**Computação Paralela, Distribuída e em Nuvem**

Plataforma Heroku

**03 de Junho de 2019  
Trabalho - Computação Paralela na plataforma Heroku**

Rafael Eduardo Feliciano Cassiolato, 81614081  
Luis Felipe Rodrigues Lima, 816126336   
Bruno Teixeira da Gama, 81614138   
Raiza Morata, 817124676  
Guilherme Sant’Clair Alves dos Santos, 81610373

[**1. O que é Heroku?**](#_9di17v9trknm) **3**

[**2. Tipo de Serviço na Plataforma Heroku**](#_pfpq8prqowd7) **3**

[**3. Deploy no Heroku**](#_i048psjwx7pl) **3**

[**4. Vantagens**](#_wqcq3i2i2bw8) **4**

[**5. Desvantagens**](#_996dffbm108h) **4**

[**6. Dynos**](#_drjgr6jx2e0a) **4**

[**7. Linguagens Suportadas**](#_v3awmy1pvyuu) **5**

[**8. Infraestrutura**](#_fcs8ps8wlvx3) **5**

[**9. Market Share**](#_ttv232xz4zha) **6**

[9.1. Market Share por Hospedagem de Sites](#_ud37rb70eqyp) 6

[9.2. Plataformas por Market Share](#_38s4grg71pu2) 7

[**10. Visão do Mercado sobre a Plataforma Heroku**](#_nh1yf9694vff) **8**

[**10.1. Comentários de seus clientes**](#_gfzc97nh4ju2) **8**

[10.1.1. Toyota](#_kvxoa9htqgrj) 8

[10.1.2. Bonobos](#_gsqz4x8s857d) 8

[10.1.3. Elycio Talen](#_eir0xll21v4v) 8

[10.1.4. Forever Living](#_qbmg3z9sht2m) 8

[**10.2. Aplicações que Usam Heroku**](#_htnq6ypnq8oh) **9**

[10.2.1. Adim Adim](#_ploypivsantg) 9

[10.2.2. Invisalign](#_455fg91uc3i1) 9

[10.2.3. AltSchool](#_ch6bvqrvvcb8) 9

[**13. Add-ons**](#_wbgsmvrg003u) **9**

[**14. Planos do Heroku**](#_rojlu9fd2co0) **11**

[**15. Persistência de Dados**](#_eftseezxgce) **12**

[**16. Limites Gerais**](#_zegkunmh4y1y) **13**

[**17. Bibliografia**](#_5685j4rl3us) **13**

## 1. O que é Heroku?

O Heroku é uma plataforma de serviços cloud que oferece um modelo de negócio **PaaS**. Ele permite que você hospede suas aplicações em um ambiente facilmente escalável e com suporte a várias tecnologias. O Heroku tem um plano gratuito, indicado para testes, possuindo diversas restrições e opções pagas com mais funcionalidades e suporte técnico.

## 2. Tipo de Serviço na Plataforma Heroku

O Heroku é um **PaaS**, ou seja, um produto de Plataforma como Serviço. Isso significa que você pode fazer deploy de seus serviços sem se preocupar com configurações de hardware e sistema operacional. O seu ponto principal de venda é o foco na sua aplicação, deixando toda a complexidade da infraestrutura e escalabilidade para a equipe da própria plataforma. Isso significa mais tempo de desenvolvimento na aplicação e nos seus componentes (Banco de Dados, filas, logs) e menos tempo pensando em conteiners, sistema operacional, hardware, deploys, ambientes de rede e afins.

## 3. Deploy no Heroku

Existem três principais formas de realizar deploy no Heroku. As mais simples são utilizando o heroku-cli (Interface de linha de comando do Heroku), git e Dropbox.

Durante nosso projeto, utilizamos deploy pelo git, por já hospedarmos o projeto no github para controle de versões e ambas as ferramentas possuírem uma ótima integração.

O Heroku faz um cache do framework de execução (versão do nodeJS, Java, python, etc), dessa forma o ambiente estará ready-to-go em qualquer versão de execução da sua linguagem, bastando indicar por meios dos arquivos de configuração os requisitos e linguagem utilizados. (Exemplo: requirements do python, package.json do nodeJS)

Também é realizado um cache das dependências do projeto, dessa forma ele somente reconfigura as dependências alteradas, e apenas quando necessário, podendo ser realizado, inclusive, um downgrade de dependência.

Para realizar o deploy no Heroku utilizando o github, basta acoplar seu repositório ao seu projeto Heroku, e toda vez que for sincronizado qualquer commit ao repositório, o Heroku irá realizar as configurações devidas e atualizar o ambiente de deploy.

