

# Tainacan

## MANUAL DO USUÁRIO

# Sumário

## Sumário

|                            |    |
|----------------------------|----|
| Sumário.....               | 2  |
| Introdução.....            | 1  |
| Inserir Usuário.....       | 2  |
| Configurando Tainacan..... | 3  |
| Interface.....             | 4  |
| Cadastro.....              | 5  |
| Primeiro Acesso.....       | 6  |
| Criar Ontologias.....      | 8  |
| Minha Ontologia.....       | 9  |
| Criação de classe.....     | 12 |
| Edição de classe.....      | 13 |
| Propriedades.....          | 14 |
| Indivíduo.....             | 20 |
| Exportar.....              | 21 |
| Compartilhamento.....      | 25 |
| Config. Ontologia.....     | 26 |
| Eventos.....               | 28 |
| Comentários.....           | 31 |
| API.....                   | 33 |
| Agradecimentos.....        | 37 |

# Introdução

Tainacan – tecendo constelações de memória em cultura

Na era da Web de Dados, em que a informação e o conhecimento são fragmentados na rede e os recursos estão em constante evolução, o desenvolvimento de aplicativos baseados em dados abertos não pode seguir o paradigma em que as bases de dados são estáticas e criadas para um problema muito específico em um domínio restrito. Neste cenário surge o requisito de bases de dados conectados, altamente compartilháveis, que permitam a interoperabilidade e a possibilidade de lidar com o acúmulo de conhecimento (novos dados conectados) disponível na Web [Isotani:2015, Hebler:2011].

Para cumprir este requisito, permitindo a conexão dos dados de diferentes bases, o compartilhamento e a interoperabilidade na Web, além de tornar as informações mais expressivas por meio da descrição de sua semântica, tem se destacado a abordagem baseada em ontologias. Como parte da proposta da Web Semântica, a concepção e o uso das ontologias estão no centro da arquitetura projetada por [Berners:2001], e têm-se mostrado uma das tecnologias chave para atender a necessidade de aplicativos mais adequados para lidar com grandes quantidades de informações, de maneira inteligente [McGuinness:2004, Horrocks:2008].

A *Web Ontology Language* (OWL) é uma linguagem projetada para representar o conhecimento e a semântica sobre coisas, grupos de coisas e as relações entre elas, sendo o padrão da Web Semântica para o desenvolvimento de ontologias, recomendado pela W3C. Baseada em lógica computacional, o conhecimento OWL pode ser explorado por software, por exemplo, para verificar a consistência do conhecimento representado ou para raciocinar novos conhecimentos implícitos [Welty:2004].

A linguagem OWL provê um conjunto de classes, propriedades, indivíduos e axiomas que são a base para a construção de ontologias que, por sua vez, são o modelo para disponibilização de informações na Web Semântica [Noy:2001]. Para permitir a utilização destes elementos, denominados *construtores* na criação de ontologias, são utilizadas ferramentas denominadas *editores de ontologias*.

Este manual apresenta o *Tainacan Ontology*, um editor de ontologia criado pelo grupo de pesquisa do Laboratório de Políticas Públicas Participativas / MediaLab, que é parte integrante do MediaLab, da Universidade Federal de Goiás (UFG). A proposta do *Tainacan Ontology* é facilitar o processo de criação de ontologias por usuários que não possuem conhecimento técnico aprofundado.

# Inserir Usuário

Como administrador de uma instalação do repositório que se utiliza o Tainacan é possível adicionar novos usuários pela própria administração, o que pode funcionar como um pré-cadastro, em termos gerais a criação de um usuário pelo administrador não difere em muito do cadastro comum a não ser pela possibilidade de associar uma determinada função ao usuário como mostra a imagem abaixo.

**Como chegar aqui:** Administração → Usuários → Adicionar Usuário

**Adicionar Novo Usuário**

Crie um usuário novinho em folha e o adicione a este site.

**Nome de usuário (obrigatório)**

**E-mail (obrigatório)**

**Nome**

**Sobrenome**

**Site**

**Senha**

**Enviar notificação para o usuário** ☒ Enviar para o novo usuário um e-mail com informações sobre a conta.

**Função**

- Assinante ▼
- Assinante
- Colaborador
- Autor
- Editor
- Administrador

*Figura 1: Inserindo usuário*

Como é possível observar temos cinco funções diferentes em que o usuário pode ser agregado, sendo.

Assinante: Função comum para usuários cadastrados. (PARTE INTERNA DO WORDPRESS, INICIAL)

# Configurando Tainacan

Por o Tainacan Ontology ser uma bifurcação do Tainacan Repositório sua instalação padrão vem como Tainacan Repositório, para configurar o Tainacan para funcionar como Ontology é necessário modificar um parametro em um dos arquivos do Tainacan.

Para realizar essa modificação o usuário deve acessar a pasta em que o Tainacan se encontra e acessar as seguintes pastas: **config>config.php**. O conteudo do arquivo é exibido na **figura 2**.

```
/*
# modo = 0; 0 ou nulo é o funcionamento atual padrão

# modo = 1; gestão arquivística e museológica

# modo = 2; gestão de ontologias

# modo = 3; Modo de debates Tainacan

# modo = 4; integração com a rede MinC
*/

global $config;
$config['mode'] = 2;
// o nome deve ser o mesmo da pasta e tambem o nome do arquivo aonde esta
// o carregamento do modulo
$config['active_modules'] = ['tainacan-ontology']; //['archival-management']['contest']['tainacan-ontology']
//importante para a retirada de metadados criados
$config['deactive_modules'] = [];
//// ['tainacan-ontology']; //['archival-management']
// os metadados a serem inseridos como default
$config['metadata_slugs'] = [
    'creation_date_repository_property',
    'status_repository_property'];
```

*Figura 2: Arquivo de configuração*

Para colocar o Tainacan no modo de ontologias deve-se definir `$config['mode'] = 2` assim como na figura 2. Por padrão o valor da variável `$config['mode']` vem definido como 0.

# Interface

A interface do Tainacan de forma geral é simples, demonstrando facilmente os caminhos que o usuário pode percorrer.

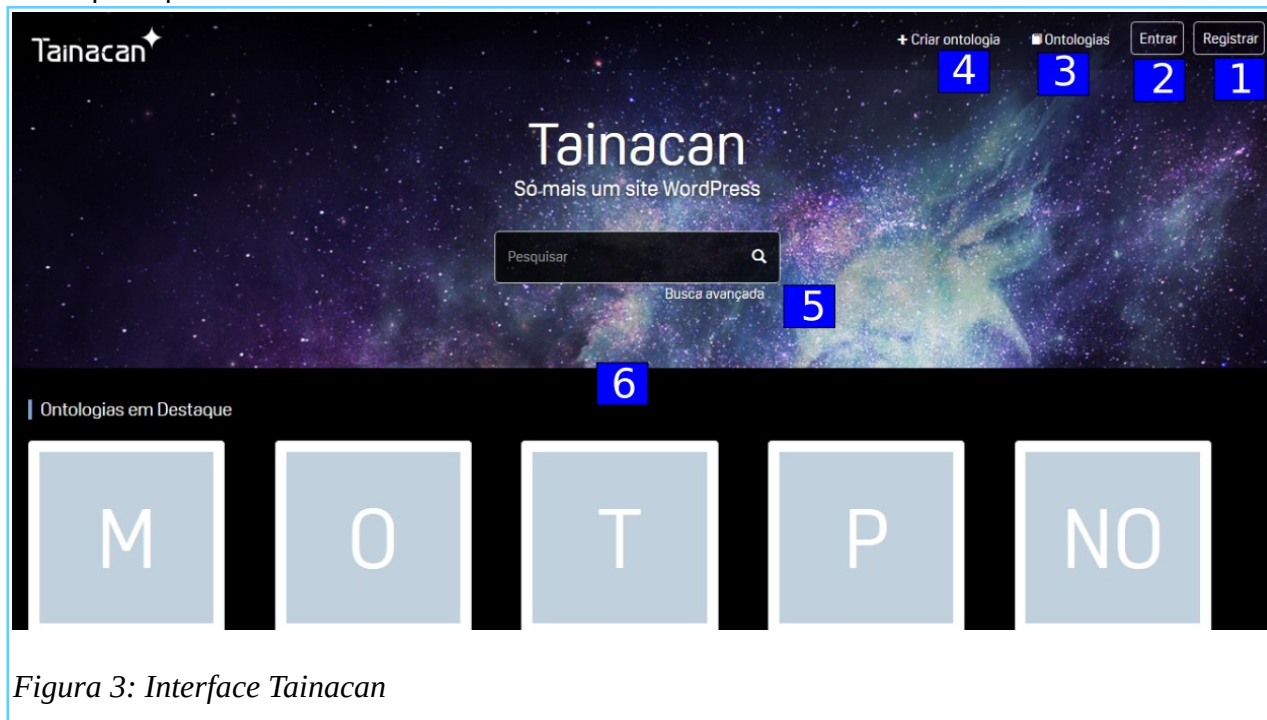


Figura 3: Interface Tainacan

- 1** Registrar: Faz referência ao cadastro de usuários no acervo, dando assim a possibilidade de criar ontologias.
- 2** Entrar: Login para usuários cadastrados.
- 3** Ontologias: Demonstra uma visão geral de todas as ontologias que foram criadas dentro do acervo.
- 4** Criar Ontologia: Possibilita a criação de ontologias pelo usuário.
- 5** Busca: Busca através das palavras-chaves digitadas.
- 6** Corpo de conteúdo: Mostra as ontologias mais populares e recentes. Em seguida os itens mais populares de todo o repositório, agrupados pelo seu tipo.

⚠ Importante: Não é possível criar uma ontologia sem ser cadastrado.

# Cadastro

Para que o usuário possa criar sua ontologia e posteriormente seus itens de ontologia, é necessário efetuar o cadastro, assim obtendo os privilégios necessários para a utilização correta da ferramenta. O cadastro pode ser feito clicando no botão “Registrar”.

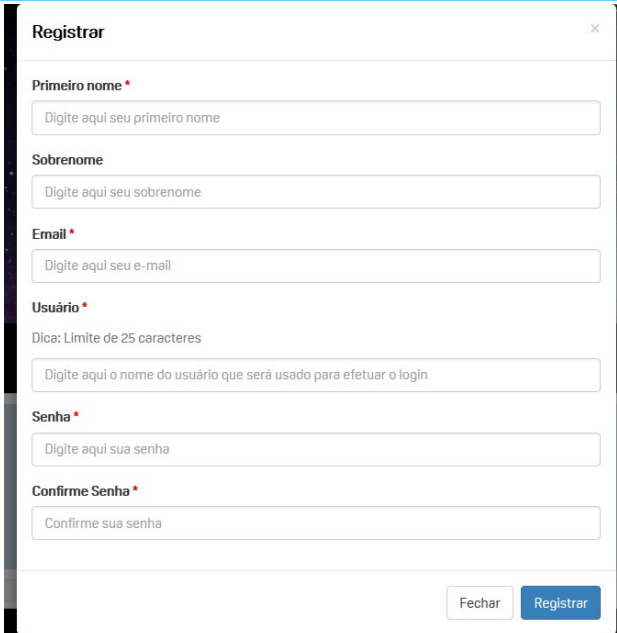
A imagem mostra uma janela modal intitulada "Registrar" com um botão de fechar no canto superior direito. O formulário contém campos para: "Primeiro nome" (com um asterisco obrigatório), "Sobrenome", "Email" (com um asterisco obrigatório), "Usuário" (com um asterisco obrigatório e uma dica: "Limite de 25 caracteres"), "Senha" (com um asterisco obrigatório) e "Confirme Senha" (com um asterisco obrigatório). Cada campo possui uma instrução de digitação: "Digite aqui seu primeiro nome", "Digite aqui seu sobrenome", "Digite aqui seu e-mail", "Digite aqui o nome do usuário que será usado para efetuar o login" e "Confirme sua senha". No canto inferior direito da janela, há dois botões: "Fechar" e "Registrar".

