UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
UNIDADE ACADÊMICA DE SISTEMAS E COMPUTAÇÃO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

VINICIUS ALENCAR AGOSTINI

DADOSJUSBR.ONLINE:

UMA FERRAMENTA PARA DEMOCRATIZAÇÃO DAS INFORMAÇÕES DE PAGAMENTOS PARA MAGISTRADOS

CAMPINA GRANDE - PB

2019

VINICIUS ALENCAR AGOSTINI

DADOSJUSBR.ONLINE:
UMA FERRAMENTA PARA DEMOCRATIZAÇÃO DAS INFORMAÇÕES DE PAGAMENTOS PARA MAGISTRADOS

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientadora: Professora Dra. Raquel Vigolvino Lopes.

CAMPINA GRANDE - PB

2019

A275f Agostini, Vinicius Alencar.
Dadosjusbr.online : uma ferramenta para democratização das informações de pagamentos para magistrados. / Vinivius Alencar Agostini. – 2019.

13 f.

Orientadora: Profa. Dra. Raquel Vigolvino Lopes.
Trabalho de Conclusão de Curso - Artigo (Curso de Bacharelado em Ciência da Computação) - Universidade Federal de Campina Grande; Centro de Engenharia Elétrica e Informática.

1. Transparência pública – pagamentos. 2. Dados públicos. 3. Remuneração de magistrados. 4. Libertação contínua de dados I. Lopes, Raquel Vigolvino. II. Título.

CDU:004(045)

Elaboração da Ficha Catalográfica:

Johnny Rodrigues Barbosa
Bibliotecário-Documentalista
CRB-15/626

VINICIUS ALENCAR AGOSTINI

DADOSJUSBR.ONLINE:
UMA FERRAMENTA PARA DEMOCRATIZAÇÃO DAS INFORMAÇÕES DE PAGAMENTOS PARA MAGISTRADOS

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Professora Dra. Raquel Vigolvino Lopes
Orientadora – UASC/CEEI/UFCG

Professor Dr. Carlos Eduardo Santos Pires
Examinador – UASC/CEEI/UFCG

Professor Dr. Tiago Lima Massoni
Examinador – UASC/CEEI/UFCG

Trabalho aprovado em: 02 de julho de 2019.

CAMPINA GRANDE - PB

dadosjusbr.online

Uma ferramenta para democratização das informações de pagamentos para magistrados

Vinicius Alencar Agostini
Orientadores: Daniel Fireman, Raquel Lopes
Departamento de Sistemas e computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
viniciusaagostini@gmail.com

RESUMO

Todos os meses o Conselho Nacional de Justiça publica em seu portal dados de pagamentos para mais de 25 mil magistrados. Esses dados são disponibilizados de forma fragmentada em 93 planilhas, em formato proprietário XLS e não estruturado, sem um identificador único para os magistrados ou outra forma de identificar as informações contidas nas diversas abas de cada planilha. Dadas estas características, uma análise global sobre esses dados torna-se inviável.

O exercício pleno da cidadania envolve entender e fiscalizar gastos públicos. Assim, esses dados são uma fonte de informação muito valiosa para cidadãos interessados em participar ativamente na gestão financeira do poder judiciário ou fiscalizá-lo.

O dadosjusbr é uma ferramenta que provê acesso às informações de pagamentos para magistrados de forma simples e rápida. Disponibilizamos o dadosjusbr.online, um portal onde os dados são publicados em um formato amplamente compatível com ferramentas de análise e processamento de dados. Tudo isso rodando em uma infraestrutura que não gera custos financeiros. Nenhuma informação de remuneração é perdida no processo, elas são apenas reorganizadas em um único arquivo de dados.

Assim, o dadosjusbr causa impactos positivos para nossa sociedade, incentivando a participação do cidadão na administração pública e fortalecendo as iniciativas de transparência e fiscalização do Conselho Nacional de Justiça, um dos maiores órgãos públicos do país.

REPOSITÓRIOS

https://github.com/dadosjusbr/parser

https://github.com/dadosjusbr/remuneracao-magistrados

PALAVRAS-CHAVE

Transparência; Dados Públicos; Remuneração dos Magistrados; Libertação contínua de dados.

1 Introdução

Em cumprimento à lei de acesso à informação[1] o Conselho Nacional de Justiça (CNJ) determinou que todos os tribunais brasileiros devem publicar os dados referentes à remuneração de seus magistrados através de uma planilha que segue o modelo[2] disponibilizado pelo próprio órgão. Esse modelo de planilha é um arquivo XLS, um formato proprietário do Microsoft Excel[3], que é composto por cinco abas com os respectivos nomes: contracheque, subsídio, indenizações, vantagens eventuais e dados cadastrais.

A aba contracheque discrimina o salário líquido, bruto, o total de benefícios e os descontos com impostos e retenção por teto salarial de cada Magistrado. Por sua vez a aba de subsídio contém as informações sobre o seus salários e acréscimos ao mesmo. Já a aba de indenizações conta com os valores recebidos através de diversos auxílios, a exemplo do auxílio moradia, auxílio natalidade. A aba vantagens eventuais contém os valores recebidos através de abonos e gratificações. Por fim, a aba dados cadastrais contém informações como a matrícula, o tribunal de origem e o cargo de cada magistrado.

Com isso, na página do CNJ são publicados mensalmente dados de pagamentos de mais de 25 mil magistrados, que especificam não só quanto cada um recebe, mas também os benefícios e subsídios que compõe o valor de seu contracheque. Tais dados possuem um valor imenso para

[1] Lei Do acesso à informação http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2011/Lei/L12527.htm

[2] Modelo de planilhas de remuneração de magistrados disponibilizado pelo Conselho Nacional de Justiça http://www.cnj.jus.br/files/conteudo/arquivo/2017/11/becada0200f03cb5a129ce57513f8ff3.xls

[3] https://products.office.com/pt-br/excel

quem deseja fiscalizar ou realizar análises sobre as remunerações recebidas pelos funcionários do poder judiciário e tem um papel muito importante para o controle social das políticas públicas[4].

No entanto, essas publicações são realizadas mensalmente através de um conjunto de 93 planilhas, uma para cada tribunal ou conselho de justiça, em um formato que não é amigável para ser analisado em ferramentas de análise e processamento de dados. Além disso, dado a quantidade de arquivos em que a informação está distribuída, a tarefa de realizar uma análise global sobre essas informações torna-se inviável.