## 4. Vantagens

As vantagens do Heroku sobre outros serviços de PaaS são inúmeras. Citamos as mais pontuais e importantes considerando o seu foco de venda, que é a menor preocupação com o ambiente físico e a camada de sistema operacional.

A ferramenta de interface de linha de comandos do Heroku, é um pacote de software que permite iniciar e criar aplicações, configurar uma integração contínua GIT, realizar deploys, abrir a aplicação, adicionar add-ons e muitos outros recursos úteis por um único meio, podendo dessa forma serem criadas diversas automações.

O Heroku também possui um forte mercado de add-ons, o que simplifica o processo de desenvolvimento pois muitas vezes podemos encontrar uma solução pronta para um problema comum.

Por exemplo, digamos que você gostaria de adicionar um banco de dados ao seu app. Analogamente a como instalar um novo plug-in em um site Wordpress, o desenvolvedor somente tem de acessar o Heroku Marketplace e clicar no serviço de banco de dados desejado. Essa integração pré-projetada reduz a carga de “horas-homem” dos desenvolvedores, que não precisam criar a ponte entre o serviço de banco de dados e a aplicação em questão.

O Heroku, de muitas formas, simplifica o processo de desenvolvimento, oferecendo grande valor para aplicações que devem ser desenvolvidas rapidamente. Como comparativo, um concorrente como o Google App Engine, não oferece as mesmas comodidades de forma tão simplificada.

## 5. Desvantagens

A principal desvantagem vista pelo mercado é o preço do serviço. A solução Heroku é considerada uma plataforma cara para aplicações com alto tráfego de dados. Outra desvantagem ao contratante é que ele mesmo precisa escalar sua aplicação manualmente.

Ainda como desvantagem ou divergência restritiva, há a restrição de que não é possível realizar uma conexão via SSH em ambientes criados no Heroku. Essa característica tem relação de causalidade com o modelo de discos efêmeros usados pela solução.

## 6. Dynos

Os dynos são definidos como containers leves que permitem que o desenvolvedor execute sua aplicação em um ambiente isolado e seguro, com suas dependências presas ao dyno.

Um projeto Heroku pode possuir vários dynos, inclusive de linguagens diferentes. De acordo com a tarefa, os dynos podem ser classificados em três tipos:

* **Web Dynos:** Responsáveis por receber e atender as requisições web. Cada aplicação pode ter no máximo um dyno desse tipo;
* **Worker Dynos:** Executam tarefas em background, sendo recomendado para processamentos mais pesados e assíncronos. Cada aplicação pode ter vários workers;
* **One-Off Dynos:** São criados apenas para uma tarefa eventual e logo são destruídos;

Em relação ao porte de computação dos dynos, o cliente tem diversas opções de escolha com diferentes capacidades de memória RAM.

## 7. Linguagens Suportadas

Inicialmente a ferramenta suportava apenas a linguagem Ruby, porém hoje em dia já é possível utilizar as seguintes linguagens:

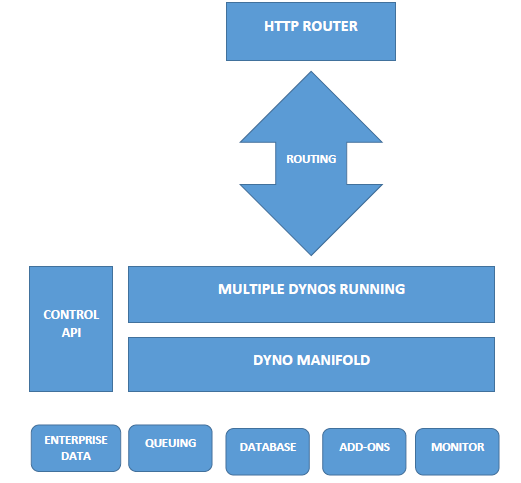
* Java;
* Node.js;
* Scala;
* Clojure;
* PHP;
* Go;
* Python;

Pode-se dizer que o Heroku é poliglota por ele ter a capacidade de executar diversas aplicações de forma semelhantes e em várias linguagens diferentes. Todas as linguagens rodam em um dyno, e junto do seu repositório é necessário um arquivo que indica qual a linguagem e o ponto de entrada da sua aplicação.

## 8. Infraestrutura

O Heroku tem sua sede em São Francisco, Califórnia - Estados Unidos da América, e possui diversos datacenters espalhados pelo globo.

