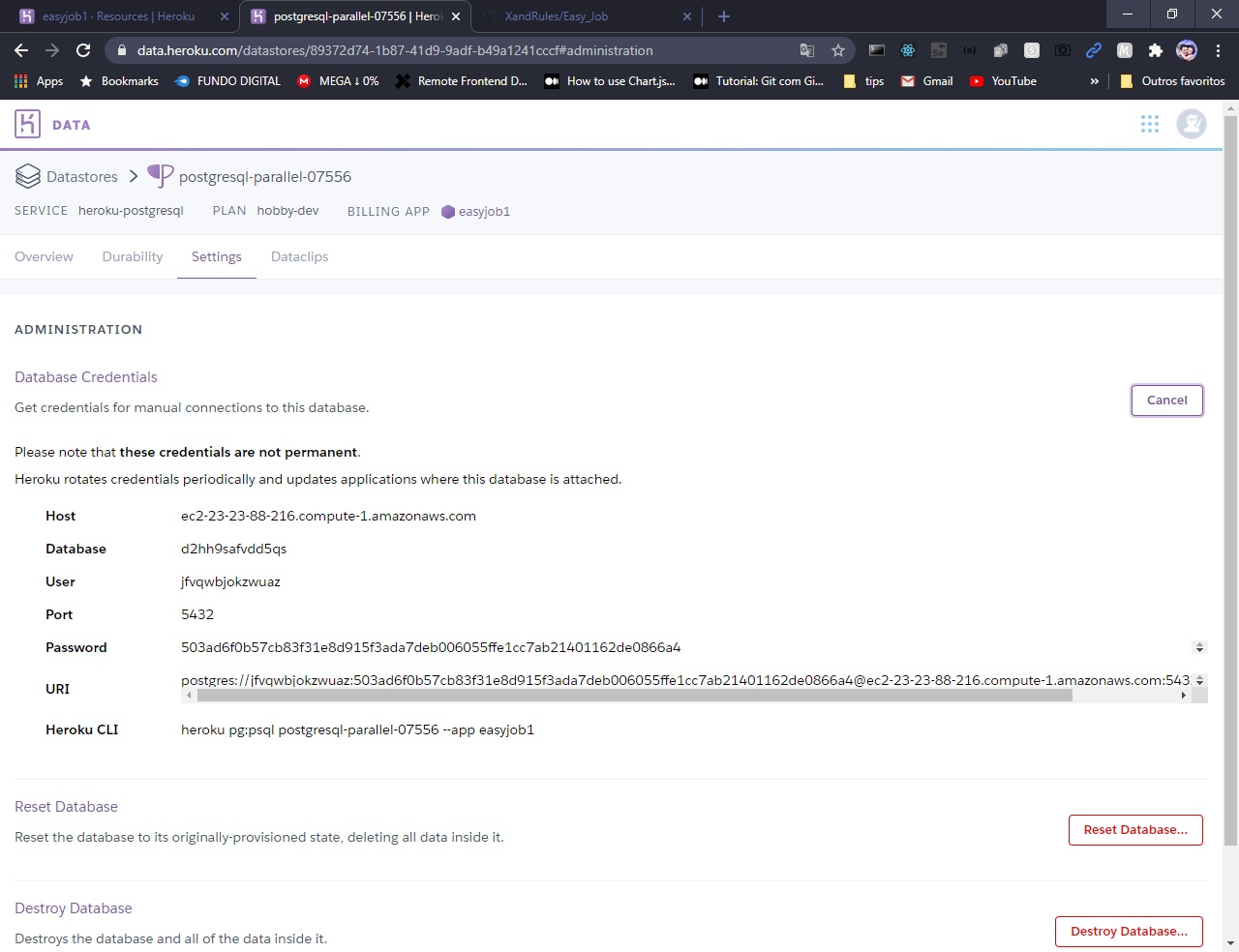


FIGURA 12 - Credenciais Postgres.

Após exibida a tela de credenciais, deve-se abrir uma nova página do projeto e clicar em “*Settings”* e em “*Reveal Config Vars”* e adicionar as variáveis do projeto *EasyJob* e seu valor, como ilustra a Figura 13.