Figura 4: Tela de registro

1

Cadastro: Preencha todos os campos do formulário de cadastro. Os campos que possuem o indicador (\*) são campos de preenchimento obrigatório.

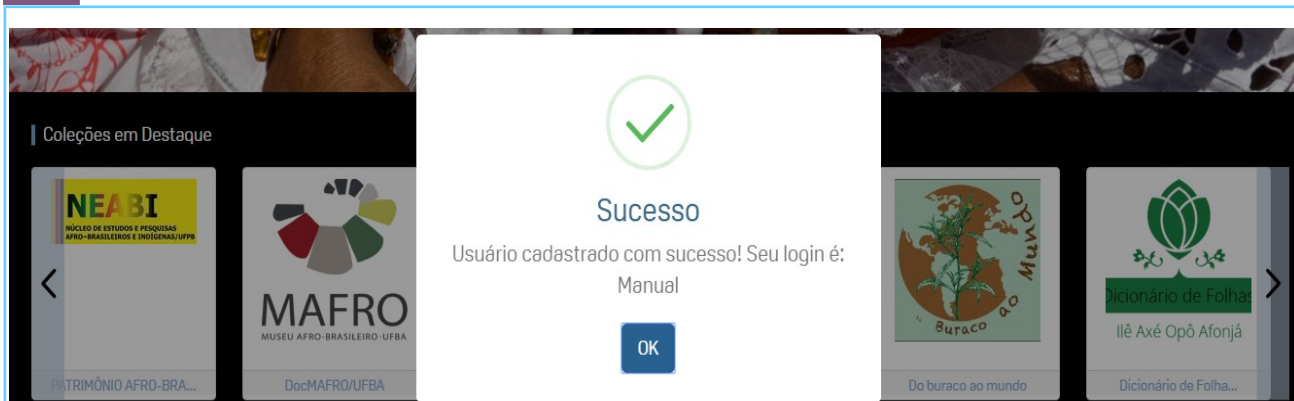


Figura 5: Registro concluído

2

Após o devido preenchimento do formulário de cadastro será recebida uma mensagem de “Sucesso”, sendo o usuário redirecionado para a página de login após clicar em “ok”.

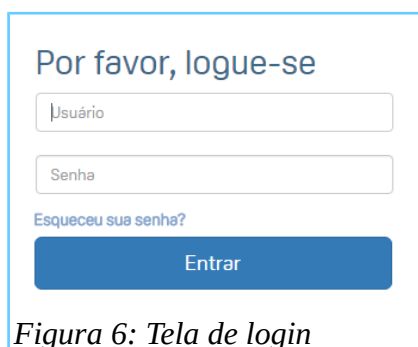
A imagem mostra a tela de login com o título "Por favor, logue-se". Há dois campos de entrada: "Usuário" e "Senha". Abaixo dos campos, há um link "Esqueceu sua senha?". Um botão azul "Entrar" está na base do formulário.

Figura 6: Tela de login



# Primeiro Acesso

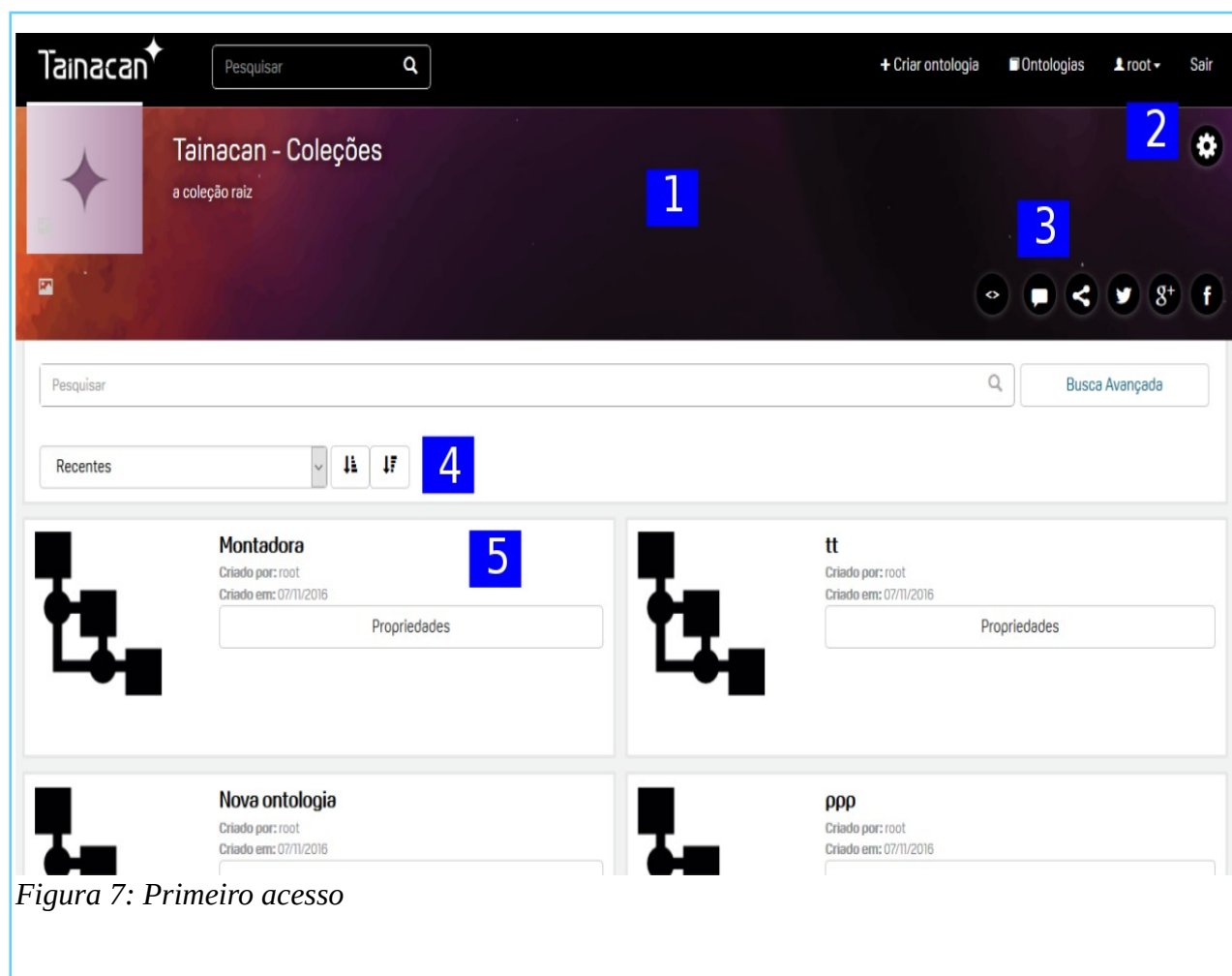


Figura 7: Primeiro acesso

Após o login, em seu primeiro acesso o usuário será direcionado à tela representada pela **figura 7** acima, que lista, organiza e dá acesso a todas as ontologias do Tainacan.

**1** Área Superior: Essa área agrega o nome da ontologia, a imagem representativa e uma imagem de capa, ambas podem ser alteradas pelo usuário.

**2** Menu Administrativo: O menu da roda dentada aparece em todas as ontologias e apresenta alguns itens para os usuários que visitam a ontologia e os donos dessa ontologia.

**3** Menu Social: Esse menu apresenta a possibilidade de compartilhar a ontologia assim como o ícone de balão para fazer comentários e o ícone a sua direita que possibilita exportar a ontologia em RDF, GRAFO.

De modo geral o item acima faz com que o usuário tenha a opção de compartilhar suas ontologias através das mídias sociais além da integração que é proporcionada pelas e exportação de arquivos que contemplam os atuais padrões web relacionados a informação e semântica.




# Primeiro Acesso

**4** Disposição de conteúdo: a partir dos filtros apresentados no ponto quatro é possível alterar a forma de apresentação de itens de uma ontologia, sendo esses apresentados, por diferentes ordens.

A ordenação de uma ontologia pode seguir diferentes padrões sendo alguns deles já predefinidos, como no caso da ordenação por itens mais recentes, que mostra os itens de acordo com a data de criação, podendo-se utilizar também metadados para a ordenação dos itens, um exemplo seria utilizar a votação dos itens como ordenação o que traria os itens mais votados como sendo os primeiros a aparecer para o usuário. Dessa forma a ordenação pode ser configurada consoante as necessidades do dono da ontologia.

**5** As ontologias ou itens de ontologias são listadas conforme definido nas configurações da ontologia

 Importante: A ontologia principal pode ser acessada a partir do menu “coleções”.

# Criar Ontologias

Uma vez obtendo acesso à ferramenta torna-se possível a criação de ontologia a partir do menu, clicando no item “+Criar Ontologia”.

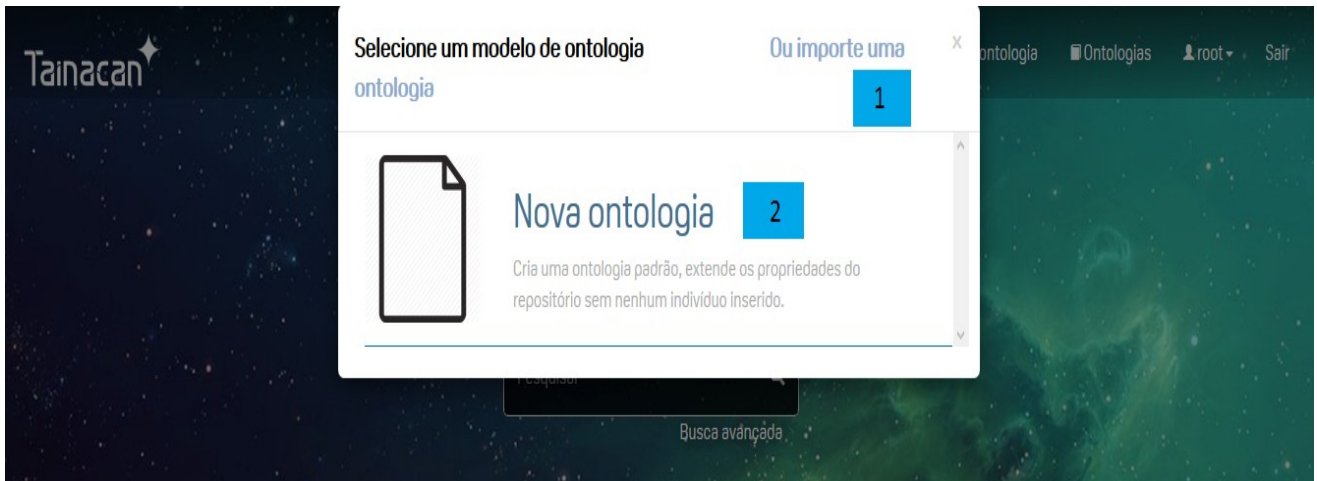


Figura 8: Criação de uma nova ontologia

**1** É possível importar ontologias que foram feitas em outros repositórios que utilizam o Tainacan, para a importação o arquivo compactado deve conter o formato do Tainacan.