Alguns exemplos são: Dublin, Frankfurt, Oregon, Tokyo. Não foram encontradas informações sobre como são distribuídos seus servidores físicos, sistema operacional utilizado, capacidade de cada máquina física e afins.

Fonte: <https://commons.wikimedia.org/wiki/File:ArchitectureHeroku.png>

## 9. Market Share

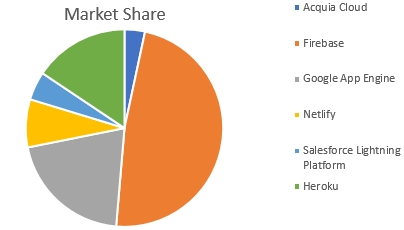
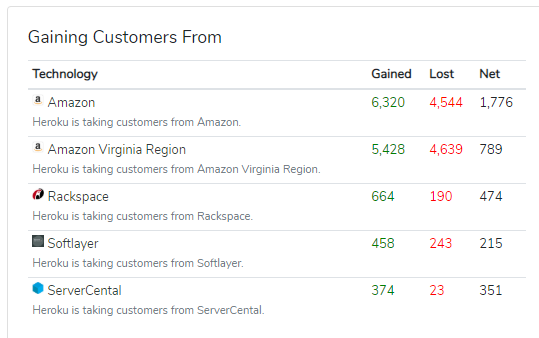
A respeito de Market Share, podemos analisar o espaço ocupado pela plataforma Heroku com os seguintes dados:

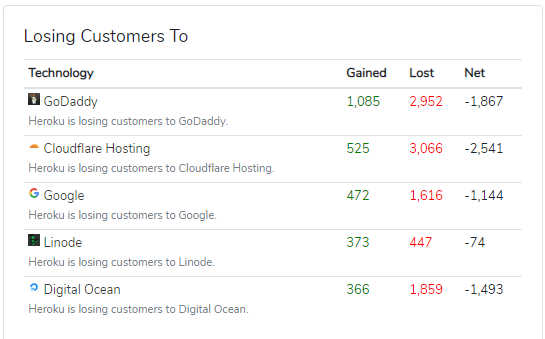
### 9.1. Market Share por Hospedagem de Sites

|  |  |  |
| --- | --- | --- |
| Plataforma | Websites | Market Share |
| Firebase | 134393 | 46,30% |
| Google App Engine | 57405 | 19,78% |
| Netlify | 22000 | 7,58% |
| Salesforce Lightning Platform | 13070 | 4,50% |
| Acquia Cloud | 9390 | 3,23% |
| Heroku | 43714 | 15,06% |

Fonte: <https://www.datanyze.com/market-share/paas/heroku-market-share>

### 9.2. Plataformas por Market Share

Fonte: <https://trends.builtwith.com/hosting/Heroku/Market-Share>

Fonte: <https://trends.builtwith.com/hosting/Heroku/Market-Share>

## 10. Visão do Mercado sobre a Plataforma Heroku

O Heroku é totalmente gerenciado, dando aos desenvolvedores a liberdade de se concentrar em seu produto principal sem a distração de manter servidores, hardware ou infraestrutura. A experiência da Heroku fornece serviços, ferramentas, fluxos de trabalho e suporte poliglota - todos projetados para aumentar a produtividade do desenvolvedor.

### 10.1. Comentários de seus clientes

#### 10.1.1. Toyota

“A Toyota Motor Europe possui um portfólio amplo e complexo de mais de 40 aplicativos interconectados voltados para o cliente. Contamos com o Heroku como plataforma para esses aplicativos de missão crítica, que está nos ajudando a transformar a forma como nos conectamos e nos engajamos com nossos clientes.”

#### 10.1.2. Bonobos

“O Heroku tornou nossos processos muito mais rápidos. Agora é muito mais fácil para qualquer um de nossos engenheiros realizar tarefas como: depurar, acessar registros, acessar a réplica de leitura ou obter informações sobre o state de aplicações.”

#### 10.1.3. Elycio Talen

“O Heroku Connect é a cola que une nossos aplicativos voltados ao cliente com nosso back-end de administração. Os dados são sincronizados sem interrupções.”

#### 10.1.4. Forever Living

“Com o Heroku Connect, conseguimos criar e implantar nossos aplicativos em mais de 150 países em menos de 90 dias. A Heroku está ajudando milhares de pessoas em toda a nossa empresa a analisar e agir com base em dados globais em tempo real. Heroku transformou a maneira como fazemos negócios e nos permite fazer qualquer coisa que possamos imaginar!”

### 10.2. Aplicações que Usam Heroku

#### 10.2.1. Adim Adim

Empresa que envolve corretores e doadores. Recebe doações online baseadas em cartões de crédito que encaminham para as contas bancárias das ONGs apropriadas.