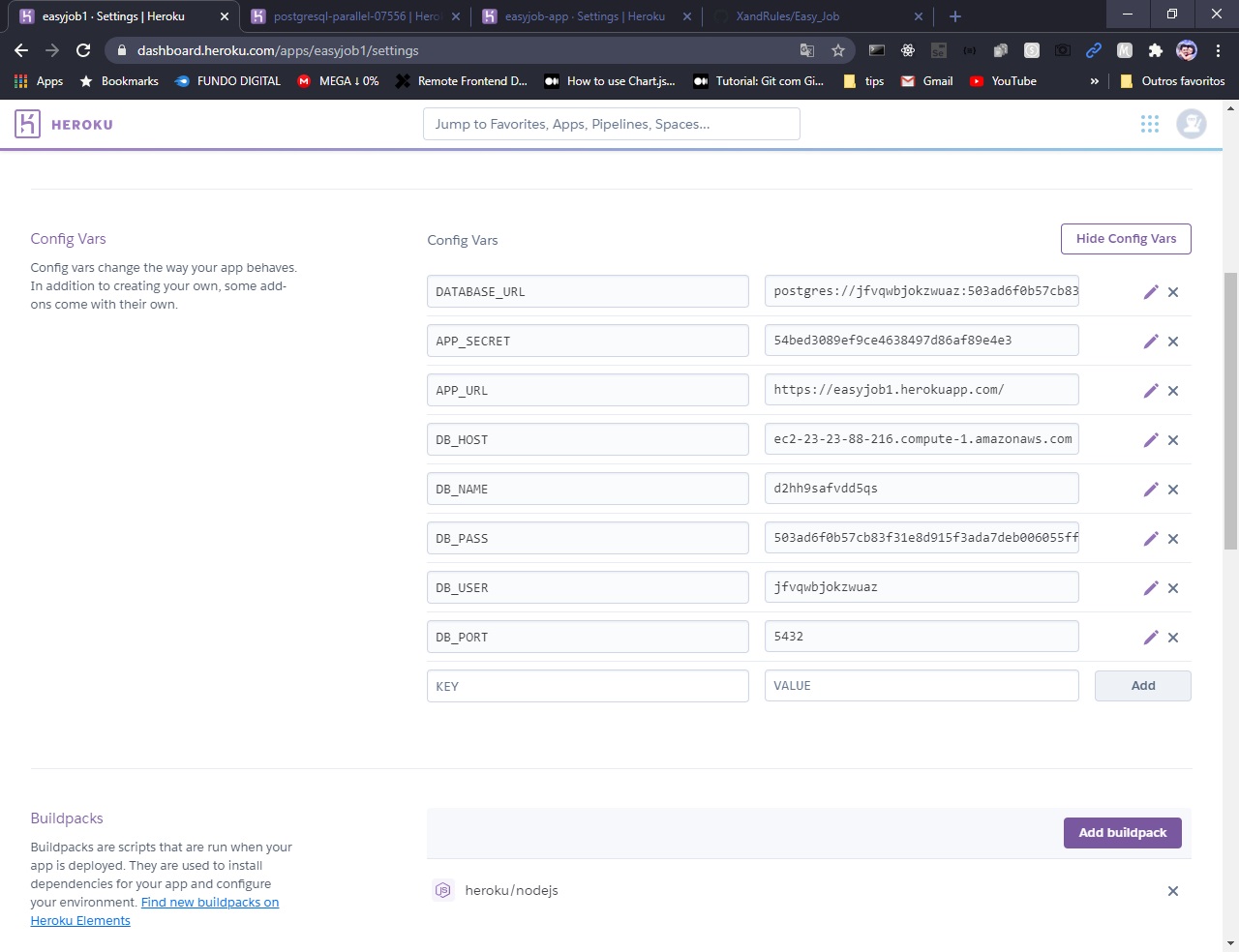


FIGURA 13 - Variáveis do projeto.

Após tudo configurado, executam-se as *migrations* para que o banco de dados “easyjob” e as tabelas do sistema serão criadas.

Para isso, deve-se navegar até “Resources” e acionar os *scripts* disponíveis .