Por exemplo, uma pessoa interessada em saber qual o gasto total em remunerações para magistrados ao longo de um ano, terá que fazer o download de 1116 planilhas, manipular uma a uma a fim de conseguir a soma local e por fim realizar a soma global de todos os resultados encontrados. Para um humano, esse tipo de tarefa repetitiva é extremamente custosa em termos de esforço e tempo, sujeita a erros e pode ser resolvida fazendo o uso de um sistema computacional, desde que os dados estejam em um formato adequado.

O dadosjusbr é uma ferramenta que tem o objetivo de prover acesso às informações completas de pagamentos para magistrados de forma simples e rápida. Para tal, disponibilizamos o dadosjusbr.online, um portal onde os dados são publicados em um formato amplamente compatível com ferramentas de análise e processamento de dados e estão organizados em uma página por mês de referência. Tudo isso rodando em uma infraestrutura que não gera custos financeiros. Nenhuma informação de remuneração é perdida no processo. As informações são apenas reorganizadas em um único arquivo de dados.

Com isso, a mesma pessoa interessada na soma anual do valor total gasto com remuneração com magistrados precisa baixar apenas 12 arquivos, um para cada mês, importá-los em uma ferramenta de análise de dados, como por exemplo o R[5], e usar funções prontas capazes de realizar essa soma.

Isso possibilita que cidadãos interessados e profissionais de áreas como: jornalismo, direito, ciências sociais, ciência de dados e afins possam construir análises e narrativas embasadas em dados concretos com o mínimo esforço, podendo assim focar nas informações ali contidas e em como elas podem contribuir para a sociedade.

Segundo Ramos Júnior (RAMOS JUNIOR, 2009, p.147) "o princípio da eficiência exige transparência na administração pública, para se ter maior controle da máquina administrativa e combate à ineficiência formal, sendo possível uma maior participação do cidadão na administração pública, inclusive, criando condições para que a sociedade possa avaliar os serviços públicos e denunciar possíveis irregularidades". Ou seja, o dadosjusbr é uma contribuição importante para a transparência dos dados públicos, sendo um facilitador de iniciativas que fazem o uso desses dados para levar poder e conhecimento à população, tornando as pessoas cada vez mais engajadas com a fiscalização de órgãos públicos e com o exercício da cidadania.

Obter e transformar esses dados para que sejam publicados no portal do dadosjusbr.online não é tarefa simples. O formato de planilhas disponibilizado pelo CNJ é feito para ser interpretado apenas pelo Microsoft Excel, o que torna difícil a extração dos dados de forma automática. Além disso, os tribunais nem sempre seguem o modelo das planilhas disponibilizadas pelo CNJ, tornando mais complexa a tarefa de extrair dos dados das planilhas.

Por isso, o dadosjusbr conta com mecanismos sofisticados que conseguem extrair os dados das planilhas tolerando alguns erros de inserção, de formatação dos dados, e estrutura das planilhas. Tudo isso com métodos que nos dão uma boa segurança sobre a qualidade e corretude dos dados coletados. Para cada mês o dadosjusbr coleta os dados e publica-os em um pacote de dados (datapackage) que traz não apenas os dados em si, em um formato amplamente usado em análise de dados (arquivo comma-separated values - CSV) e um arquivo JSON com os metadados.

2 Trabalhos Relacionados

O brasil.io[6] é um portal que captura, converte, limpa e disponibiliza dados abertos do governo em formato estruturado e não proprietário. O site conta com diversas bases de dados e uma delas é a remuneração dos magistrados, onde os dados são disponibilizados através de um único arquivo CSV que reúne todos os dados de pagamentos para magistrados publicados até determinada data.

Contudo, a ferramenta de extração, coleta e publicação do brasil.io possui algumas limitações. A primeira diz respeito à lógica de coleta dos dados das planilhas, que consegue apenas coletar dados da primeira aba das planilhas, que contém informações gerais de quanto cada magistrado recebeu, mas não possui os dados úteis para decompor essa renda, esses, estão contidos em outras abas e são uma parte importante do conteúdo, mas não tão trivial de se extrair.

[4] http://www.polis.org.br/uploads/1058/1058.pdf
[5] https://www.r-project.org/
[6] https://brasil.io/home

Além disso, todos os dados são publicados em apenas um arquivo CSV, isso faz com que a cada publicação, seja necessário realizar toda a coleta e extração dos dados para todas as planilhas publicadas até então e conforme o tempo vai passando, o número de arquivos vai crescendo, bem como o tamanho do arquivo gerado. Isso quer dizer que ao longo do tempo essas operações vão demorar cada vez mais e vão demandar mais recursos computacionais, tornando essa abordagem não-escalável.

Além disso, a coleta e publicação dos dados não são feitos de maneira integrada, ou seja, ambos são realizados em etapas distintas e dependem de uma interação humana para que os dados coletados sejam publicados no portal. Isso aumenta consideravelmente o esforço necessário e consequentemente o atraso que o portal tem em relação à disponibilização dos dados mais atuais publicados pelo Conselho Nacional de Justiça. Por exemplo, no dia 14 de junho de 2019 os dados mais atuais de remuneração de magistrados disponíveis no brasil.io eram de 28 de junho de 2018, quase um ano de diferença em relação aos dados disponíveis na página do CNJ. Por fim, manter toda essa infraestrutura gera custos financeiros, o que pode ser uma ameaça ao brasil.io, que é um projeto sem fins lucrativos e que não possui uma instituição para financiá-lo.

Existem também outras iniciativas como o Justa[7], que é um site voltado para a fiscalização do poder judiciário, que provê diversos dados e análises focando em 4 eixos: orçamentário, legislativo, suspensão de segurança e composição de raça e de gênero. Embora o Justa conte com diversas análises envolvendo valores monetários utilizados pelo poder judiciário, ainda não há informações que façam referência à remuneração de Magistrados.

3 Arquitetura e Projeto da Solução

3.1 Visão geral do projeto

O dadosjusbr é uma plataforma que visa facilitar o acesso e o uso de dados públicos por iniciativas interessadas em prover um maior controle social sobre as contas públicas. Seu foco são os dados de remuneração dos magistrados, que são publicados na página do Conselho Nacional de Justiça[8] ao longo de 93 planilhas mensais no formato XLS e que contam com dados de mais de 25 mil magistrados.