Ela usa PHP, que possui suporte pela Heroku, o que tornou suas implementações muito mais eficientes, segundo relato.

#### 10.2.2. Invisalign

A Invisalign executa vários aplicativos voltados ao consumidor, desde o site corporativo até os sites de produtos voltados para o consumidor. Para o Invisalign, tornou-se cada vez mais difícil realizar atualizações rápidas, e essa limitação era desafiadora não apenas para as equipes de negócios, mas também para os desenvolvedores de TI.

Escolheram a solução PaaS da Heroku e não precisam se preocupar mais com patches, vulnerabilidades de segurança e adicionar armazenamento. A plataforma é gerenciada e suportada pela equipe Heroku.

#### 10.2.3. AltSchool

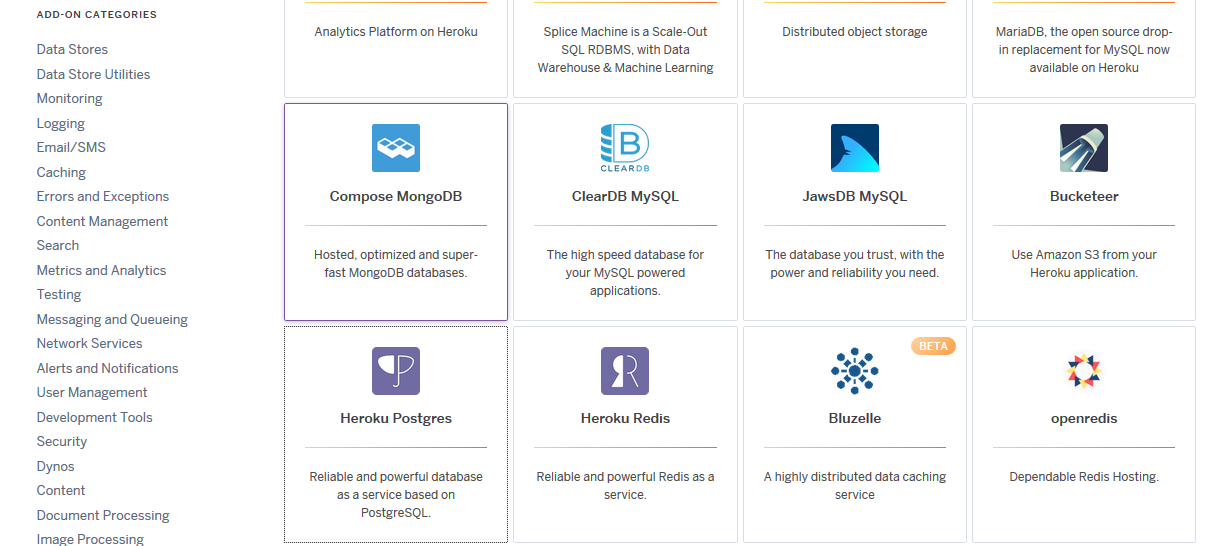
A plataforma facilitou a criação e implantação de novos aplicativos. Para criar seus aplicativos, a equipe usou [Python](https://www.heroku.com/python) no lado do servidor e [Node.js](https://www.heroku.com/nodejs) para o front-end com um pouco de [Go](https://www.heroku.com/go). Eles usam vários [Complementos Heroku](https://elements.heroku.com/addons) para estender sua implementação, incluindo [SSL](https://elements.heroku.com/addons/ssl) para criptografia, [Heroku Scheduler](https://elements.heroku.com/addons/scheduler) para agendamento assíncrono, [Implantação de Ganchos](https://elements.heroku.com/addons/deployhooks) para notificação de implantação, [Redis To Go](https://elements.heroku.com/addons/redistogo) para gerenciar sua instância Redis, [New Relic APM](https://elements.heroku.com/addons/newrelic) e [Papertrail](https://elements.heroku.com/addons/papertrail) para gerenciamento de log, [SendGrid](https://elements.heroku.com/addons/sendgrid) para entrega de e-mail, o serviço [MemCachier](https://elements.heroku.com/addons/memcachier) e o [Airbrake Bug Tracker](https://elements.heroku.com/addons/airbrake).

