

---

# **SciELO Publishing Schema Documentation**

***Release 1.0***

**SciELO**

December 23, 2015



## CONTENTS

<b>1</b>	<b>Histórico e versões</b>	<b>3</b>
<b>2</b>	<b>Arquitetura</b>	<b>5</b>
<b>3</b>	<b>Modos de operação e replicação</b>	<b>7</b>
3.1	Replicando apenas o site . . . . .	7
3.2	Replicando serviços e preparo dos dados . . . . .	8
3.3	Replicação completa . . . . .	9
<b>4</b>	<b>CRÉDITOS E LICENÇAS</b>	<b>11</b>
<b>5</b>	<b>Indices and tables</b>	<b>13</b>



Versão 1.0 - dezembro de 2015.

Projeto *Cuidando do Meu Bairro*: abertura e geolocalização do orçamento municipal.

**Acesso:**

- Portal e implantação para o município de São Paulo em [Cuidando.vc](#).
- Fontes de dados orçamentários nos [webservices](#) do projeto [GastosAbertos.org](#).

**Códigos-fonte** (repositórios [git](#)):

- Site: [okfn-brasil/cuidando2](#)
- Módulos de dados orçamentários:
- Fontes dos engines e extratores: [okfn-brasil/gastos\\_abertos](#), [teaser.gastosabertos.org](#).
- Dados e documentação: [okfn-brasil/gastos\\_abertos\\_dados](#), [okfn-brasil/documents/gastos\\_abertos](#).
- Autenticação de usuários: [okfn-brasil/viralata](#)
- Comunicação dos usuários: [okfn-brasil/tagarela](#)
- Intermediação do eSIC: [okfn-brasil/esiclivre](#)

**Documentação:**

- Tutoriais na [Wikiversity-Português](#), Projeto Cuidando do Meu Bairro.
- Instruções para o desenvolvedor: [okfn-brasil/cuidando2-doc](#).
- Administração do projeto: [wiki.okfn.org](#).



## HISTÓRICO E VERSÕES

Este projeto teve origem no [Cuidando do Meu Bairro de 2013](#), que buscava mapear a execução do orçamento municipal de São Paulo, não só em regiões, mas também colocando um ponto no mapa para cada despesa. Por isso o presente projeto foi apelidado de **Cuidando2**: uma nova interface, novos recursos e melhoras nos dados gerados pelo software.

Como os [dados obtidos da prefeitura](#) não têm as coordenadas geográficas de cada despesa (no máximo algumas tem a região a que se destinam), o ponto de partida são os endereços nos textos das descrições das despesas, que por sua vez podem ser submetidos à [geocodificação](#) em ferramentas adequadas.

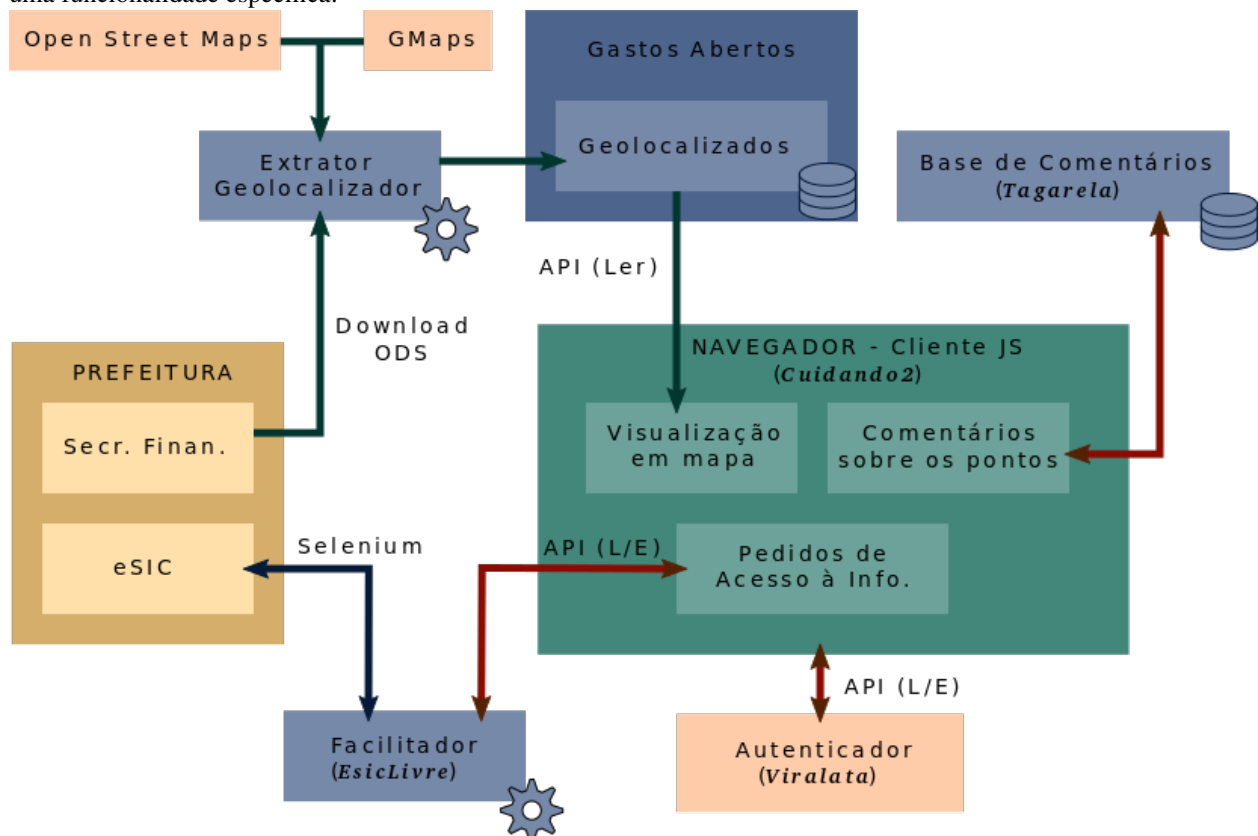
A extração dos endereços a partir do texto é obtida através de [expressões regulares](#). A geocodificação desses endereços é realizada pelas APIs [OpenStreetMap-Nomination](#) e [Google-geocoding](#). Esse processo não é perfeito, acumula erros tanto da fase de extração como da fase de geocodificação, de modo que muitas despesas não são mapeadas, ou acabam exibidas no local errado. No Cuidando2 o processo foi aperfeiçoado para reduzir a taxa de erros.