Para tornar essas informações amigáveis a ferramentas de análise e processamento de dados, essas 93 planilhas passam por um processo de extração dos dados, que posteriormente são limpos e estruturados. Após essa etapa os mesmos são empacotados em um datapackage e publicados no portal do dadosjusbr[9].

Além de disponibilizar os dados, o dadosjusbr provê também uma interface simples e robusta para que os administradores do sistema consigam publicar os dados de cada mês de referência. Essa interface é uma ferramenta de linha de comando (CLI) em que o usuário pode configurar uma série de parâmetros via variáveis de ambiente e após isso, basta executá-la para que todo fluxo de execução que culmina na coleta, extração e publicação dos dados seja iniciado (Figura 1).

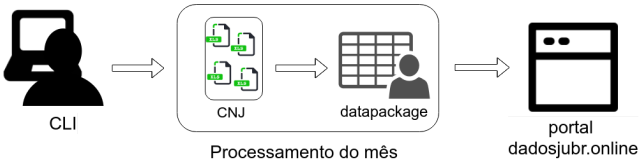

Figura 1: **Fluxo de execução disparado pelo usuário via CLI.**

Uma propriedade importante do projeto que foi levada em consideração desde sua concepção é que ele não deve gerar custos financeiros. Custos financeiros de um projeto desse tipo envolve os serviços de armazenamento e envio de email utilizados e toda a infraestrutura para manter o portal no ar. Para tal, focamos em uma arquitetura que permite que os serviços possam rodar por tempo indeterminado em contas gratuitas.

3.2 O portal dadosjusbr.online

Para que os usuários possam baixar os dados providos pelo dadosjusbr, existe um portal, o *dadosjusbr.online*. O portal consiste de uma página inicial (Figura 2) que exibe informações gerais do site, bem como os links para o repositório do github do projeto e para as redes sociais da iniciativa. Ainda na página inicial, existe um menu lateral através do qual o usuário pode acessar os dados dos meses ali disponibilizados. Ao clicar em uma das opções desse menu, o usuário é redirecionado para a página do mês selecionado.

[7] http://justa.org.br/

[8] Portal da transparência do Conselho Nacional de Justiça http://www.cnj.jus.br/transparencia/remuneracao-dos-magistrados

[9] http://dadosjubr.online

Figura 2: **Página inicial do *dadosjusbr.online*.**

A página do mês (Figura 3) contém o mesmo menu lateral visto na página inicial para que os usuários possam navegar entre os meses. Além disso, essa página conta com um gráfico de barras que exibe o total gasto naquele mês com: subsídios (salário base dos magistrados), auxílios (conjunto de benefícios que os magistrados recebem) e total de rendimentos (valor bruto que todos os magistrados receberam). Logo abaixo, existe uma tabela que exibe estatísticas sobre esses campos como média, mediana e desvio padrão. Por fim, são exibidos, o total de magistrados cujas informações foram disponibilizadas, o botão para download do arquivo contendo o datapackage e o botão para o download das planilhas originais do mês reunidas em um arquivo ZIP.

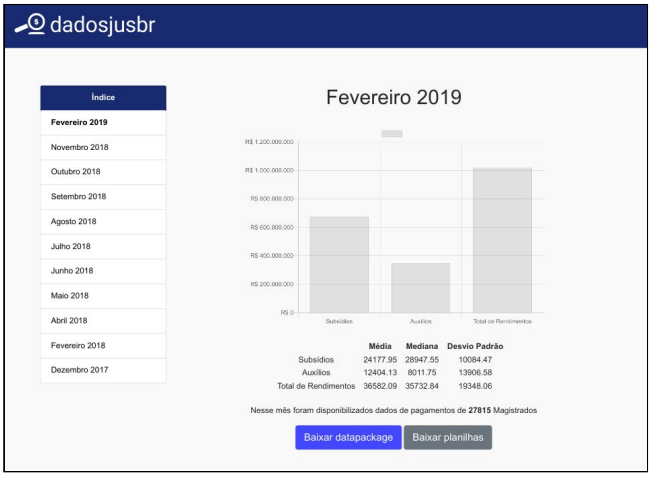

Figura 3: **Página com os dados do mês de fevereiro de 2019 no *dadosjusbr.online*.**

3.3 Os dados publicados

A fim de potencializar ao máximo a usabilidade dos dados publicados, o dadosjusbr adotou o tabular *datapackage* como formato para a disponibilização dos dados mensais processados. *Tabular datapackage* é um formato simples usado para publicar e compartilhar dados no estilo tabular. Seu foco está na simplicidade e facilidade de uso, especialmente na Web. Além disso, o mesmo é focado em dados que podem ser apresentados em uma estrutura tabular e em facilitar a produção (e o consumo) de pacotes de dados tabulares a partir de planilhas e bancos de dados relacionais[10]. Ele é composto essencialmente por dois arquivos: um *arquivo comma-separated values* (CSV) que contém os dados tabulares, e um arquivo chamado datapackage.json, que é um descritor dos dados contidos no arquivo CSV. Este arquivo JSON de descritores, descreve propriedades como: a ordem, o nome, o tipo e a descrição de cada coluna dos dados, bem como os recursos e nome do *datapackage*. Dessa forma, provemos os dados publicados no Conselho Nacional de Justiça em um formato não-proprietário, estruturado, simples e amplamente compatível com ferramentas de análise e processamento de dados.

3.4 Processamento de um mês de remunerações

Uma aplicação CLI (*command line interface*) foi desenvolvida para permitir que os administradores do sistema possam iniciar esse fluxo de processamento de um mês de remunerações, que termina com a publicação dos dados no portal. Esta aplicação CLI permite que uma série de configurações sejam realizadas. Por exemplo, o usuário pode informar qual o mês e ano a serem processados, bem como a URL origem dos dados (página do CNJ ou local), além de configurações referentes aos serviços utilizados, como as credenciais para o serviço de armazenamento, para o banco de dados e para o serviço de envio de emails automáticos. Esta aplicação CLI que dá início ao processo de coleta e estruturação dos dados poderia ser facilmente executada periodicamente de forma automática através de programação específica em ferramentas como o crontab.