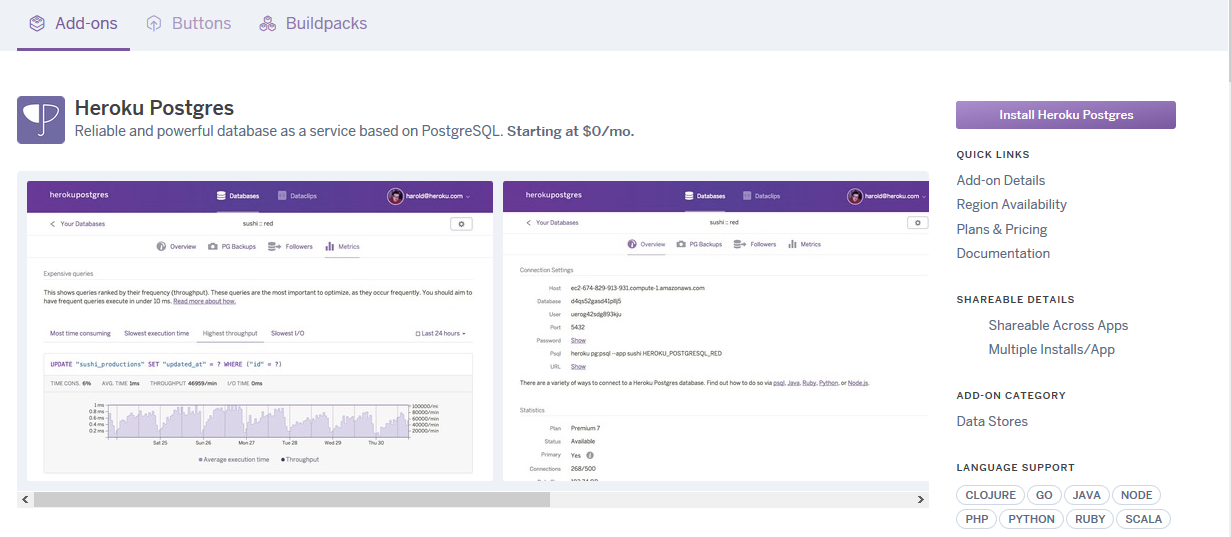
## 13. Add-ons

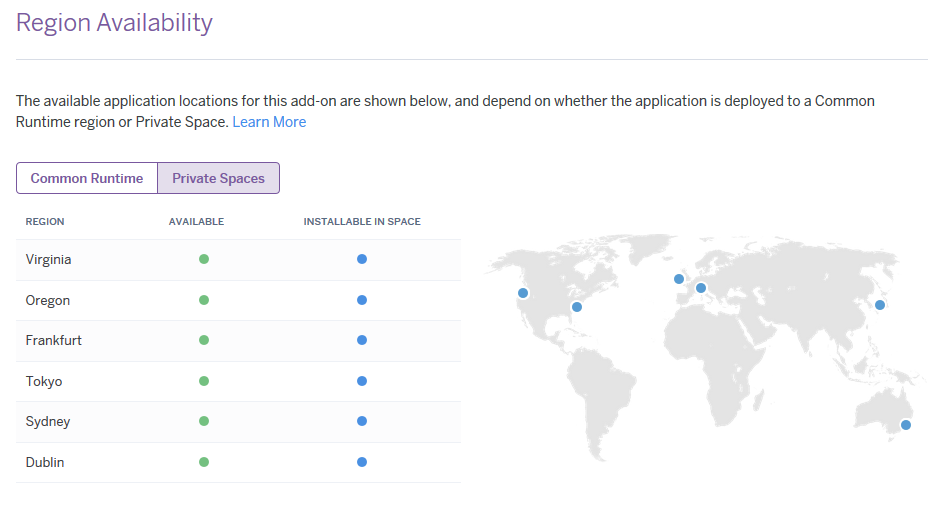
O heroku possui um forte mercado de add-ons que facilitam a vida do desenvolvedor.

Esses add-ons são componentes que dão suporte a aplicação desenvolvida, como monitoramento, armazenamento de dados em banco, processamento e mais. É possível compartilhar um add-on entre apps.

Algumas imagens do ambiente de add-ons:

Fonte: <https://elements.heroku.com/addons> 

Fonte: <https://elements.heroku.com/addons/heroku-postgresql> 

Fonte: <https://elements.heroku.com/addons/heroku-postgresql>

## 14. Planos do Heroku

O Heroku utiliza planos baseados na quantidade de dynos que você utiliza. Abaixo temos uma imagem informativa da Heroku, com recursos e seus respectivos preços.