A Figura 14 ilustra os *scripts.*

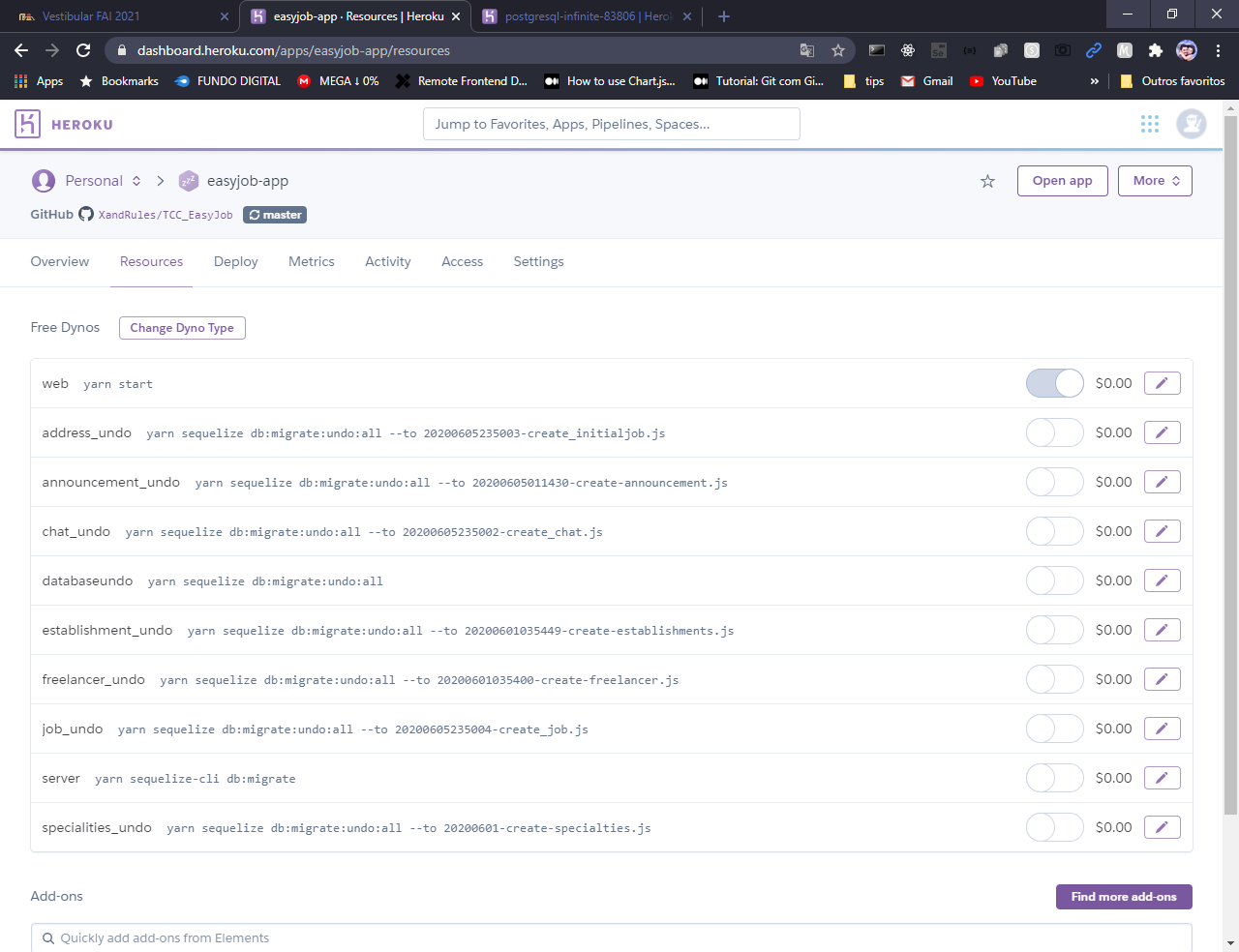


FIGURA 14 - *Scripts* do sistema

Deve-se, agora, tornar indisponível o *scripts* *web* e habilitar o script *server*, para que o sistema rode as *migrations* para criação do banco de dados e as tabelas.