Após essas configurações, o usuário aciona a aplicação CLI, inicializando todo o fluxo de execução, que começa acionando um módulo chamado *Crawler*. O Crawler recebe uma página HTML, encontra todos os endereços para planilhas dentro desta página, faz o download das mesmas e mantém o conteúdo em memória, junto com os respectivos nomes dos arquivos.

Com os conteúdos das planilhas em mãos, o próximo passo é extrair os dados nelas contidos. Para tal, as planilhas são enviadas uma a uma via HTTP para um micro serviço chamado *Parser*. O *Parser* possui uma API simples, que recebe como entrada uma planilha XLS e retorna um arquivo CSV contendo as informações de todas as abas dessa planilha limpas e estruturadas. A extração dos dados dessas planilhas é um processo complexo e por isso existe uma série de passos, conversões e heurísticas utilizadas para tal, que serão detalhados em uma seção focada exclusivamente no Parser.

[10] https://frictionlessdata.io/specs/tabular-data-package/

O resultado do *Parser* é uma coleção de arquivos CSV, um para cada planilha processada, que serão concatenados a fim de formar um CSV com todos os dados de remuneração para magistrados do mês. Esse arquivo, juntamente com o descriptor.json são empacotados e formam o datapackage que será publicado. São empacotadas também as planilhas originais baixadas do CNJ, pois as mesmas são disponibilizadas no portal do dadosjusbr em um arquivo ZIP.

Nessa etapa, com todos os dados do mês carregados, são coletadas as estatísticas que serão exibidas na página do mês em questão.

Por fim, o datapackage e o arquivo ZIP contendo as planilhas originais são enviados para nosso serviço de armazenamento, que devolve um link de acesso para cada um dos arquivos, esses, juntamente com as estatísticas são salvos no banco de dados e isso já faz com que o mês processado fique disponível no portal do dadosjusbr.

Sempre que todo o fluxo de execução for bem sucedido, ou seja, caso não ocorra nenhum erro, os administradores do sistema são notificados via email com a informação de que novos dados foram disponibilizados no portal. Caso algum erro ocorra, os administradores também são notificados, mas com uma mensagem com o máximo de informações possíveis sobre o erro, para que dessa forma possam agir, seja de forma a corrigir alguma falha na aplicação ou solicitar ao CNJ que corrija alguma falha crítica na planilha que gerou o erro.

3.5 Principais módulos do dadosjusbr

3.5.1 Crawler

O *Crawler* é o módulo responsável por varrer a página do Conselho Nacional de Justiça, coletar os links paras as planilhas ali publicadas e fazer o download das mesmas. Essas planilhas são mantidas em memória, em uma estrutura de dados composta por uma lista de tuplas contendo o nome do arquivo e um array de bytes que corresponde ao arquivo da planilha.

3.5.2 Processor

O *Processor* é o orquestrador de todo o fluxo de processamento de um mês. Ele é responsável por chamar o *Crawler* para fazer o download das planilhas e posteriormente chamar o *Parser* várias vezes para extrair os dados das planilhas obtidas, uma a uma. Em seguida o *Processor* deve chamar os outros módulos responsáveis pelas etapas necessárias até o fim do fluxo principal, como o módulo que coleta as estatísticas, o módulo de acesso ao banco de dados e o de envio de emails automáticos.

Há também um sistema de tratamento de erros, que adiciona informações úteis sobre o tempo de execução de cada etapa e qual delas gerou o erro. Isso é muito importante do ponto de vista dos administradores pois quanto mais precisa for a mensagem de erro enviada a eles, mais fácil será para entender o motivo do erro e decidir quais medidas devem ser tomadas.

3.5.3 CLI

Esse módulo é responsável por coletar as configurações do usuário e passá-las para o *Processor*, que é quem de fato irá executar todo o fluxo de coleta e publicação dos dados.

3.6 Os serviços utilizados

Nesta seção apresentamos as decisões tomadas sobre uso de serviços e plataformas para manter o dadosjusbr de forma gratuita.

3.6.1 PCloud

O PCloud[11] é um serviço de armazenamento de arquivos em nuvem que é utilizado pelo dadosjusbr para armazenar os *datapackages* e arquivos zip com as planilhas originais de cada mês. Ele foi escolhido por conta da sua API que permite o envio de arquivos de forma simples e rápida e também por conta do seu plano gratuito, que oferece um amplo espaço de armazenamento.

3.6.2 Heroku

O Heroku[12] é um serviço de nuvem que oferece "Platform as a Service", ou seja, ele permite que você hospede suas aplicações em um ambiente facilmente escalável e com suporte a diversas tecnologias. O dadosjusbr faz uso do seu plano gratuito, que possui algumas limitações em poder de processamento, memória e disponibilidade do serviço. Por exemplo, após um período de inatividade, o serviço é desativado. Em consequência, na próxima vez que o serviço for acessado haverá um período de espera para o início do serviço. No entanto, a forma como a aplicação que serve o *dadosjusbr.online* e o *Parser* são implementados eles conseguem ser mantidos nessa plataforma sem a necessidade de um plano pago. Isso é possível porque as chamadas para o *Parser* são implementadas de forma serem tolerantes com esse tempo de inicialização do serviço e o servidor do *dadosjusbr.online*, que necessita de mais recursos computacionais, é implementado em uma linguagem altamente performática em termos de processamento e gerenciamento de memória, sendo assim suficientes os recursos computacionais providos pelo Heroku.

3.6.3 Sendgrid

O Sendgrid[13] é uma plataforma de envios de email, que possui uma API que permite que emails automáticos sejam enviados sem a necessidade de toda complexidade em volta da utilização de um servidor de SMTP próprio. Nesse sentido, o dadosjusbr utiliza esse serviço, também com o plano gratuito, que permite o envio de 100 emails diários,

[11] https://www.pcloud.com/pt/
[12] https://www.heroku.com/
[13] https://sendgrid.com/

número mais que suficiente para notificações sobre publicações de dados ou erros ocorridos durante o fluxo de execução da aplicação.