**2** Ontologia vazia: cria uma nova ontologia sem itens, onde o usuário estará responsável por administrar e adicionar indivíduos.

A próxima etapa consiste em nomear sua ontologia partindo para configurações específicas que irão ser apresentadas com os itens adicionados, algumas dessas configurações são propriedades, filtros, classes, imagem e imagem de capa da ontologia, como veremos a seguir!!



Figura 9: Nomeando uma ontologia

# Minha Ontologia

## 1 – Interface geral

Após efetuar a criação de uma ontologia é levado para a página da ontologia, representada abaixo, inicialmente a página se mostra simples e sem muito conteúdo o que será mudado a medida que forem adicionados novos indivíduos a ontologia.

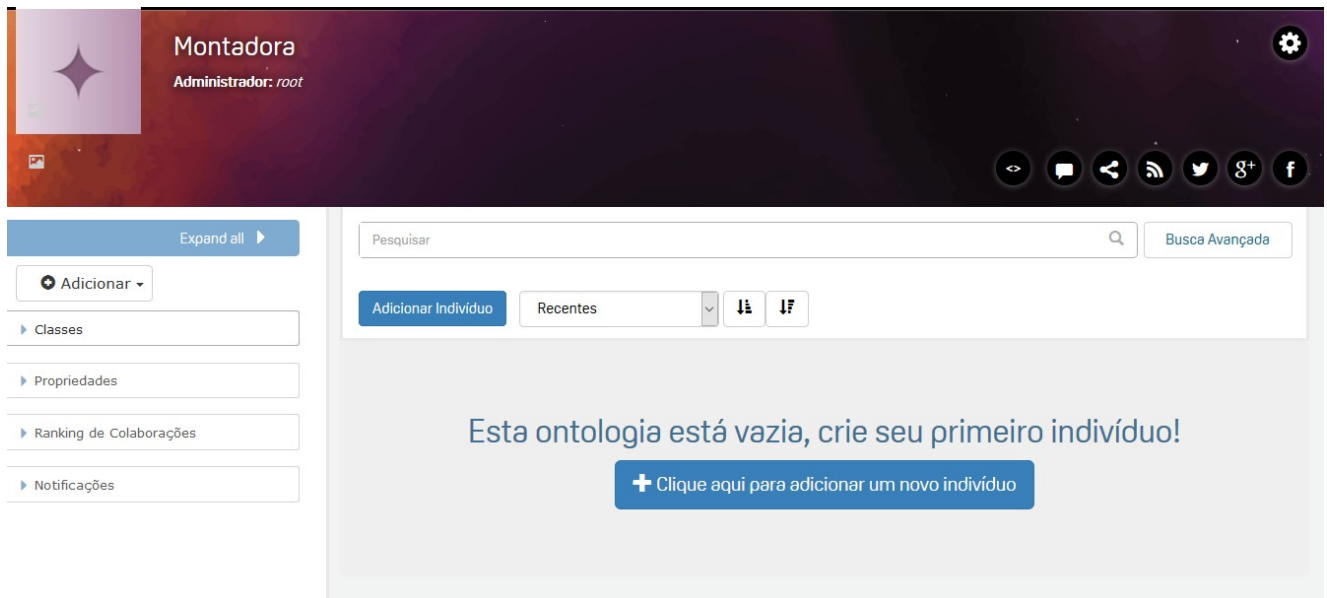
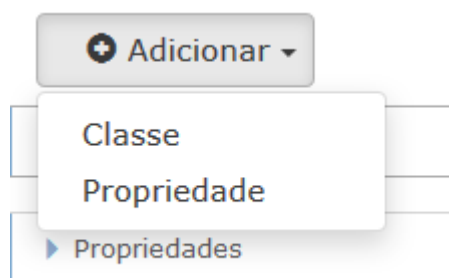


Figura 10: Visão geral de uma ontologia recém-criada

Na página de sua ontologia (Figura 9), existem alguns itens específicos que demonstram diferentes opções, que auxiliam na mudança de configurações e compreensão do ambiente pelos visitantes da ontologia.



**Adicionar (Figura 11):** o botão adicionar permite ao usuário adicionar classes e propriedades a sua ontologia. O usuário pode adicionar os seguintes tipos de propriedades:

- **Atributo:** são elementos que definem a estrutura de uma classe
- **Relação:** define a relação existente entre classes.

Figura 11: Botão Adicionar

# Minha Ontologia

## Adicionar Indivíduo

[Voltar](#)

Nome do indivíduo

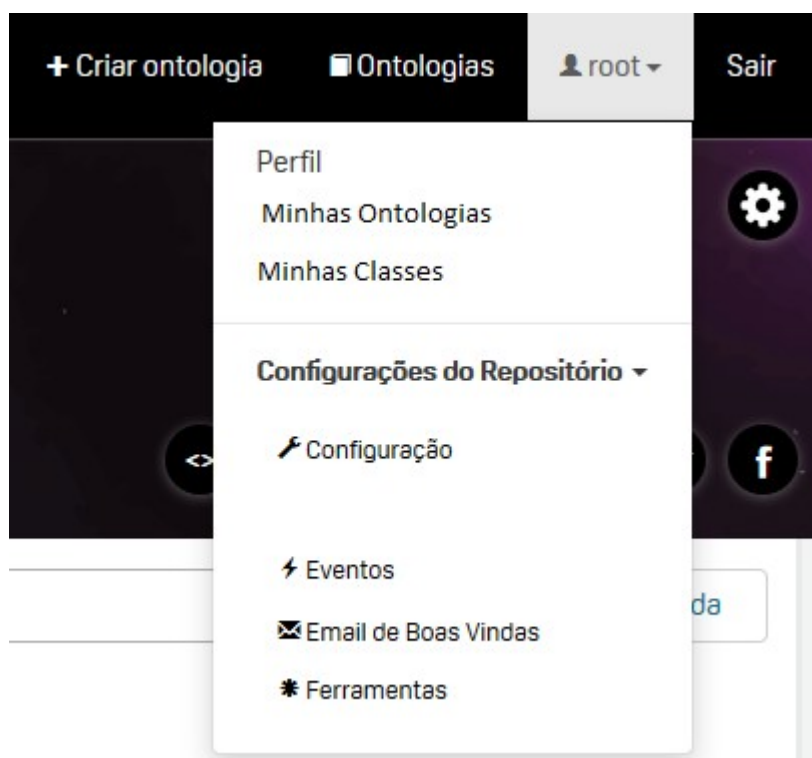
Miniatura do Item (Opcional)

Nenhum arquivo selecionado.

Descrição do Item

Figura 12: Adicionar novo indivíduo

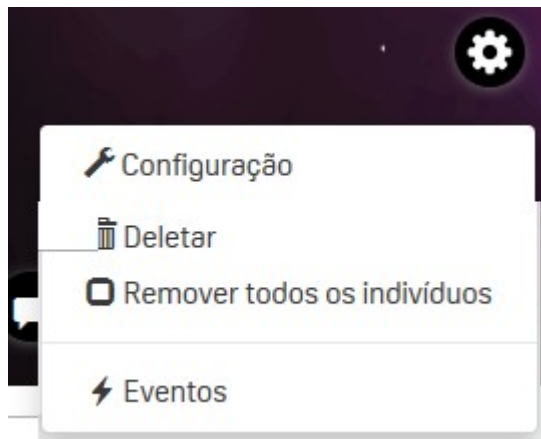
**Adicionar indivíduo (Figura 12):** ao clicar no botão adicionar indivíduo o usuário pode adicionar novos elementos a ontologia. Como pode ser visto na figura acima devemos dar um nome ao nosso novo indivíduo e temos a opção de adicionar uma miniatura a para exibição além de podermos dar uma descrição a esse novo item que está sendo adicionado.



No **menu de usuário (Figura 13)** é possível ver quais são as ontologias já cadastradas (**Minhas Ontologias**) e também é possível acessar as classes cadastradas (**Minhas Classes**). Configurações de Repositório serão tratados posteriormente.

Figura 13: Menu de Usuário

# Minha Ontologia



Acessando o menu representado por uma engrenagem (**Figura 14**) temos acesso as opções de configurações de ontologia, deletar a ontologia atual, remover todos os indivíduos e aos eventos gerados.

*Figura 14: Configurações de ontologia*

## 2 – Criação de classes e edição de classes

### 2.1 – Criação de classes

Para criar uma classe basta acessar o **botão adicionar (Figura 11)** e em seguida clicar no botão classe. Na **figura 15** vemos as opções que surgem para a criação de uma classe.

*Figura 15: Criar classe*

O campo nome da classe é de preenchimento obrigatório, e como o próprio nome diz deve-se colocar o nome que classe deve ter. A opção de classe pai é usada para fazer com a classe que está sendo criada herde de alguma classe criada anteriormente, ou seja, tenha as mesmas propriedades e relacionamentos que a classe pai.

# Criação de classe

Após a criação da classe a mesma pode ser editada, para isso basta pressionar o botão direito do mouse sobre a classe que se deseja editar e selecionar a opção **editar**, e todas as opções definidas anteriormente poderão ser modificadas. Ao clicar em **Opções avançadas** a tela da **figura 16** será exibida.