Fonte: <https://www.heroku.com/pricing> 

Em resumo, os planos da Heroku são:

* **Plano Free** 
  + dynos adormecem após 30 minutos de inatividade;
  + Utiliza um pool de dyno-hours;
    - Não verificada: 550 horas;
    - Verificada: 1000 horas;
  + 512MB RAM;
  + Dynos: 1/1 (worker/web);
* **Plano Hobby**
  + Dyno nunca adormece;
  + SSL Gratuito e Gerenciamento automático de certificados;
  + Métricas da aplicação;
  + Pode ter diversos dynos workers;
  + 512MB RAM;
  + 10 tipos de processo;
  + $7 dólares / dyno / mês;
* **Plano Standard**
  + Escalabilidade horizontal simples;
  + Alertas por eventos;
  + Métricas de runtime próprios da linguagem;
  + 512MB ou 1GB RAM;
  + $25 ~ $500 dólares / dyno / mês;
* **Plano Performance**
  + Servidor dedicado;
  + Autoescalonamento;
  + 2.5GB ou 14GB RAM;
  + $25~$500 dólares / dyno / mês;

## 15. Persistência de Dados

O Heroku se utiliza da tecnologia de disco do tipo efêmero, o que significa que você pode escrever arquivos para o disco, mas após a aplicação ser reiniciada, esses arquivos não serão persistidos.

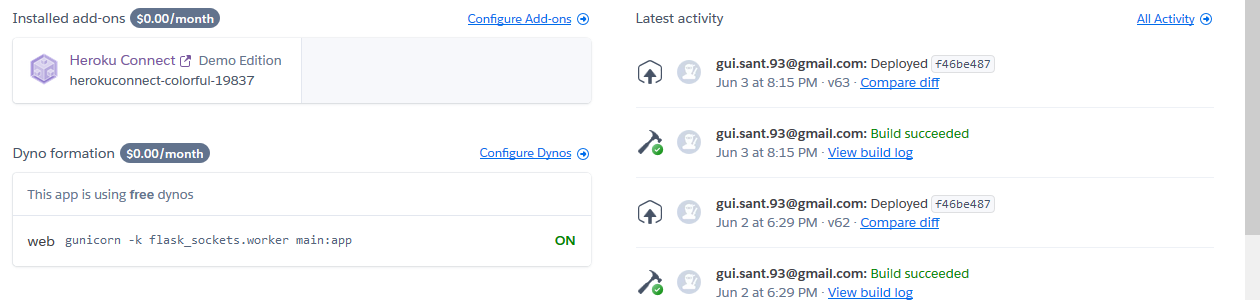
Segundo a documentação do próprio Heroku, nos eventuais casos em que se fizer necessário criar persistir dados de disco e armazenamento interno que não sejam relacionados a banco de dados, a melhor prática para persistência dos dados é utilizar um serviço de cloud de armazenamento como o Amazon S3.

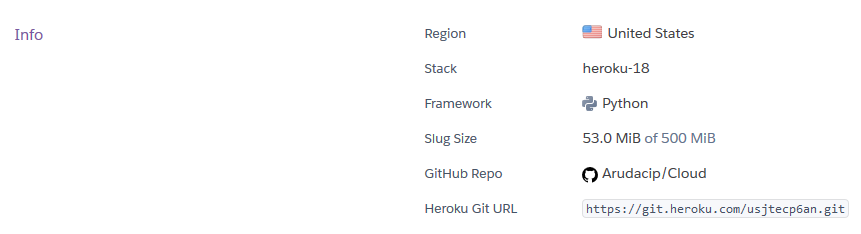
## 16. Limites Gerais

Em limites gerais do serviço, independentes do plano adquirido e voltados para qualquer cliente, temos:

* As variáveis de configuração (**config vars**) podem ter um tamanho máximo de 32Kb por app.
* Só podem ser realizadas 75 requests de compilação por hora.
* Só é possível utilizar add-ons com uma conta comprovada.
* Só é possível ter até 500 membros colaboradores por conta corporativa, e 25 colaboradores por conta de pessoa física.
* Usuários comprovados podem ter no máximo 100 aplicativos. Já usuários não comprovados podem ter até 5.