3.6.4 MLab

MLab[14] é um serviço que que hospeda na nuvem um banco de dados MongoDB completamente gerenciável. Ele oferece gratuitamente 500 MB para armazenamento de dados. O que é um tamanho suficiente para manter a base de dados do dadosjusbr. Atualmente, os resultados de cada mês ocupam cerca de 1,6 KB de memória no banco de dados, assim, com esses 500 MB disponíveis, podemos armazenar os resultados de mais de 300 mil meses.

3.7 Parser

O *Parser* é um micro serviço que faz parte do dadosjusbr como um todo, mas sua base de código vive em um repositório separado dada a sua complexidade, o volume de código e a tecnologia utilizada. Este é o módulo mais complexo e importante do dadosjusbr e a principal contribuição deste Trabalho de Conclusão de Curso.

3.7.1 API

O Parser é um micro serviço expõe uma API que espera receber uma planilha XLS, via array de bytes ou uma URL para a mesma, e responde com os dados dessa planilha limpos e estruturados em um *arquivo* CSV. O Parser vai retornar uma mensagem de erro caso algum problema ocorra no processamento da planilha.

3.7.2 Funcionamento Geral

Extrair dados de arquivos no formato XLS é uma tarefa naturalmente complexa, tendo em vista que esse é um formato proprietário e feito para ser interpretado apenas pelo Microsoft Excel. No entanto, essa é uma demanda tão frequente e útil que existem hoje diversas bibliotecas capazes de realizar esse trabalho. Dentre elas, a mais popular, usada por mais de 22 mil projetos no github, com o maior número de funcionalidades e mais robusta é a js-xlsx[15], que é escrita na linguagem Javascript[16] e tem uma API simples e bem documentada.

Usando essa biblioteca conseguimos converter cada uma das abas da planilha recebida para uma matriz de strings, onde cada linha e coluna dessa matriz reflete cada linha e coluna da respectiva aba. A seguir apresentamos detalhes do processamento dessas matrizes para geração do datapackage. A complexidade envolvida nesta tarefa vem em geral do fato dos tribunais não seguirem o modelo de planilha sugerido pelo CNJ. Para lidar com as diferentes versões foram necessárias heurísticas definidas com base na avaliação das planilhas já publicadas até a presente data.

O primeiro problema surgiu quando percebemos que alguns tribunais invertem a ordem das abas dentro do XLS e modificam os seus nomes (das abas) em relação ao modelo do CNJ. Com isso, para identificar que matriz corresponde a que aba, criamos uma heurística que usa partes dos nomes de cada aba que são comuns ao formato original e às modificações realizadas pelos tribunais.

Com essas matrizes em mãos, conseguimos manipular facilmente os valores contidos na mesma, mas o seu conteúdo ainda é apenas um reflexo de tudo que estava contido na planilha, e isso inclui muitas informações não relevantes para nosso propósito, como por exemplo instruções de preenchimento, além de células vazias.

O segundo problema enfrentado é como encontrar os dados dentro dessa matriz. Para lidar com este problema, criamos uma outra heurística, capaz de detectar em que linha da matriz encontra-se o cabeçalho da aba (Figura 4). Sabendo onde está o cabeçalho da aba, identificamos onde estão os dados que queremos coletar.

CPF	Nome	Subsídio(1)					
		Abono de permanência (R$)	Outra (R$)	Detalhe	Outra (R$)	Detalhe	Total de Direitos Pessoais
xxx.xxx.xxx-xx	ACACIO JULIO KEZEN CALDEIRA	0	0				R$ -
xxx.xxx.xxx-xx	ADALBERTO ELLERY BARREIRA NETO	0	0				R$ -

Figura 4: **Cabeçalho da aba de subsídios do modelo de planilhas de remuneração de magistrados disponibilizado pelo CNJ.**

Uma vez que sabemos onde estão os dados, para coletá-los, basta iterar sobre as linhas da matriz, salvando cada linha em uma estrutura de dados que armazena as informações de cada magistrado. No entanto, mais uma vez encontramos problemas pois existem alguns tribunais que adicionam novas colunas nas abas, ou modificam os nomes das colunas. Isso faz com que novamente, heurísticas devam ser estabelecidas para que os dados coletados em cada uma das planilhas possuam exatamente os mesmo campos, isso é importante para a formação do *datapackage*. Esta heurística consegue detectar quais são as colunas que pertencem ao modelo padrão do CNJ, as que não pertencem são agregadas em uma coluna chamada "outras despesas", que contém a soma dos valores coletados nessas colunas.

Usando a heurística mencionada acima, podemos coletar os dados de cada magistrado iterando sobre a matriz linha a linha. Durante essa coleta, todo dado extraído passa por uma função que identifica qual o tipo desse dado e aplica limpeza e normalização sobre o mesmo. Por exemplo, se passarmos como entrada para essa função um valor do tipo string "R$ 20,50" a mesma irá retornar 20.5 do tipo numérico. Isso é importante porque diversas planilhas possuem problemas de formatação, principalmente em campos de datas e valores monetários.

Por fim, como cada aba possui seus próprios campos e uma estrutura única, a extração dos dados de cada uma delas é realizada de forma independente. Com isso, como queremos que nosso CSV possua uma linha por magistrado, temos que unir todos os dados de cada magistrado que estão distribuídos ao longo das abas. Isso é o que chamamos de estruturar os dados. No entanto isso também não é um problema simples, pois existem casos

[14] https://www.mlab.com/
[15] https://github.com/SheetJS/js-xlsx
[16] https://developer.mozilla.org/pt-BR/docs/Web/JavaScript

onde os dados de um magistrado existem em uma aba mas não existem em outra. Como não há um identificador único nas abas para cada magistrado, usamos o seu nome como chave para unir os dados de todas as abas. Não há garantia de que não existem homônimos entre magistrados em uma mesma planilha e isso pode impactar os dados dada a forma como coletamos os mesmos atualmente. Contudo, realizamos experimentos usando cerca de 1200 planilhas distintas comparando os nomes dos magistrados contidos nas mesmas e em nenhuma delas existiam duas pessoas com o mesmo nome. Com base nisso, decidimos manter essa estratégia de usar o nome do magistrado como identificador para unir as informações de cada magistrado em abas diferentes da planilha.