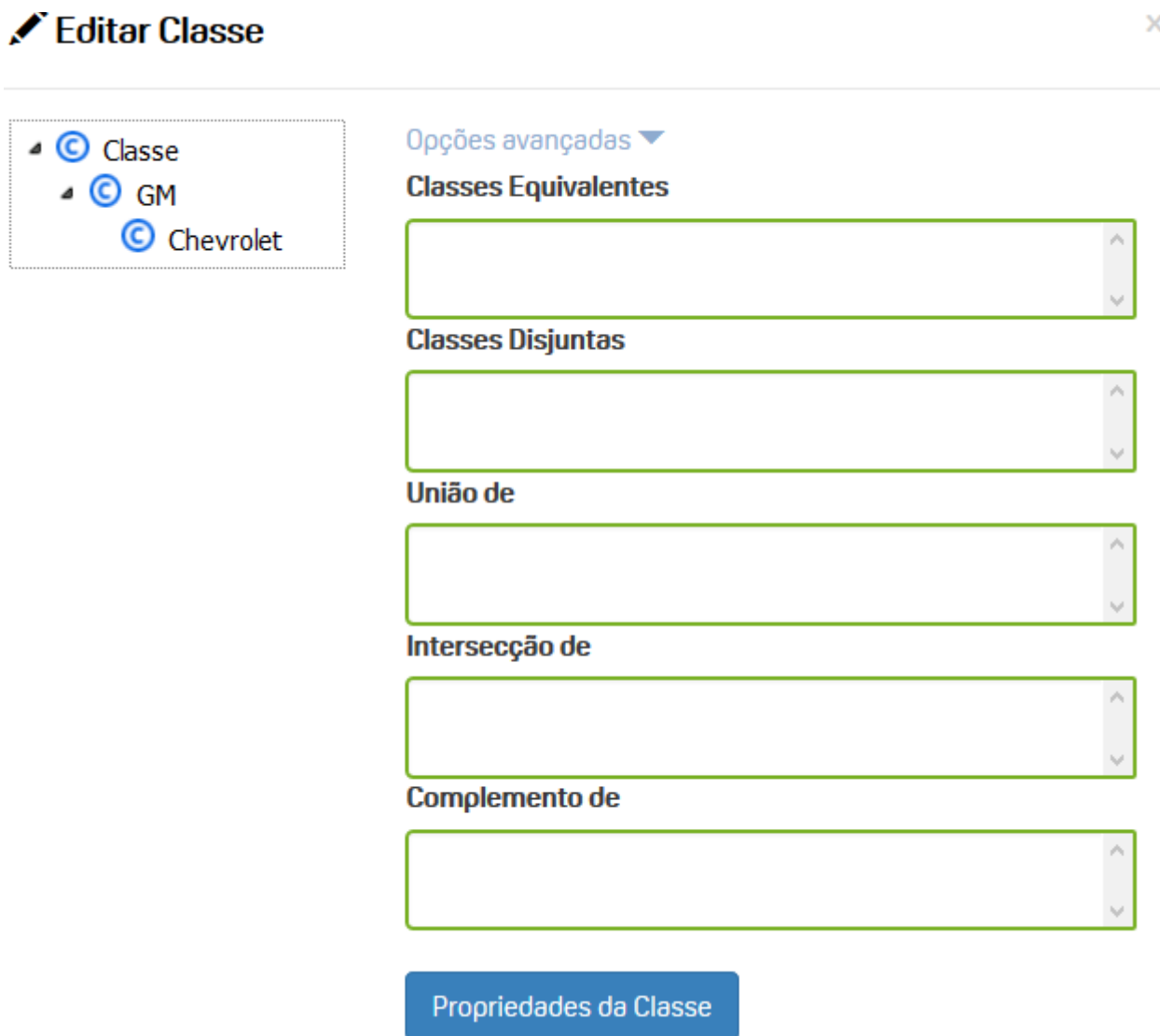


Figura 16: Opções avançadas de classe

As opções apresentadas na figura 16 podem ser vistas como:

- **Classes equivalentes** possuem a mesma instância. Igualdade pode ser usada para criar classes que são sinônimos. Por exemplo, carro pode ser visto com uma classe equivalente a automóvel.
- **Classes disjuntas** nos diz que se classes A e B são disjuntas então se p pertence a A então p não pertence a B pois A e B são disjuntas.

# Edição de classe

- **União de, Intersecção de, Complemento de** são combinações booleanas. Por exemplo, usando união de podemos dizer que uma classe contém coisas que são ou CidadãosBrasileiros ou CidadãosMexicanos. Usando complemento de podemos dizer que criança não é um cidadão idoso. Cidadania brasileira poderia ser definida como a união da cidadania de todos os estados brasileiros.

## 2.2 – Edição de classe

Para editar uma classe é necessário pressionar o botão direito do mouse sobre a classe que se deseja editar assim como na **figura 17**.

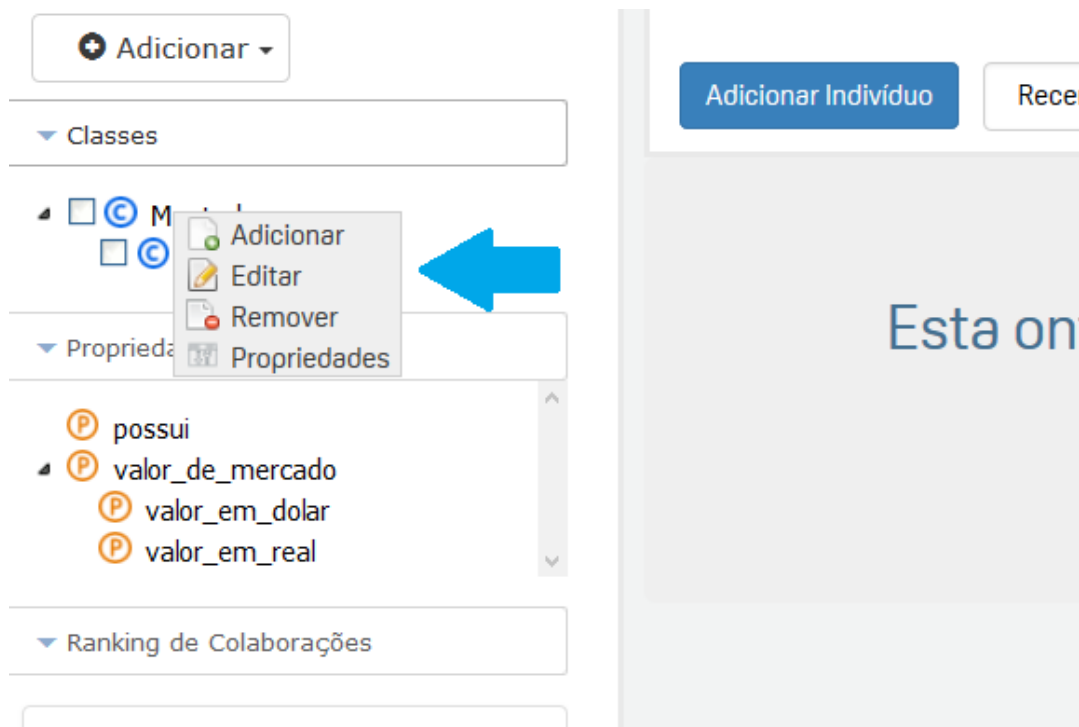


Figura 17: Editar classe

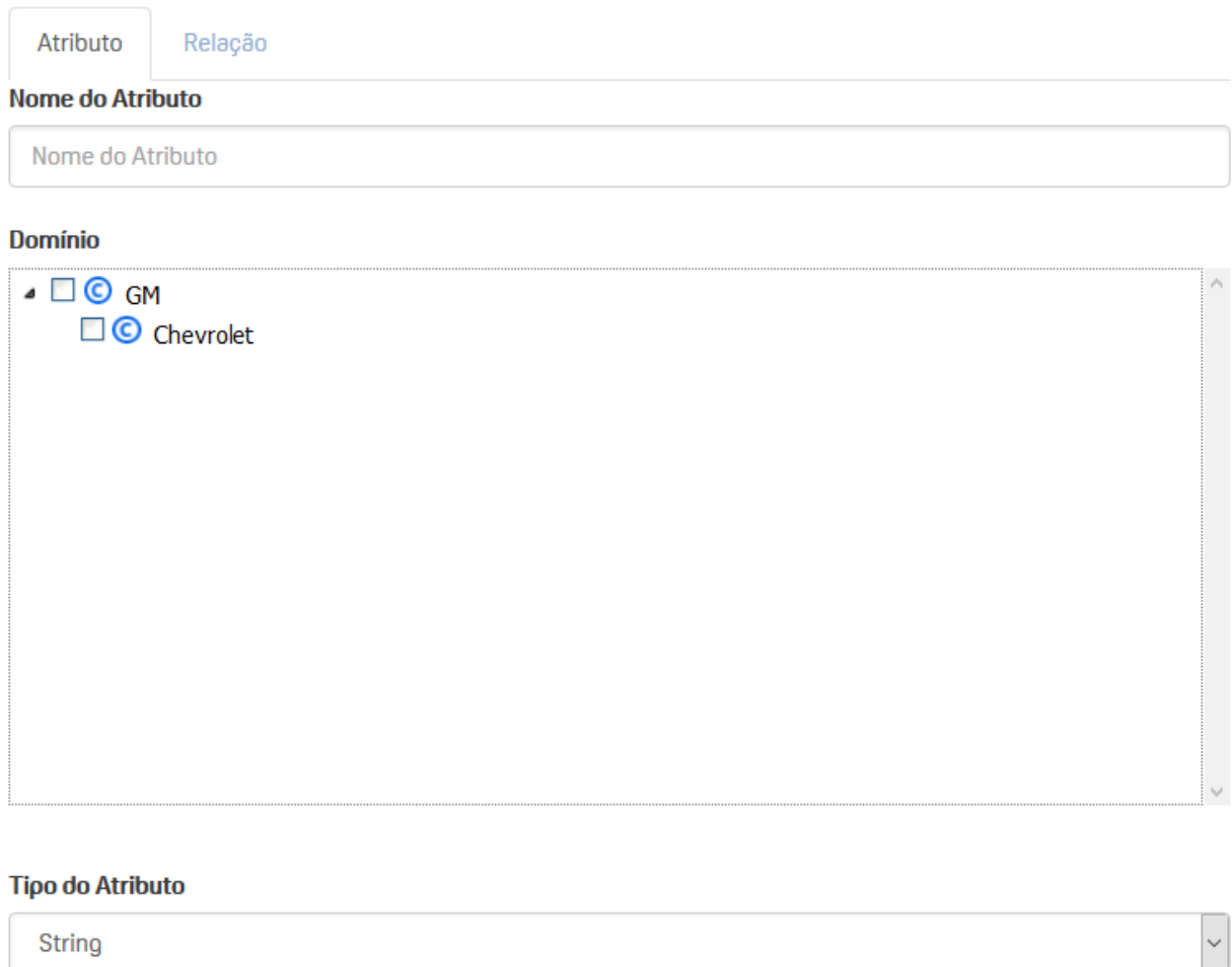
Após clicar em editar a tela de edição será aberta.



# Propriedades

## 3 – Propriedades

Podemos criar propriedade de classe clicando na opção **propriedade**, exibida na figura 11. A **figura 18** mostra o menu que será exibido.



The form is titled "Propriedades" and has two tabs: "Atributo" (selected) and "Relação". Below the tabs, there is a section labeled "Nome do Atributo" with a text input field containing the placeholder "Nome do Atributo". Below this is a section labeled "Domínio" with a list of items: "GM" and "Chevrolet", each with a checkbox and a circular icon. Below the domain list is a section labeled "Tipo do Atributo" with a dropdown menu showing "String".

Figura 18: Criação de propriedade

É possível criar tanto propriedades de atributo quanto propriedades de relação.

### 3.1 – Atributo

Para criar uma propriedade de atributo é necessário dar um nome a essa propriedade e escolher que tipo de atributo será esse, os seguintes tipos são possíveis:

- String: consiste em uma sequência de caracteres
- Boolean: é um tipo de dado que aceita apenas dois valores 0 (Falso) ou 1 (Verdadeiro)

# Propriedades

- Data: utilizado para guardar arquivos binários (foto, vídeo, música etc.)
- Datetime: utilizado para guardar data e hora
- Float: utilizado para guardar números de ponto flutuante.
- Integer: utilizado para guardar números inteiros
- Time: utilizado para guardar valores de hora (hora: minuto: segundo)

Outra opção que se tem é a seleção do domínio do atributo, quando alguma classe é selecionada como domínio logo o atributo que está sendo criado se torna uma propriedade daquela classe, caso nenhuma classe seja selecionada então o atributo será criado sem ligação com classe alguma.