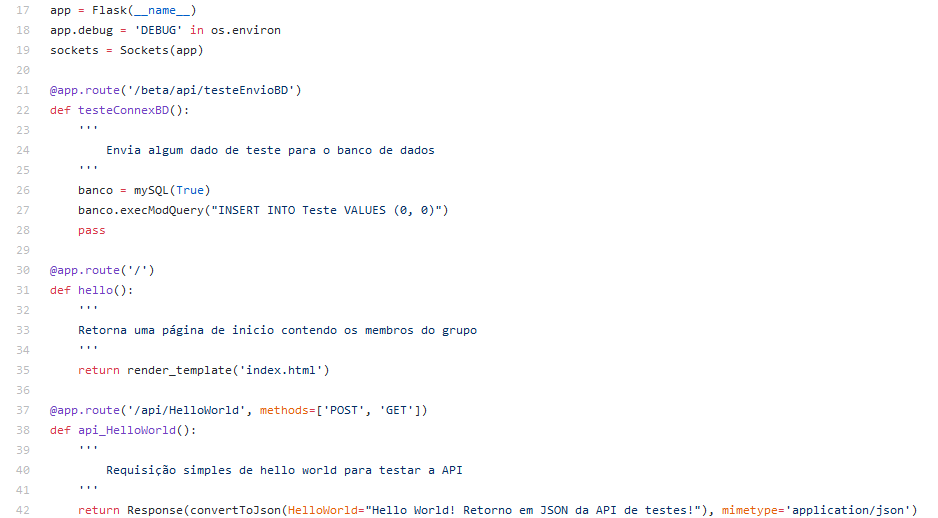
## 17. Projeto Heroku





### 17.1. Introdução

O projeto Heroku apresentado foi de um aplicativo web, desenvolvido na linguagem Python, utilizando-se do microframework flask.



Flask é um microframework para que aplicações python possam fazer uso facilmente de uma ambiente WSGI e utilize de templates JINJA para o rápido desenvolvimento de um aplicativo, oferecendo ao mesmo tempo escalabilidade, suporte a microserviços e extensões. A documentação encontra-se na bibliografia.

O aplicativo está hospedado no endereço: <http://usjtecp6an.herokuapp.com/>, nos servidores dos estados unidos.

### 17.2. Rotas

#### 17.2.1. Index (‘/’)

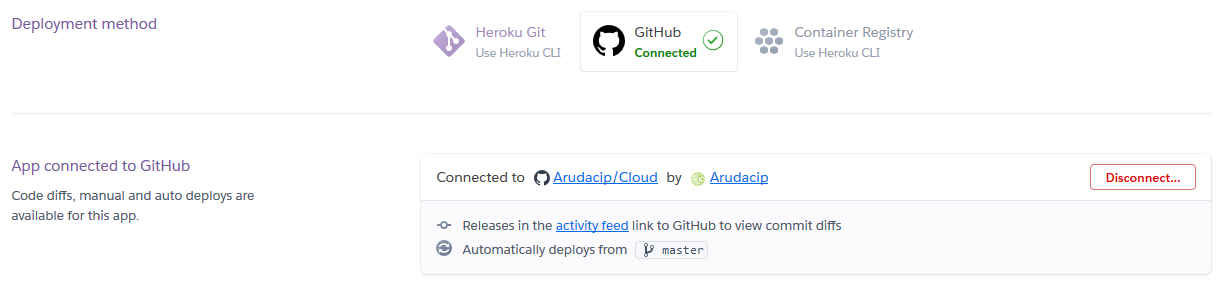
Essa rota apresenta os integrantes do grupo com seus respectivos contatos.

#### 17.2.2. Game (‘/game’)

Essa rota apresenta um simples jogo do gênero de RPG desenvolvido pelo grupo. Além do Heroku, esse jogo utiliza o SaaS remotemysql como provedor de banco de dados MySQL.