3.7.3 Qualidade dos dados

Com uma lógica tão complexa para extrair uma quantidade tão grande de dados, é difícil saber saber se os dados estão de fato sendo coletados corretamente para todos os casos, se estão bem formatados e estruturados. Como uma forma de aumentar a confiabilidade do sistema nesse sentido, foram implementados diversos testes automáticos, tanto de unidade quanto de regressão, que dão ao Parser uma cobertura de 97% sobre as linhas de código[17]. Grande parte desses testes são implementados fazendo uso de planilhas reais, que servem como entrada para as funcionalidades a serem testadas, com isso, sabemos que de fato o sistema funciona para as planilhas que serão recebidas.

Cada heurística implementada foi resultado de experimentos em que executamos as principais funcionalidades do *Parser* usando uma base de cerca de 1200 planilhas coletadas do portal do CNJ.

Cada experimento consiste em iterar sobre essa base de planilhas, e passar cada planilha com parâmetro para a funcionalidade em questão, analisando seu retorno, seja através de asserções automáticas ou de verificações manuais. Com isso, pudemos ter uma boa confiança de que a heurística aplicada funciona para diversos cenários distintos encontrados nos dados modificados pelos tribunais em relação ao modelo.

É importante destacar que isso não quer dizer que conseguimos coletar dados de todas as planilhas publicadas no portal do CNJ. Existem casos particulares de planilhas que possuem um modelo que foge muito do padrão do CNJ, ou planilhas que não podem ser lidas, como por exemplo as que são protegidas por senha.

A forma como optamos tratar esses casos é fazer o *Parser* lançar um erro sempre que ocorrer um caso desses. Assim, os administradores do sistema serão notificados com um feedback preciso do que ocorreu e podem tratar esse caso individualmente.

Atualmente em nosso portal temos os dados de 11 dos 15 meses que estão com todas as planilhas publicadas no portal do CNJ. Isso quer dizer que esses 4 meses que ainda não foram publicados no *dadosjusbr.online* apresentaram erro em pelo menos uma das planilhas e esses devem ser tratados pelos administradores do dadosjusbr.

3.7.4 Tratamento de Erros

Tratar e lançar erros que contenham mensagens precisas é muito importante para esse serviço, pois o mesmo lida com dados que são muito suscetíveis a erros de preenchimento e formatação, e que em certas situações não são possíveis de ser extraídos, como por exemplo, casos em que tribunais publicam planilhas protegidas por senha. Quando isso ocorre, não é possível coletar os dados dessas planilhas de forma automática e um erro deve ser lançado.

Como a publicação dos dados do mês depende da extração das informações de cada planilha, é muito importante que qualquer caso de erro reúna o máximo de informações possíveis sobre o problema, pois isso será enviado para os administradores do sistema, que usarão essa mensagem para identificar e agir sobre o problema. Quanto melhor e mais precisa a mensagem de erro, mais rápida e eficiente será a solução dele.

Para prover uma mensagem de erro precisa, o *Parser* possui uma lógica de tratamento de erros, em que sempre que um problema ocorre, é criado um objeto que contém informações sobre: a etapa do fluxo de execução em que o erro ocorreu, a pilha de execução da aplicação no momento do erro, uma mensagem personalizada caso o erro seja lançado pela própria aplicação e um código de erro, que é um identificador único para cada tipo de erro.

3.8 Arquitetura do dadosjusbr

O dadosjusbr é composto por três artefatos de software. O primeiro é o remuneracao-magistrados[18], que é onde se encontra a aplicação CLI, o *Crawler* e o *Processor* todos implementados fazendo o uso da linguagem de programação Go[19]. Temos também o *Parser*, que é um micro serviço implementado em Javascript que expõe uma API HTTP e é responsável por extrair os dados das planilhas. Por fim temos uma aplicação responsável por servir os templates HTML que compõe o portal do dadosjusbr, também implementado em Go.

A seguir apresentamos um diagrama que especifica todo o fluxo de execução bem sucedido para a publicação dos dados de um mês (Figura 5), incluindo a interação com o Parser e os outros serviços externos.

[17] https://coveralls.io/builds/23837664

[18] https://github.com/dadosjusbr/remuneracao-magistrados

[19] https://golang.org/

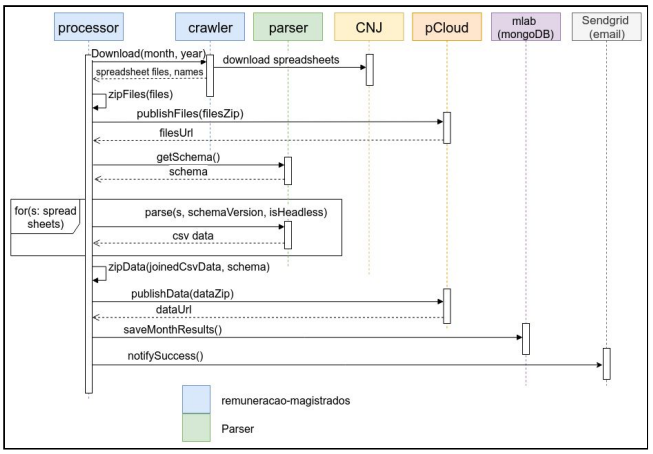

Figura 5: **Diagrama de sequência do fluxo de sucesso para a publicação dos dados de um mês.**

Esse fluxo de execução é disparado pela aplicação CLI e depende dos parâmetros recebidos pelo mesmo. Ao fim da execução, os dados daquele mês são persistidos no banco de dados, que é o mesmo banco de dados utilizado pelo servidor do dadosjusbr, e com isso, o mês já está disponível no portal de forma automática. Esse processo é apresentado na Figura 6.

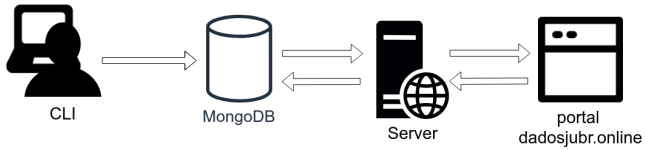

Figura 6: **Fluxo dos dados no dadosjusbr.**

3.9 Organização de repositórios

Nossa base de código é mantida em dois repositórios pertencentes a uma organização no github chamada dadosjusbr[20] (Figura 6).