## ARQUITETURA

Buscando ampliar a chance de reuso do código desenvolvido e permitir um nível maior de interação com outros aplicativos, o projeto seguiu uma arquitetura de micro serviços. Ou seja, ao invés de ter um único código monolítico que implementasse todas as funcionalidades desejadas, elas foram distribuídas em pequenos módulos, cada um com uma funcionalidade específica:



As setas avermelhadas indicam conexões em que as escritas provavelmente necessitarão de um `token`.

A navegação HTML é de responsabilidade do `Cuidando2`, implementado atualmente em `cuidando.vc`, que através de `Ajax` faz a comunicação com cada módulo, nos respectivos `endpoints`:

Função	Responsabilidade	Endpoint em uso	Notas
Geolocalização	<i>Gastos Abertos</i>	<a href="http://site-cuidando.rhcloud.com/dados/api/v1">site-cuidando.rhcloud.com/dados/api/v1</a>	leitura das coordenadas
Dados de execução orçamentária	<i>Gastos Abertos</i>	<a href="http://demo.gastosabertos.org">demo.gastosabertos.org</a>	consulta à base de dados
Autenticação dos usuários	<i>Vira-Lata</i>	<a href="http://viralata-cuidando.rhcloud.com">viralata-cuidando.rhcloud.com</a>	token de acesso, leitura/escrita
Comentários dos usuários	<i>Tagarela</i>	<a href="http://tagarela-cuidando.rhcloud.com">tagarela-cuidando.rhcloud.com</a>	leitura/escrita dos textos
Interface com eSIC	<i>EsicLivre</i>	<a href="http://cuidando.vc/esiclivre">cuidando.vc/esiclivre</a>	realização de pedidos de informação, leitura/escrita

Cada um desses *endpoints* apresenta também uma documentação mais detalhada da API quando acessado diretamente do navegador, exemplificando o uso de cada uma das operações **REST** disponíveis no microserviço.

## MODOS DE OPERAÇÃO E REPLICAÇÃO

Conforme a finalidade, o projeto pode ser copiado e adaptado, parcial ou integralmente. O caso típico é a replicação para atuar com outros municípios, mudando apenas o escopo de dados no Gastos Abertos e adaptando-se o *EsicLivre* para as peculiaridades do eSIC do município.

Uma outra forma comum de replicação é a integral, quando se deseja instalar e testar localmente (o programador no seu computador) para entender melhor o funcionamento do software como um todo, ou para realizar (através de *fork*) adaptações maiores.

### 3.1 Replicando apenas o site

Na ausência de dados da cidade, operaria em modo “somente leitura”. Util para designers e trabalhos restritos ao site.

Uma versão compacta das instruções a seguir, é oferecida também em [site.sh](#).

#### 3.1.1 Escolhas e convenções

O projeto pode rodar nas mais diversas plataformas com um mínimo de adaptações. A título de homologação, todavia, apenas um ambiente foi apreciado.

Para a obtenção de uma documentação mais clara e limpa foi adotada *convention over configuration*, ou seja, não serão comentadas todas as possibilidades de configuração, pressupomos certas convenções, e ficaria a cargo do usuário ou de uma documentação secundária qualquer desvio da convenção adotada. Padrões de referência:

- Sistema Operacional: Arch Linux 64 bits.
- Navegadores homologados: Firefox 42+, Chrome 46+.