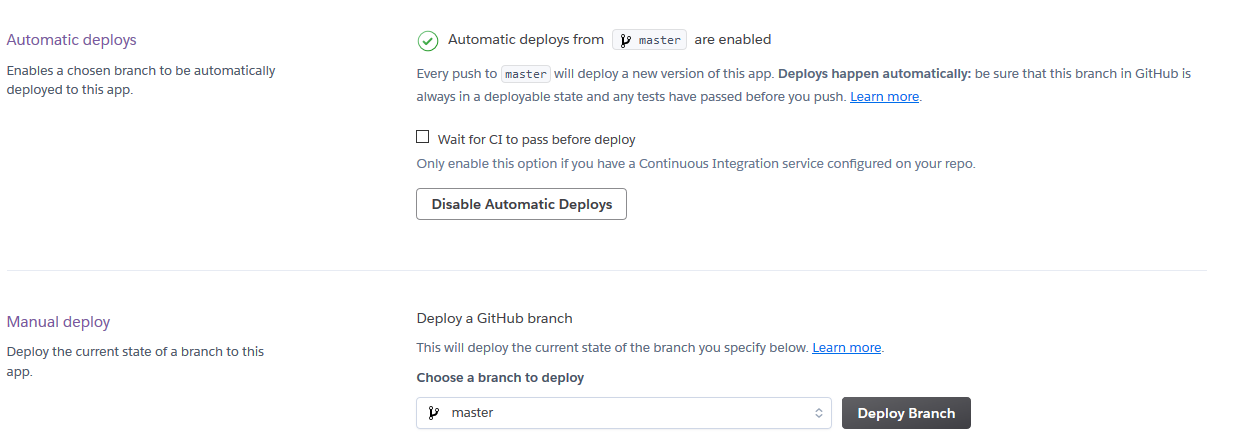
### 17.3. Repositório e código-fonte

O código fonte do projeto está hospedado no github, sendo assim, utilizamos o git como controle de versionamento. O repositório é público e pode ser acessado no link: <https://github.com/Arudacip/Cloud>.



### 17.4. Integração github e Heroku

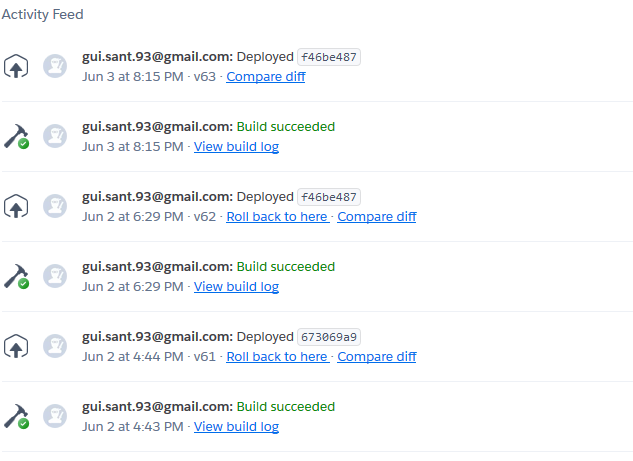
Para a realização dos deploys, utilizamos da integração do github com o Heroku e os deploys acontecem de forma automática, toda vez que houver um novo commit (Alteração do código fonte) no branch (“braço” do código fonte) master. Também é possível realizar deploys manuais.



### 17.5. Deploys

Como utilizamos o github para hospedar nosso repositório, utilizamos a integração do Heroku com o github para a realização dos deploys. Ao enviar um novo commit no branch master do repositório git, o Heroku automaticamente captura que houve uma atualização e realiza um deploy automático. Também é possível realizar deploys manuais.

Ao realizar um deploy, o Heroku busca pelos arquivos: “runtime.txt”, “Procfile” e “requirements.txt”



#### 17.5.1. Requirements.txt

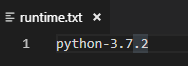
Com o controle de dependências do heroku, podemos utilizar a melhor prática possível de python para ambientes de desenvolvimento. Utilizamos um ambiente virtual (virtualenv) do python, de forma que todas as dependências do projeto existam apenas dentro desse ambiente, e sempre que houvesse uma modificação nas dependências do projeto, bastava atualizar o arquivo requirements.txt (comando: “*pip freeze > requirements.txt*”). O Heroku automaticamente buscava todas as dependências dentro do arquivo (Ele busca pelo requirements.txt para apps Python) e as instalava sem nenhuma interação com o usuário.

#### 17.5.2. Procfile

O Heroku requer um arquivo especial para indicar o tipo de dyno e a localização do nosso app. Para isso ele busca um arquivo texto simples, sem extensão, chamado “*Procfile*”. O nosso arquivo precisou de apenas uma linha com o seguinte comando: “*web: gunicorn -k flask\_sockets.worker main:app*”.

#### 17.5.3. Runtime.txt

O arquivo “runtime.txt” indica a linguagem de programação e versão da mesma, utilizada para o desenvolvimento do aplicativo. O nosso arquivo ficou com o seguinte comando: “*python-3.7.2*”, indicando que é um aplicativo escrito em python, na versão 3.7.2.



## 18. Bibliografia

* Heroku Docs: <https://devcenter.heroku.com/articles>
* Datanyze: <https://www.datanyze.com>
* Quora: https://[www.quora.com](http://www.quora.com)
* Wikipedia: <https://commons.wikimedia.org>
* Crunchbase: <https://www.crunchbase.com>
* Riptutorial: <https://riptutorial.com>
* Flask Docs: <http://flask.pocoo.org/>
* WSGI Docs: <https://modwsgi.readthedocs.io/en/develop/>
* Jinja2 Docs: <http://jinja.pocoo.org/>
* Git: [https://www.git-scm.com/](https://git-scm.com/)
* Github: <https://www.github.com>
* Repositório do projeto: [https://www.github.com/Arudacip/Cloud](https://git-scm.com/)