# Estilo funcional e Análise de dados

## Contexto

Nesta atividade, você será introduzido a tarefa de Análise de Dados combinadas a técnicas/estilo de Programação Funcional. O cenário proposto utiliza dados de usuários extraídos do GitHub, como usuários que atribuíram estrelas, ou "*stargazers*", (Figura 1), “*watchers*” (aqueles que acompanham modificações) e “*releases*” (usuários que criaram marcos importantes da aplicação). Os registros fornecidos pela API do próprio Github incluem diversas informações sobre esses usuários que podem ser usadas por desenvolvedores e pesquisadores para conhecerem melhor sua comunidade.

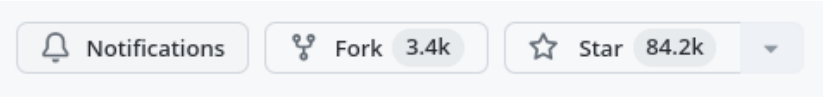


Figura 1 - Funcionalidades sociais no Github

RepoInsights (https://repo-insights.hsborges.dev) é uma ferramenta desenvolvida com objetivo de permitir que desenvolvedores analisem seus projetos a partir da coleta direta de dados do próprio GitHub (Figura 2). Para isso é necessário que você faça o login na aplicação para que a mesma possa fazer as requisições à API do GitHub.


Figura 2 - *RepoInsights*

Atualmente a plataforma oferece uma visão simplificada que, para uma visão aprofundada da comunidade, ainda não provê informações detalhadas para uma análise estatística adequada. Nesta atividade, você deverá extrair informações detalhadas sobre algumas métricas apresentadas na plataforma a partir dos dados fornecidas por ela própria. Essa abordagem permitirá a aplicação prática de conceitos da programação funcional, ao mesmo tempo, em que promove o desenvolvimento de habilidades analíticas aplicadas ao contexto de desenvolvimento de software.

# Atividade Proposta

Nesta atividade você deverá criar um *script* para extração dos valores **mínimo, máximo, média, mediana e desvio padrão** das métricas de ***followers***, ***following*** e "***account age***". O resultado deverá ser escrito em formato CSV cujas linhas representam as métricas indicadas e as colunas os valores obtidos para cada uma das medidas.

A seguir são apresentadas informações sobre cada uma das métricas:

- **Followers**: Valor numérico indicando a quantidade de seguidores do usuário (campo *followers_count*).
- **Following**: Valor numérico indicando a quantidade de usuários que o indivíduo segue (campo *following_count*).
- **Account age**: Tempo que usuário possui a conta na plataforma sendo calculado a partir da data de criação da conta (campo *created_at*).

**[BÔNUS 50%]** Você poderá criar também um segundo *script* para fornecer uma **listagem das principais localizações fornecidas pelo usuário**, em quantidade de ocorrências. Este script deverá escrever, em formato CSV, a localização e a quantidade de ocorrências da mesma. Nesta atividade não será necessário fazer normalizações das localizações, podendo usar diretamente aquelas que foram informadas. Contudo, recomenda-se transformar todas as localizações fornecidas em minúsculo somente.

⚠️ **Visão integradora aplicada:** É obrigatório a criação de testes unitárias para as funções criadas. Você poderá possuir uma função principal a ser chamada pelo sistema para interagir com usuários e esta não precisará possuir testes.

## Coleta e descrição dos Dados

Os dados podem ser coletados diretamente na aplicação, acessando o endereço da ferramenta (https://repo-insights.hsborges.dev) e acessando o repositório de sua preferência.

⚠️ Recomenda-se o uso de projetos não tão grandes para esta atividade, principalmente pelo fato que a coleta dos dados pode demorar caso se escolham repositórios com 20k+ estrelas.

Após a coleta de todos os dados, você poderá fazer o download dos mesmos a partir da opção “Download” que está ao final da tabela de dados (Figura 3).


Figura 3 - Funcionalidade de *download* de dados

Os dados fornecidos estão disponíveis em arquivos no formato JSON, contendo amostras que seguem a estrutura do exemplo abaixo:

```javascript
[
      {
          "id": "MDQ6VXNlcjQzNzU0NzM=",
          "__typename": "User",
          "login": "rolandpeelen",
                                                "avatar_url":
      "https://avatars.githubuserconten.....",
          "created_at": "2013-05-08T10:12:09.000Z",
          "email": "",
          "is_campus_expert": false,
          "is_github_star": false,
          "is_hireable": true,
          "location": "Amsterdam",
          "name": "Roland Peelen",
          "followers_count": 30,
          "following_count": 73,
          "events": [
            {
              "type": "starred",
              "date": "2020-06-18T18:06:13.000Z"
            }
          ]
      }

]
```

# Entregáveis

Você deverá submeter um *script* que seja capaz de realizar a leitura de um arquivo de dados no formato JSON e fornecer como saída os dados extraídos, conforme especificado anteriormente. O *script* deve gerar uma saída, no próprio terminal, no formato CSV (separado por vírgulas) contendo as seguintes colunas para cada métrica:

- min (valor mínimo encontrado para a métrica)
- max (valor máximo encontrado para a métrica)
- avg (valor médio da métrica)
- median (valor da mediana da métrica)
- std (valor do desvio padrão)

Para o *script* bônus, a saída esperada apresenta duas colunas: (i) nome da localização e (ii) quantidade de ocorrências. A saída deverá apresentar as localizações de maior ocorrência no início, ou seja, a saída deve estar ordenada de forma decrescente em número de ocorrências.

# Comentários Finais

A atividade requer a aplicação do estilo de programação funcional para manipular e transformar os dados, extraindo as informações mencionadas de forma eficiente e precisa. Em especial, você deverá buscar seguir os princípios de pureza e imutabilidade dos dados, além de usar funções de alta ordem típicas da programação funcional.

Nesta atividade você poderá usar a linguagem de programação de sua preferência, desde que a mesma ofereça construções funcionais mínimas.

- ☐ Busque usar de construções funcionais, como definição e uso de **funções puras**
- ☐ Manter a **imutabilidade** dos dados permitirá que os dados passem pelo pipeline necessário

- ☐ As seguintes **funções** são **altamente recomendadas** para trabalho com conjunto de dados: reduce, map, filter
- ☐ Busque criar **funções independentes passíveis de serem compostas** para atingir o objetivo definido
- ☐ Use a linguagem de programação que possua maior **conforto e conhecimento** para usar