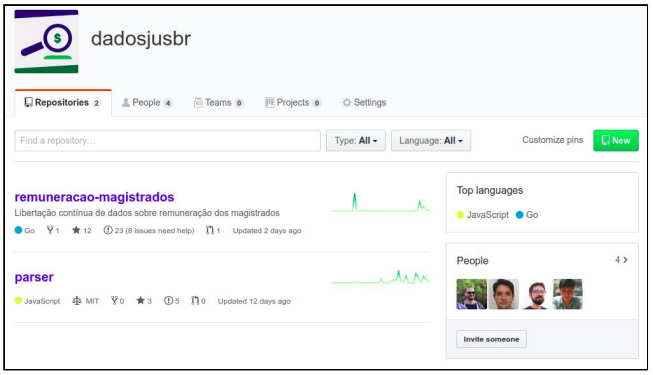

Figura 6: **Página da organização dadosjusbr no github.**

No primeiro repositório, chamado remuneracao-magistrados, são mantidos todos os módulos que compõe o fluxo de processamento de um mês de remuneração de magistrados, bem como o servidor do portal *dadosjusbr.online* e o CLI.

[20] https://github.com/dadosjusbr

O segundo, contém toda a base de código do *Parser*, bem como seus testes e as planilhas utilizadas nos mesmos.

4 Demonstração da contribuição

O dadosjusbr é uma iniciativa que surgiu em Setembro de 2018 durante um hackathon, um evento em que pessoas da área de tecnologia se reúnem para implementar uma solução de software com uma temática bem definida durante um curto período de tempo, geralmente um ou dois dias. O evento, chamado Firefest, ocorreu na cidade de Maceió e foi o pontapé inicial para o dadosjusbr.

Como resultado do evento, o dadosjusbr era capaz de renderizar uma página web que continha apenas um link para o arquivo zip com as planilhas originais deste mês, mas ainda sem dados extraídos das planilhas originais.

Foram implementados também uma versão inicial do Crawler, do módulo de envio de emails automáticos e de interação com o serviço de armazenamento.

Além disso, a lógica de extração dos dados das planilhas não era robusta, funcionava apenas para as planilhas de um mês e possuía muitas falhas, a principal delas era não reconhecer erros na coleta de dados e injetá-los no arquivo CSV retornado.

Dessa forma, esse trabalho teve por objetivo melhorar as funcionalidades implementadas durante o Firefest e evoluir a corretude e a usabilidade do sistema.

A primeira etapa da minha contribuição foi a implementação de um Parser melhor, com uma API mais simples, mais robusto, com um sistema de tratamento de erros e com propriedades que garantam uma boa confiabilidade sobre a qualidade dos dados.

Assim, podemos observar no gráfico abaixo (Figura 7) que o Parser foi, em sua totalidade, implementado durante esse trabalho.

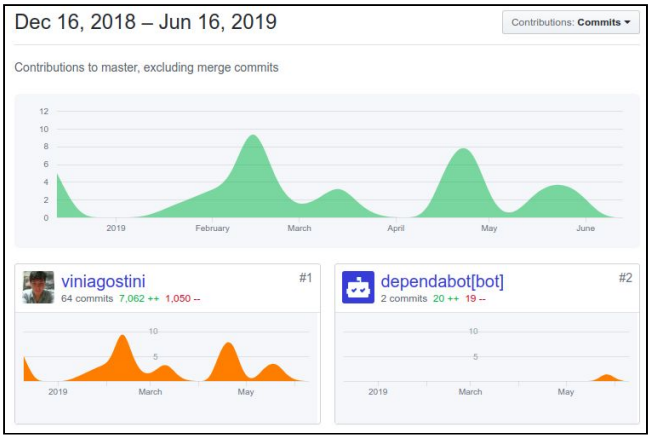

Figura 7: **Gráfico de contribuidores do Parser.**

Ainda sobre o Parser, grande parte desse trabalho foi investido em testes que hoje dão uma boa confiança sobre a qualidade dos dados e a corretude das funcionalidades e experimentos que permitiram que as diversas heurísticas

criadas para a extração dos dados pudessem ser adaptadas e generalizadas para um grande número de planilhas.

A simplificação da API do Parser permitiu que o código do Processor, que é responsável pela comunicação com o micro serviço, se tornasse muito mais simples e eficiente.

Com o Parser implementado, o foco da minha contribuição foi refatorar a base de código do Processor e Crawler deixando a interação entre os componentes mais simples e legível.

A próxima etapa foi parametrizar o fluxo de execução para a publicação dos dados de um mês, fazendo com que o mesmo possa ser executado para diversos meses.

Após isso, foi implementado o módulo de interação com o banco de dados, que permitiu que os resultados do processamento dos meses pudessem ser salvos.

Com toda a lógica de processamento dos dados do mês pronta, foi implementada a aplicação CLI para servir como interface para que os administradores possam publicar os dados no portal do dadosjusbr.online.

Por fim, com os dados no banco de dados, foi implementado o servidor que usa essas informações para renderizar os templates HTML que formam o portal do dadosjusbr.

Podemos através do gráfico abaixo (Figura 8) que o volume de contribuições para o remuneracao-magistrados durante o período desse trabalho comporta todas as funcionalidade descritas.

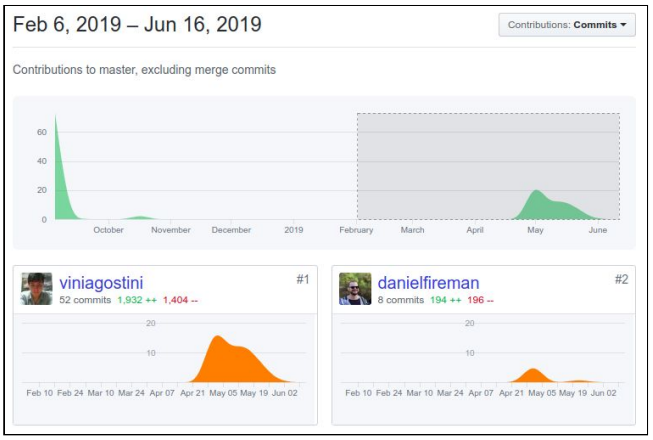

Figura 8: **Gráfico de contribuidores do remuneracao-magistrados durante o período de vigência deste trabalho.**

Com isso, pudemos ver que as contribuições realizadas durante esse trabalho tornaram o sistema não só utilizável, mas também tiveram um impacto positivo na qualidade do código, na robustez e na manutenibilidade do mesmo.