Algumas opções avançadas também estão disponíveis durante a criação, como pode ser visto na **figura 19**.

Opções avançadas ▼

## Descrição

Descreva sua Propriedade

## Propriedade pai

☒ Sem parentesco

## Cardinalidade


Selecione...

Valor

## Restrições

☐ Funcional

Propriedade Equivalente

☐  modelo

☐  Valor

Figura 19: Opções avançadas de criação de atributo

# Propriedades

As opções avançadas não são de preenchimento obrigatório. É possível adicionar uma descrição a propriedade, apenas para fins de documentação, definir se essa propriedade possui algum parentesco, definir a cardinalidade do atributo e definir restrições. As opções de cardinalidade são as seguintes:

- Cardinalidade: um indivíduo deve ter  $X$  atributos desse tipo definidos.
- Cardinalidade máxima: um indivíduo pode ter no máximo  $X$  atributos desse tipo e no mínimo zero.
- Cardinalidade mínima: um indivíduo deve ter no mínimo  $X$  atributos desse tipo.

Dentre as restrições temos a **funcional** e a **propriedade de equivalência**. Quando um atributo é definido como funcional ele se torna de preenchimento obrigatório. Definir um atributo como equivalente a outro, ou seja, utilizar a propriedade de equivalência, indica que aqueles atributos podem ser considerados iguais, equivalentes.

## 3.2 – Relação

A tela de criação de um atributo de relação pode ser visto na **figura 20**. Um atributo de relação deve possuir um nome e qual relação ele faz.

Atributo | Relação | Metadado de termo

**Nome da Relação**

Nome da Relação

**Domínio**

- ☐ Montadora
- ☐ Marca

**Relação**

- ☐ Montadora
- ☐ Marca

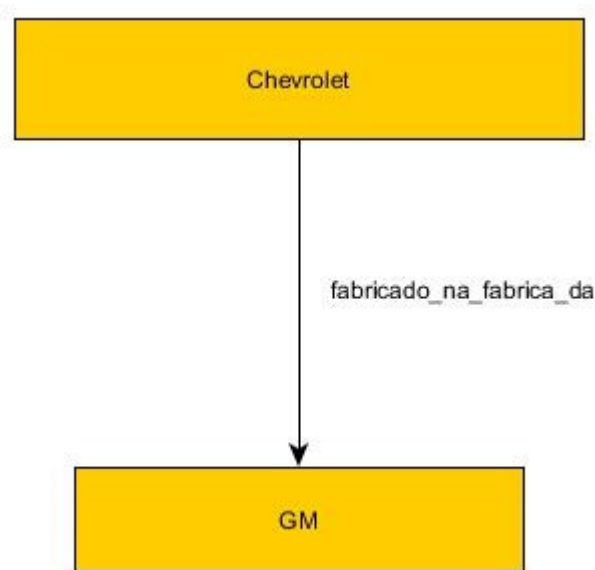
Figura 20: Criação de atributo de relação

# Propriedades

Na caixa domínio deve ser selecionado quem é o “dono” da relação e na caixa relação deve ser selecionado quem “recebe” a relação. Exemplo:

1. Criar a relação *fabricado\_na\_fabrica\_da*, com domínio **Marca** e relação **Montadora**
2. Criar indivíduo de Montadora chamado *GM*
3. Criar o indivíduo de Marca chamado *Chevrolet* com a relação *fabricado\_na\_fabrica\_da* apontando para GM

Como resultado teremos grafo (simplificado) de relação apresentado na **figura 21**.



*Figura 21: Grafo de relacionamento*

Assim como os atributos as relações também possuem opções avançadas, mostrado na **figura 22**.

# Propriedades

The form is titled 'Propriedades' and contains several sections for configuring a relationship:

- Descrição:** A text input field with the placeholder 'Descreva sua Propriedade'.
- Propriedade pai:** A checkbox labeled 'Sem parentesco'.
- Cardinalidade:** A dropdown menu with 'Selecione...' and a 'Valor' input field.
- Propriedade Inversa:** A dropdown menu with the text 'Nenhuma propriedade encontrada para a classe de destino selecionada'.
- Restrições:** Three checkboxes labeled 'Funcional', 'Transitiva', and 'Simétrica', followed by another dropdown menu with 'Selecione...'.

A blue 'Salvar' button is located at the bottom right of the form.

Figura 22: Criação de relação, opções avançadas

Nas opções avançadas há a opção de adicionar uma descrição, apenas para fins de documentação. É possível adicionar também uma propriedade pai, ou seja, definir que a esse novo atributo de relacionamento herda de alguma outra relação.

Há três opções de cardinalidade, são elas:

- Cardinalidade: o novo indivíduo deve ter relação com exatamente X indivíduos.
- Cardinalidade Mínima: o novo indivíduo deve ter relação com no mínimo X indivíduos.
- Cardinalidade Máxima: o novo indivíduo deve ter relação com no máximo X indivíduos e no mínimo zero.

A opção de **propriedade inversa** faz com que os dois elementos da relação possuam referência entre si.

É possível configurar uma relação com as seguintes restrições:

# Propriedades

- **Funcional:** torna o preenchimento da relação obrigatório.
- **Transitiva:** uma propriedade pode ser transitiva, se  $P$  é uma propriedade transitiva e os pares  $(x, y)$ ,  $(y, z)$  são instâncias de  $P$  então o par  $(x, z)$  também é uma instância de  $P$ .
- **Simétrica:** define uma relação valida tanto de  $Y$  para  $X$  quanto de  $X$  para  $Y$ . Exemplo, a relação de amizade é simétrica pois se Pedro é amigo de Deborah então Deborah é amiga de Pedro.
- **Propriedade equivalente:** indica que aquela relação pode ser considerada equivalente a relação selecionada.
- **Todos os valores de:** restringe uma relação a certos tipos de valores, por exemplo a classe *pessoa* pode ter a propriedade *tem\_filha* restringida a classe *mulher*.
- **Alguns valores de:** restringe uma relação a ter, pelo menos, um tipo de valor dentre os selecionados.
- **Tem valor:** restringe uma relação a ter um valor específico.

## 3.3 – Edição de propriedade

Após uma propriedade ser criada é possível editar essa propriedade o processo é muito similar a edição de atributos, é necessário apenas que o usuário clique com o botão direito do mouse sobre a propriedade que se deseja editar e clicar em editar (**figura 23**)

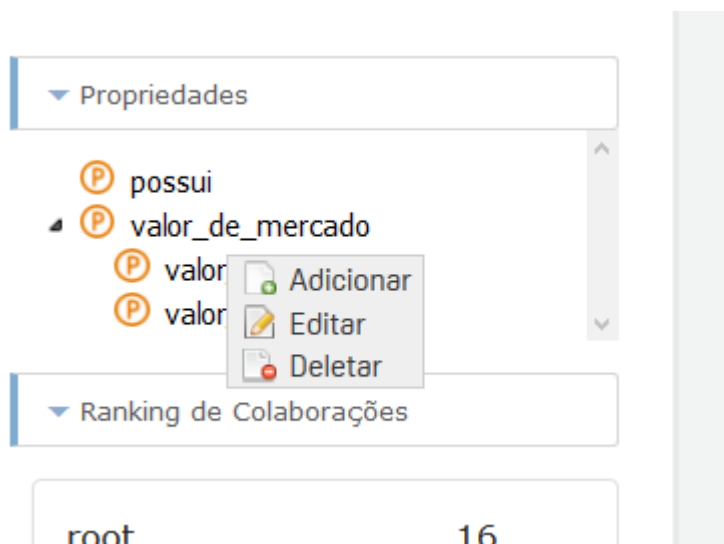


Figura 23: Edição de propriedade

# Indivíduo

## 4 – Indivíduo

Na imagem da figura 9 podemos ver o botão adicionar indivíduo, podemos adicionar um novo indivíduo clicando em alguma classe criada pelo usuário e em seguida clicando nesse botão (**Figura 24**).

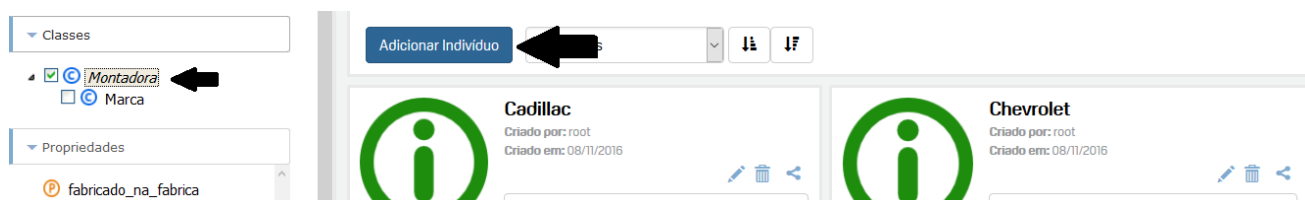


Figura 24: Adicionar novo indivíduo

A tela da **figura 25** será exibida.

Figura 25: Adicionar indivíduo

Os campos presentes durante a adição de um novo indivíduo já foram explicados anteriormente, a única novidade aqui é o campo **valor**, este campo é uma propriedade que foi adicionada a classe montadora. Todas as propriedades adicionadas a uma classe serão adicionadas a tela de cadastro de indivíduo daquela determinada classe.



# Exportar

Após a criação de uma ontologia é possível exportar essa ontologia criada. É possível exportar uma ontologia em formato **RDF**, **CSV** ou até mesmo gerar um grafo orientado da ontologia criada. Para realizar a exportação basta apenas abrir a ontologia que se deseja exportar e em seguida clicar no botão indicado na **figura 26**. Lá podemos ver a opções de exportação disponíveis.