A Figura 15 ilustra a execução das *migrations.*

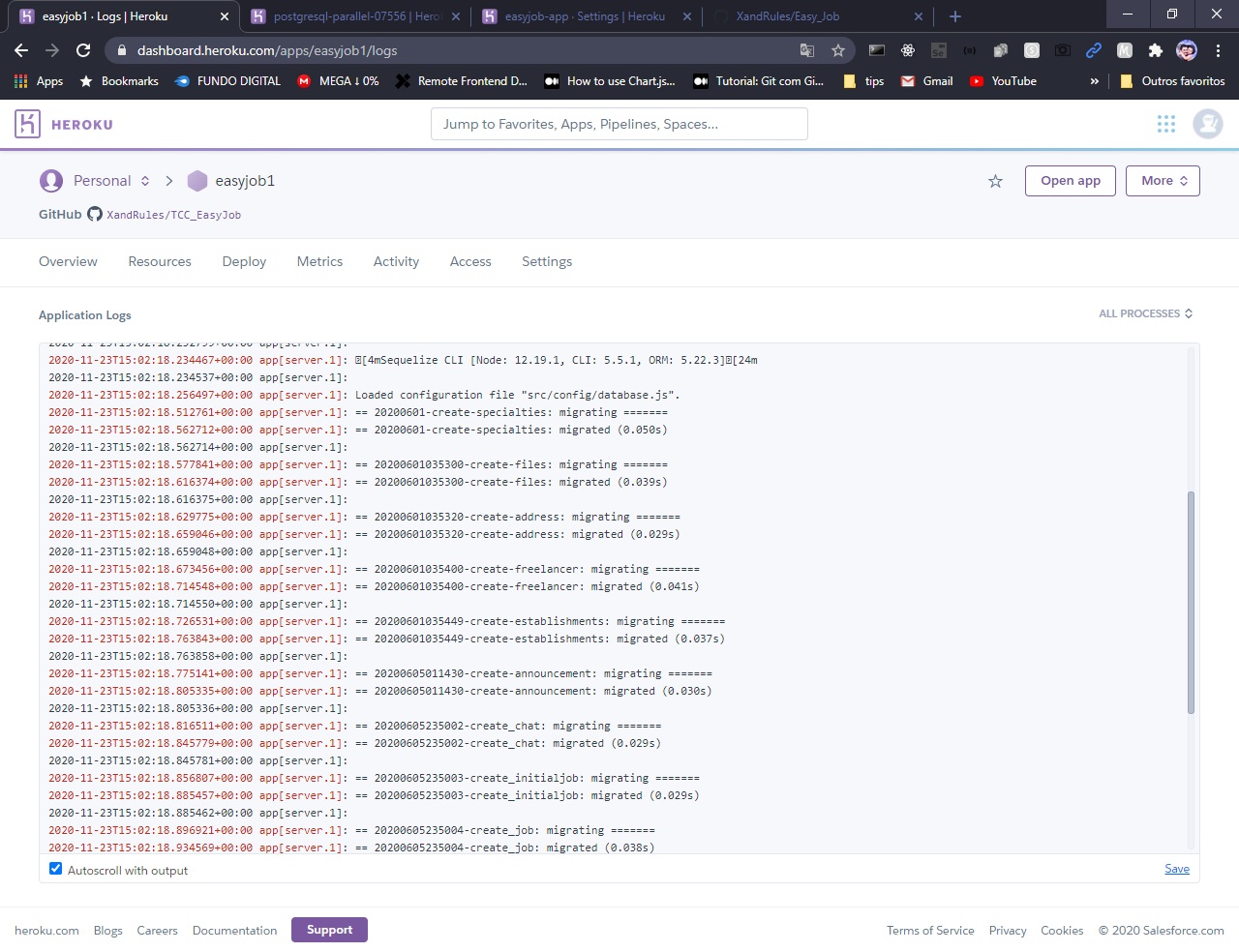


FIGURA 15 - Execução das *migrations*

Depois das *migrations* criadas, deve-se tornar indisponível o *script* ***server*** e habilitar o script ***web*** novamente.