5 Experiências e lições aprendidas

A construção do dadosjusbr foi um processo de aprendizado constante. Pude exercitar ao longo desse projeto diversas competências que adquiri ao longo do curso se ciência da computação e evoluí-las.

No domínio da Engenharia de Software, os conceitos de Quality Assurance (Q&A) foram de extrema importância principalmente na etapa do Parser, onde foram necessários diversos testes e coleta de métricas para que o dadosjusbr possa entregar os dados com uma boa confiança no quesito qualidade. Além disso, as técnicas de planejamento de atividades me permitiram lidar com os riscos que são inerentes à construção de qualquer sistema de software, como atrasos em cronogramas ou tarefas mal estimadas. Por fim, todo o conhecimento e maturidade adquiridas ao longo do curso sobre processos de desenvolvimento foram cruciais para o sucesso desse projeto.

Dando uma ênfase maior no desenvolvimento de software, todo o conhecimento sobre padrões de projeto e arquitetura adquiridos em Sistemas de Informação foram muito úteis para a implementação de um software robusto, com boa performance e escalável. Além disso, os princípios boas práticas de programação permitiram a produção de uma base de código simples e intuitiva. Isso é muito importante em projetos de código aberto que visam ter muitos contribuidores.

Ainda falando sobre escalabilidade, para projetar um sistema baseado em uma arquitetura de micro serviços, foram necessários diversos conceitos de Sistemas Distribuídos, principalmente no âmbito de protocolos de comunicação entre esses diferentes serviços.

Para extrair as planilhas da página do CNJ e coletar os dados das planilhas, foram usadas diversas técnicas de Recuperação da Informação. Além disso, para coletar as estatísticas e algumas verificações manuais sobre os dados, foram usados conhecimentos sobre Ciência de Dados Descritiva.

Pensando em manter o código rodando sem custos de infraestrutura, tivemos que escolher uma linguagem rápida, leve e eficiente para implementar o dadosjusbr. É nesse sentido que surgiu a ideia de usar Go, que é uma linguagem moderna e com abstrações de alto nível, mas que possui a performance de uma linguagem de baixo nível.

Ao início do projeto eu não possuía nenhuma experiência em Go, e após algum tempo de estudo comecei a codificar na mesma e com mentoria de Daniel Fireman, através de revisões de código e longas discussões de problemas, ao fim do projeto, pude adquirir um bom nível de proficiência na linguagem, que é sem dúvidas uma ferramenta muito útil para qualquer engenheiro de software.

Por fim, esse projeto nos estado atual foi fruto de muitas discussões sobre funcionalidades, arquitetura, algoritmos e usabilidade. Então, uma competência que foi fundamental, que pude evoluir bastante e é de extrema importância para qualquer profissional é a comunicação.

5.1 Processo de Desenvolvimento

O dadosjusbr foi implementado fazendo uso de um processo incremental, em que novas funcionalidades foram sendo implementadas e integradas à base de código

principal, desde que seu código passe em verificações como a execução de testes automáticos e análise estática, seguindo o conceito de integração contínua (Continuous Integration). Ainda, toda nova funcionalidade integrada à base de código principal ela é, de forma automática, disponibilizada para os usuários em um ambiente de produção, sendo assim um processo de entrega contínua (Continuous Delivery).

Para assegurar a qualidade do código, mantemos um processo de revisão de código, onde cada nova funcionalidade é revisada por outro desenvolvedor. Isso é muito útil não só como mecanismo de validação de código, mas também como forma de compartilhar conhecimento, gerando aprendizado tanto para quem revisa o código quando para quem tem o código revisado.

5.2 Principais desafios

Um requisito bem desafiador desse projeto foi fazê-lo funcionar sem custos de infraestrutura, dado que queremos manter nosso portal com uma alta disponibilidade para os usuários, precisamos armazenar um grande volume de dados, enviar emails de forma automática, rodar operações pesadas em termos de processamento e memória, tudo isso usando planos gratuitos.

Para atender este requisito foi necessária a elaboração de uma uma arquitetura específica, um alto nível de conhecimento sobre os serviços a serem utilizados, e principalmente, muita criatividade para pensar em soluções que muitas vezes não são ótimas, mas que satisfazem esse requisito.

Contudo, o grande desafio deste trabalho foi a implementação do Parser. Isso porque dado o volume de planilhas com que o mesmo precisava lidar, era impossível prever todos os casos de erros e todos os problemas contidos nas planilhas.

Todas as funcionalidades implementadas precisaram ser validadas através de experimentos, cujos resultados foram verificados manualmente. Isso demandou muito tempo e esforço.

6 Trabalhos Futuros

Essa é apenas a primeira versão do dadosjusbr. Existem diversas melhorias nas funcionalidades atuais e diversas outras que estão por vir.

A próxima versão do dadosjusbr irá contar com um sistema de publicação de dados automático, que é capaz de detectar quando novos dados são publicados no portal do CNJ e rodar um fluxo de execução para aquele mês. Com isso, o papel dos administradores do sistema será apenas intervir em casos de erro.

Com o foco incentivar ainda mais a criação de iniciativas e aplicações focadas em nossos dados, vamos disponibilizar uma API, que permite a automação do acesso aos dados disponíveis em nosso portal.

Além disso, temos a perspectiva de começar a publicar dados de remuneração de outros órgãos do sistema de justiça.

Por fim, como a publicação das planilhas depende de cada um dos 93 tribunais, geralmente passam-se alguns meses até que todas as planilhas mensais sejam disponibilizadas no portal do CNJ. Pensando em agilizar o acesso à informação, vamos realizar processamentos parciais, publicando em nossa página meses que ainda possuem planilhas faltando, sinalizando para os usuários esse fato. Com isso, pessoas interessadas nos dados de tribunais que já disponibilizaram suas planilhas não precisam esperar até que todos tenham o feito.

REFERÊNCIAS

[1] RAMOS JÚNIOR, Hélio S. Princípio da eficiência e Governo Eletrônico no Brasil: o controle da Administração Pública pelo cidadão brasileiro. In: Revista Democracia Digital e Governo Eletrônico. Florianópolis: UFSC, 2009.