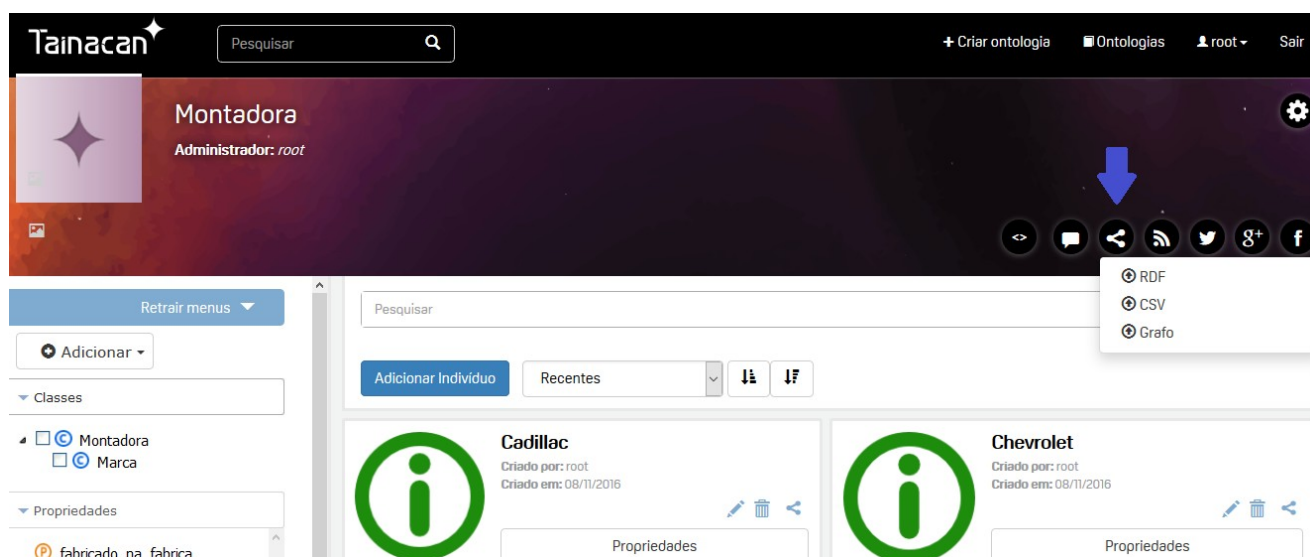


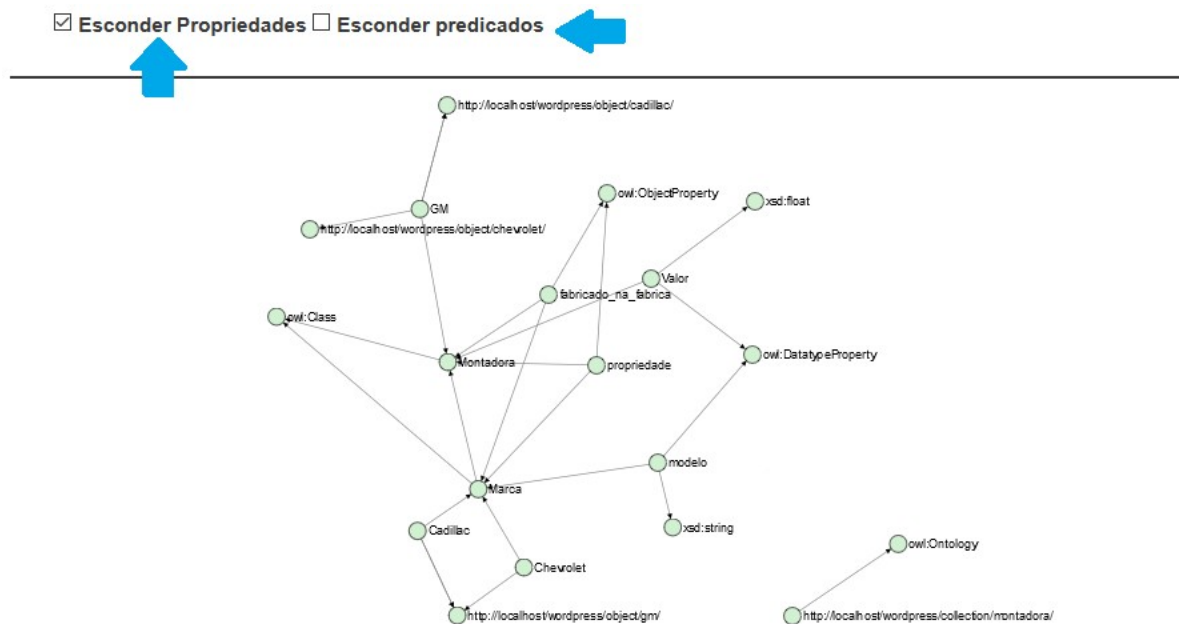
Figura 26: Exportar ontologia

Quando a opção RDF é escolhida uma nova aba, que exibe o arquivo RDF corresponde, é aberta no navegador. Caso a opção CSV seja selecionada então o arquivo .csv será baixado para o computador.

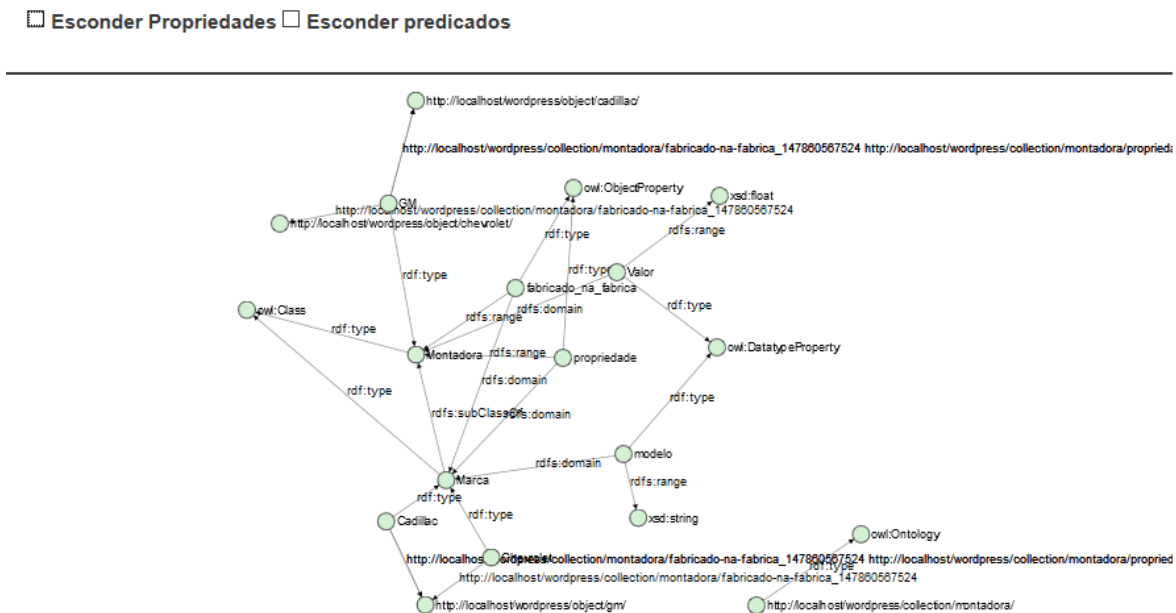
A opção grafo gera um grafo que é exibido no próprio navegador, um exemplo pode ser visto na **figura 27**. Após a geração do grafo há duas opções que podem auxiliar a exibição:

- **Esconder propriedades**, um grafo com as propriedades sendo exibidas pode ser visto na **figura 28**.
- **Esconder predicados**, quando a opção esconder predicados é selecionada então uma segunda caixa é aberta para que o usuário possa escolher quais predicados deseja esconder, a **figura 29** mostra um exemplo.

# Exportar



*Figura 27: Grafo gerado*



*Figura 28: Grafo com propriedades sendo exibidas*

# Exportar

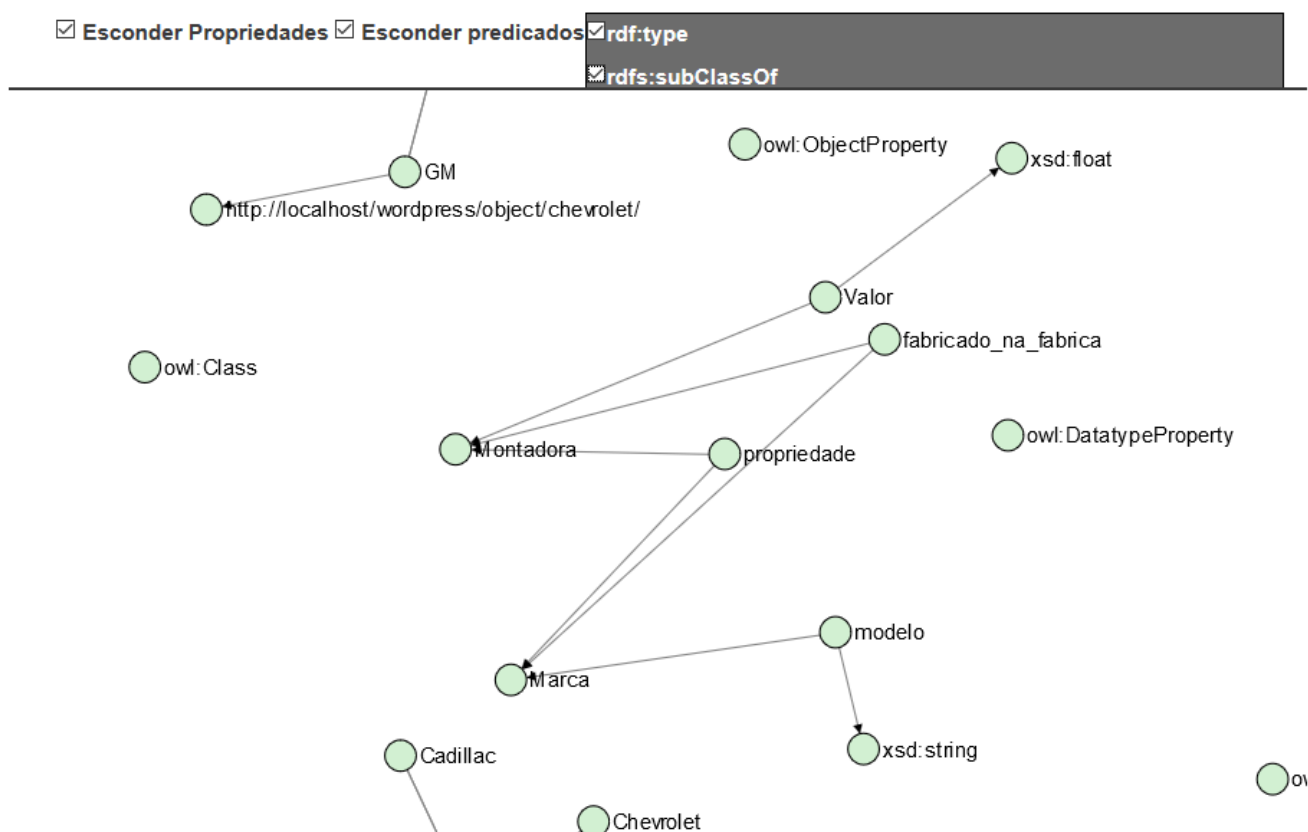


Figura 29: Esconder predicados

Além de poder exportar toda uma ontologia o usuário tem a opção de exportar uma classe (figura 30), propriedade (figura 31) e indivíduos (figura 32).