#### 3.1.2 Conferir ambiente

Será entendido como “ambiente” do servidor do projeto,

- Linux **Ubuntu 14.04+ LTS**: pode ser mais atualizado, mas são pressupostas restrições do LTS nos exemplos de atualização. Conferir com `lsb_release -a` (resultará ex. 14.04.3).
- **Git 1.9+**: `git --version` (ex. 1.9.1).
- Server-side **Javascript engine V8, v4.6+**: `node -p process.versions.v8` (ex. 4.6.85)
- **NodeJS v5.2+**: `nodejs --version` (ex. v5.2.0)
- **npm 3.5+** (do NodeJS): `npm -v` (ex. 3.5.2) ou `npm version`, que mostra também o nodejs e o v8.

## Atualizar o ambiente

As versões mínimas indicadas de *NodeJS* e *npm* precisam ser respeitadas. Na sua instalação default o UBUNTU 14 LTS, todavia, não oferece versões atualizadas, nem mesmo após o tradicional `apt-get install`. O procedimento mais simples e correto para instalar é o seguinte:

```
curl -sL https://deb.nodesource.com/setup_5.x | sudo -E bash -  
sudo apt-get install -y nodejs
```

que já é suficiente para ambos, *NodeJS* e *npm*. Confira as versões com `npm -v` e `nodejs --version`. O V8 em geral já se encontra atualizado, mas vale conferir com `nodejs -p process.versions.v8`.

Nota: o *npm* vem com o *nodejs*, *evite* o `apt-get install npm` pois pode aparentar erro ou gerar versão muito antiga. Para garantir a atualização do *npm* use ele mesmo,

```
sudo npm install npm -g
```

### 3.1.3 Clonar e instalar o projeto

```
git clone https://github.com/okfn-brasil/cuidando2.git  
cd cuidando2
```

Em seguida verifique as configurações de `configs/config.js`. O caso típico é mudar apenas o domínio para o local de teste,

```
nano configs/config.js  
...  
const _domain = 'http://localhost:'  
...
```

Caso não vá apenas simular o site, ou apenas reusar os microserviços de São Paulo, há que se configurar mais parâmetros. Feito isso, roda a instalação:

```
npm i
```

O ultimo comando vai um barra de progresso... Sinal que está indo tudo bem.

### 3.1.4 Rodar

Para rodar o site:

```
npm run dev
```

Depois acesse `localhost:5001` em um navegador. Se quiser que o código atualize automaticamente conforme editar os arquivos, acesse `localhost:5001/webpack-dev-server/`.

(disponível também em [install-site.md](#)).

## 3.2 Replicando serviços e preparo dos dados

Para reproduzir os microserviços, deve-se reproduzir também as bases de dados.

As instruções para replicação do *software* estão em [install-services.md](#). A preparação dos dados pode ser realizada de três modos:

- *sandbox*: base de dados mínima com dados de teste.

- *referência*: dados da base em operação, em um site Cuidando já implantado.
- *novo*: construir uma base de dados nova (por exemplo para um novo município). A metodologia e dicas encontram-se descritas [nesta Wiki](#).

### 3.3 Replicação completa

Para replicar ambos, o site e os serviços, um script mais consido também é oferecido: [src/full.sh](#).



## CRÉDITOS E LICENÇAS

A presente documentação e todos os scripts [deste módulo de documentação](#) estão licenciados sob [CC-BY 4.0](#).

O Cuidando2 é um projeto realizado por várias mãos,

- Alexandre Evangelista de Souza Júnior ([@alexandre](#))
- Andrés M. R. Martano ([@andresmrm](#))
- Gisele S. Craveiro (coordenação),
- Jutta Machado Schimdt
- Solaine Lima
- ... e todos os parceiros e beta-testers, que auxiliaram com sugestões criativas e construtivas

com apoio de,

- [COLAB-USP](#)
- [OKBR](#)
- ? (financiamento)

e disperso por alguns módulos, listados a seguir com indicação das respectivas licenças e equipes:

- [Site Cuidando do Município](#): licença [AGPLv3](#).
- *equipe*: Gisele (coordenação), Andrés (software), Solaine (design).
- *Software*: [@andresmrm](#) e [@LuizArmesto](#)
- *Gastos Abertos*: licença [AGPLv3](#) em nome de [OKBR/Projeto Gastos Abertos](#).
- *\*Dados\**: [@aivuk](#), [@andresmrm](#).
- *\*Software\**: [@andresmrm](#), [@aivuk](#), [@LuizArmesto](#), [@waffle-iron](#), [@carlosandrade](#), [@lpirola](#).
- Demais [microserviços](#) e módulos de apoio:
- *eSIC Livre*: licença [AGPLv3](#).
  - *equipe*: [@alexandre](#) e [@andresmrm](#)
- *Tagarela*: licença [AGPLv3](#), software [@andresmrm](#).
- *Viralata*: licença [AGPLv3](#), software [@andresmrm](#).
- *Viratoken*: licença [AGPLv3](#), software [@andresmrm](#).





## INDICES AND TABLES

- `genindex`
- `modindex`
- `search`