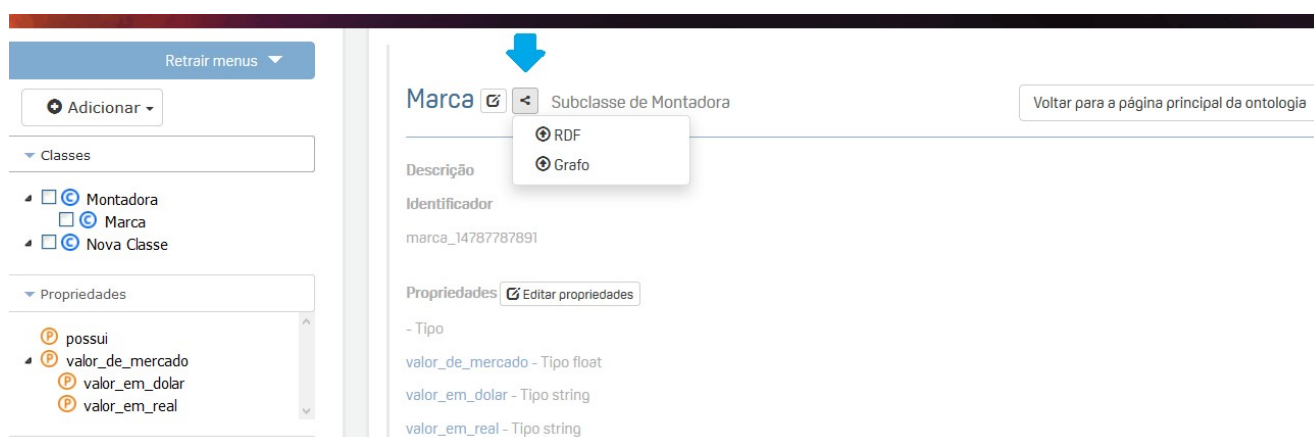


Figura 30: Exportar classe

# Exportar

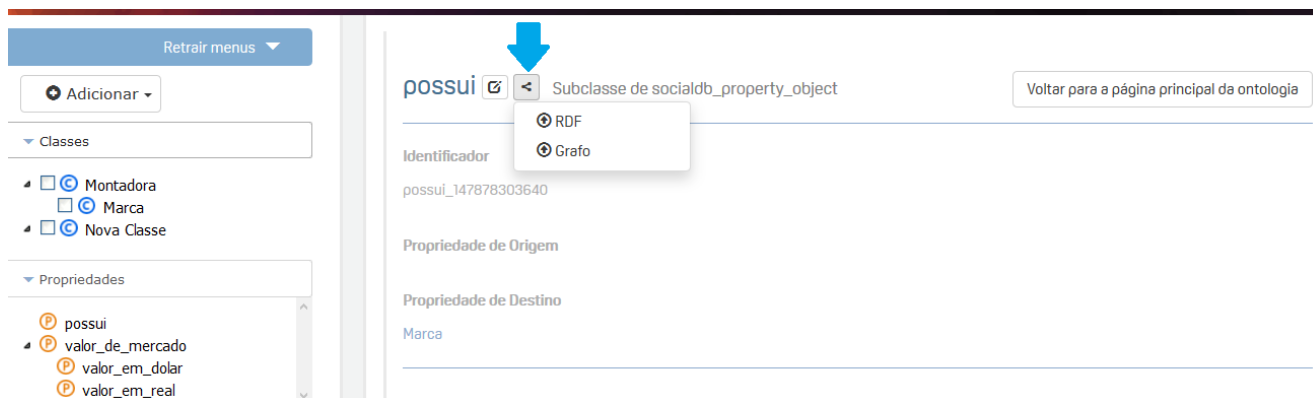


Figura 31: Exportar propriedade



Figura 32: Exportar indivíduo

Ao exportar uma propriedade, classe ou indivíduo o usuário pode escolher se exporta em RDF ou se gera o grafo do elemento.

# Compartilhamento

O Tainacan permite que o usuário compartilhe ontologias (**figura 33**) e indivíduos (**figura 34**) de ontologia nas redes sociais.

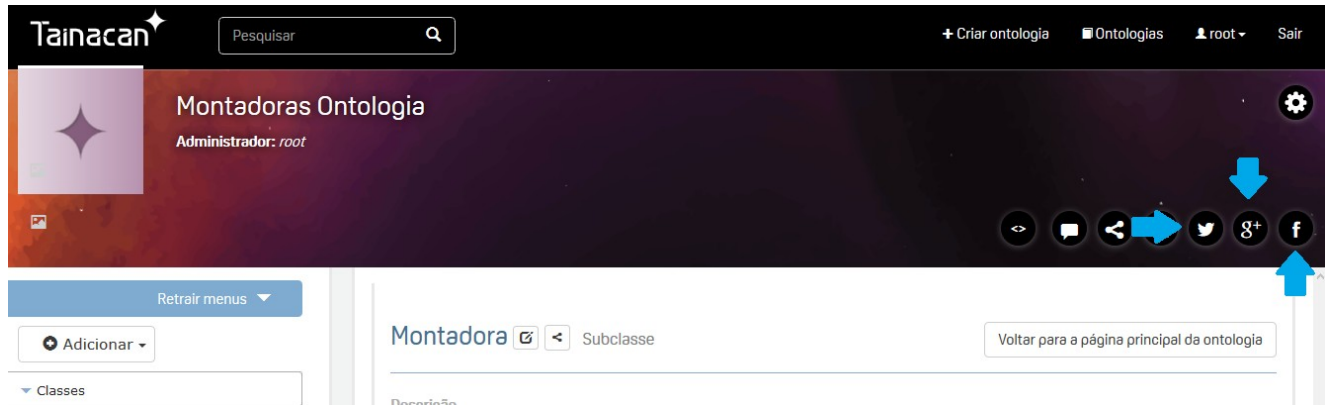


Figura 33: Compartilhamento de ontologia

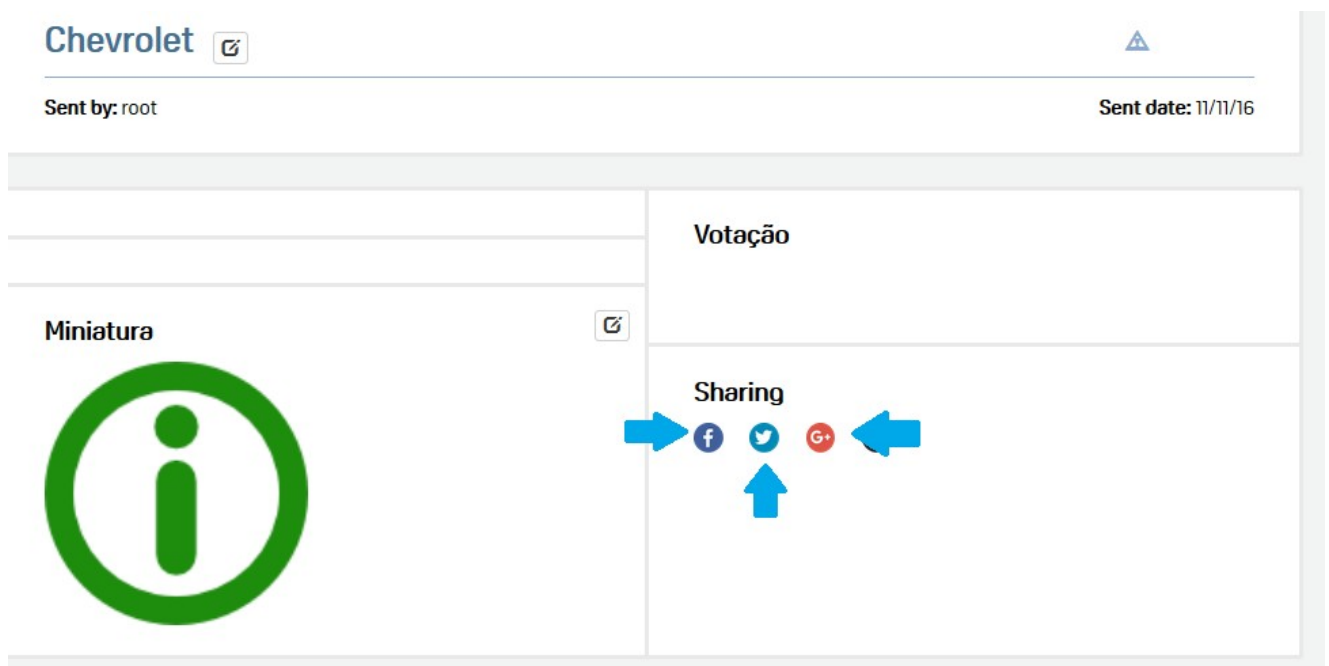


Figura 34: Compartilhando indivíduo

O Tainacan permite o compartilhamento em três redes: Facebook, Twitter e Google+.

# Config. Ontologia

Há diversas opções que podem ser modificadas sobre uma ontologia para fazer que ela se adeque às suas necessidades. Para acessar as configurações avançadas de uma ontologia deve-se inicialmente clicar na engrenagem apresentada na figura 6 e seguir os seguintes passos **Configurações > Configurações avançadas** então será exibida a tela apresentada na **figura 35**.

Endereço da ontologia ⓘ

http://localhost/wordpress/collection/ montadora

Objeto da ontologia

Item

Privacidade da ontologia

Público

Ontologia Pai

Ontologia raiz

Hierarquia da ontologia ⓘ

Sim

Tipo de Moderação

Aprovação por moderador

Moderadores da ontologia

Digite os três primeiras letras do nome do usuário

Permissões - Escolha as permissões para cada uma das ações abaixo

| Entidade          | Criar   | Editar    | Deletar   |
|-------------------|---------|-----------|-----------|
| Classe            | Membros | Aprovação | Aprovação |
| Classificação     | Membros |           | Aprovação |
| Indivíduo         | Membros |           | Aprovação |
| Comentários       | Anônimo | Aprovação | Aprovação |
| Tags              | Anônimo | Aprovação | Aprovação |
| Atributo          | Membros | Aprovação | Aprovação |
| Valor do Atributo |         | Aprovação |           |
| Relação           | Membros | Aprovação | Aprovação |
| Valor da Relação  |         | Aprovação |           |
| Metadado de termo | Membros | Aprovação | Aprovação |

Entidade

Criar

Editar

Deletar

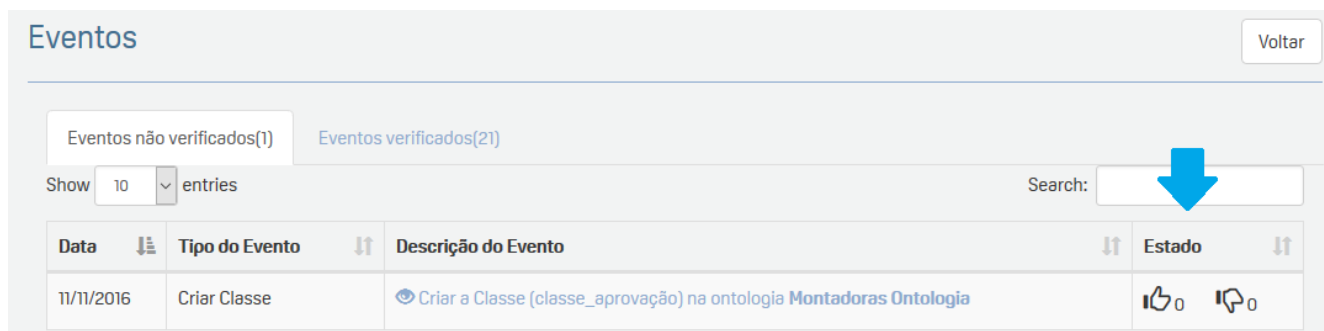
Salvar e Continuar Salvar

Figura 35: Configurações avançadas

# Config. Ontologia

Há muitas opções que avançadas, vamos comentar a função de cada uma.

- **Endereço da ontologia:** utilizado para se definir o endereço da ontologia, como o próprio nome diz, este endereço é o link com o qual o usuário poderá ter acesso a aquela ontologia.
- **Objeto da ontologia:** pode-se definir o nome com o qual os objetos de uma ontologia serão referenciados.
- **Privacidade da ontologia:** uma ontologia pode ser pública ou privada. Quando uma ontologia é pública então ela pode ser acessada por qualquer pessoa, quando privada somente o criador da ontologia pode acessá-la.
- **Ontologia pai:** define se a ontologia herda de alguma ontologia já criada.
- **Hierarquia da ontologia:** define se a ontologia poderá ser herdada por alguma outra ontologia.
- **Tipo de moderação (figura 36):** há duas opções disponíveis: aprovação por moderador e Aprovação democrática (eleição por maioria). Sempre que alguém sugerir alguma modificação na ontologia evento são gerados para que o moderador decida se aplica ou não aquela alteração, isso na opção aprovação por moderador. Quando a opção escolhida é Aprovação democrática (eleição por maioria) então é realizada uma votação entre os moderadores, por um tempo determinado anteriormente, para decidir se a modificação é ou não aprovada. Em caso de empate a modificação é aplicada.



The screenshot shows a web interface titled 'Eventos'. At the top right is a 'Voltar' button. Below the title, there are two tabs: 'Eventos não verificados(1)' and 'Eventos verificados(21)'. Below the tabs, there is a 'Show' dropdown set to '10' and the text 'entries'. To the right is a 'Search:' input field with a blue arrow pointing to it. Below this is a table with the following columns: 'Data', 'Tipo do Evento', 'Descrição do Evento', and 'Estado'. The table contains one row with the following data: '11/11/2016', 'Criar Classe', 'Criar a Classe (classe\_aprovação) na ontologia Montadoras Ontologia', and a status icon with '0' votes.

Figura 36: Aprovação democrática

- **Moderadores da ontologia:** pode ser definido quem serão os moderadores da ontologia.
- **Permissões:** para cada entidade podem haver três ações disponíveis criar, editar e excluir. Para cada ação há quatro tipos de permissão anônimo, aprovação, membros e não permitido. Quando uma permissão está restrita a:
  - **Anônimo:** qualquer usuário pode realizar a ação.
  - **Aprovação:** qualquer usuário pode solicitar a realização daquela ação mas antes de ser aplicada ela deverá ser aprovada pelo moderador.
  - **Membros:** apenas usuários cadastrados podem realizar a ação.
  - **Não permitido:** a ação não pode ser realizada de forma alguma.

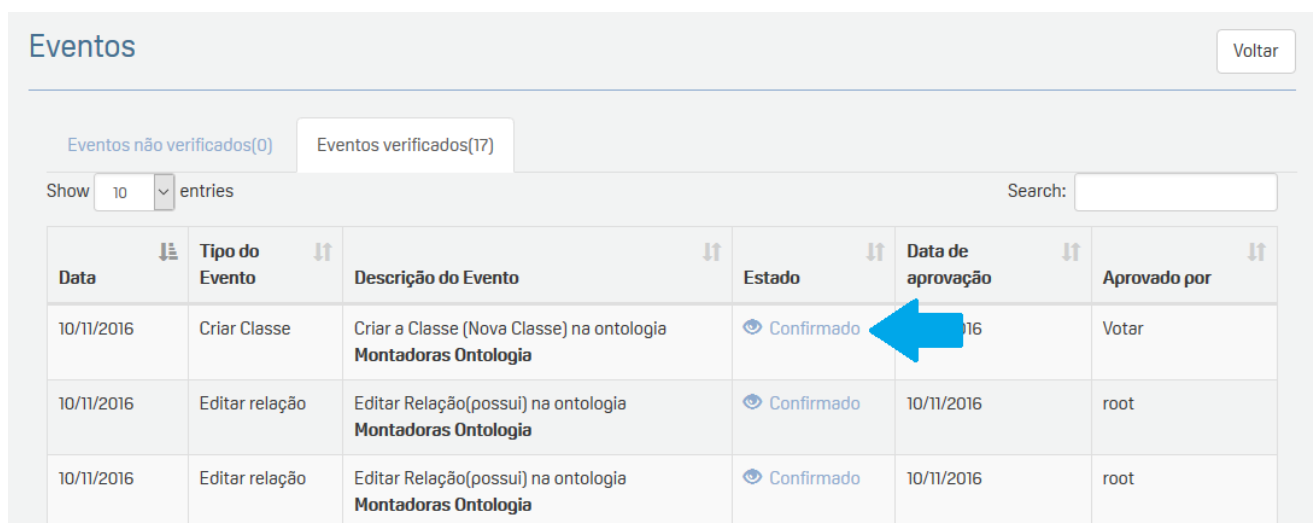


# Eventos

Algumas ações realizadas geram eventos para o moderador, com base nas permissões citada anteriormente. As entidades que podem gerar eventos são as seguintes:

- Classe
- Classificação
- Indivíduo
- Comentários
- Tags
- Atributo
- Valor do atributo
- Relação
- Valor de relação
- Metadado de termo

Quando a permissão para uma ação é colocada para que qualquer pessoa possa realizar (**Anônimo**) então o evento gerado vai direto para os eventos verificados, ou seja, eventos que já foram aprovados.



| Data       | Tipo do Evento | Descrição do Evento  | Estado     | Data de aprovação | Aprovado por |
|------------|----------------|--|------------|-------------------|--------------|
| 10/11/2016 | Criar Classe   | Criar a Classe (Nova Classe) na ontologia Montadoras Ontologia | Confirmado | 10/11/2016        | Votar        |
| 10/11/2016 | Editar relação | Editar Relação(possui) na ontologia Montadoras Ontologia       | Confirmado | 10/11/2016        | root         |
| 10/11/2016 | Editar relação | Editar Relação(possui) na ontologia Montadoras Ontologia       | Confirmado | 10/11/2016        | root         |

Figura 37: Evento com permissão anônima

A **figura 37** mostra um exemplo da criação de uma classe, a permissão de criação de classes foi definido como anônimo logo o evento não precisa ser aprovado.

Se a permissão for definida como aprovação então qualquer usuário pode realizar uma ação mas para que essa ação seja de fato realiza ela deve ser aprovada pelo moderador da coleção. A **figura 38** mostra um evento pendente após a realização de alguma ação que exige aprovação.

# Eventos

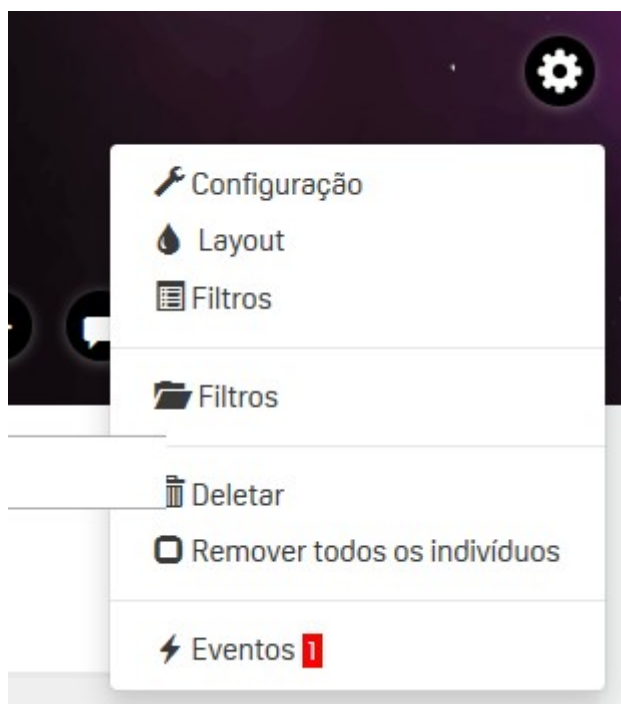


Figura 38: Evento pendente

Após clicar em **Eventos** o usuário poderá ver a lista de eventos não verificados (**figura 39**).

## Eventos

Voltar

Eventos não verificados(1)

Eventos verificados(17)

Show 

10

 entries 

Search:

| Data       | Tipo do Evento | Descrição do Evento   | Estado         |
|------------|----------------|---|----------------|
| 10/11/2016 | Criar Classe   | Criar a Classe (Classe por aprovação) na ontologia <a href="#">Montadoras Ontologia</a> | Não verificado |

Showing 1 to 1 of 1 entries

Previous

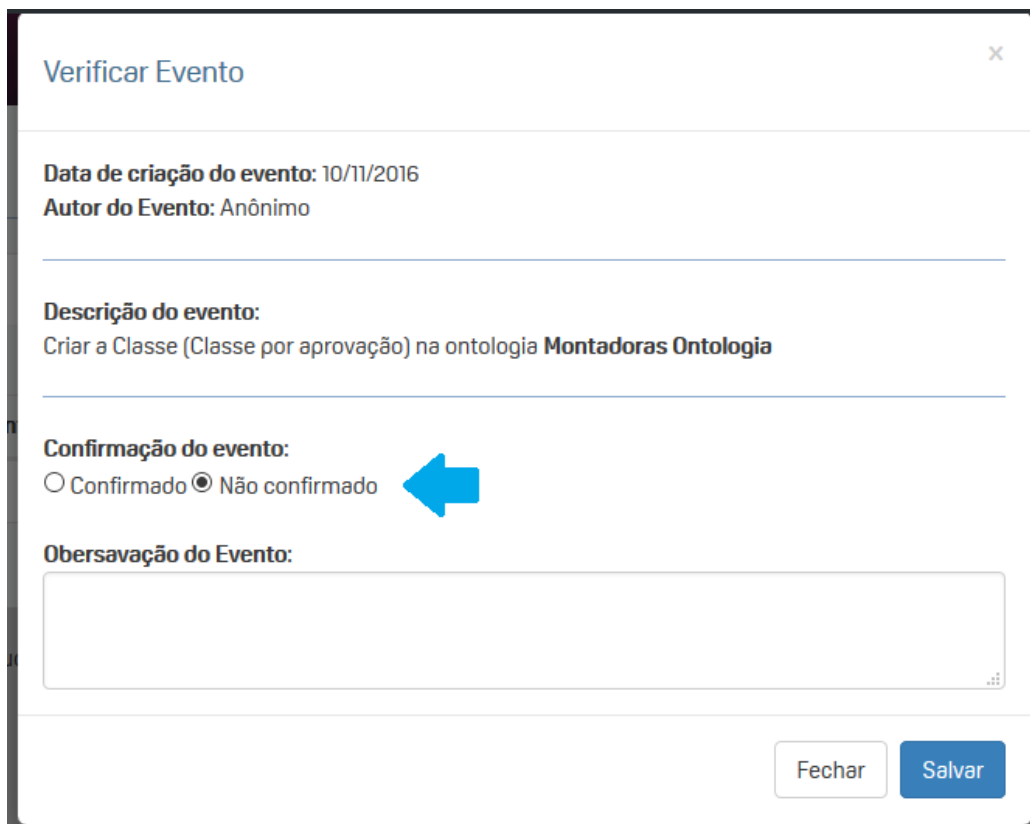
1

Next

Figura 39: Lista de eventos não verificados

Ao clicar em um dos eventos não verificados então o moderador poderá decidir se aprova ou não a execução daquele evento (**Figura 40**).

# Eventos



Verificar Evento

Data de criação do evento: 10/11/2016  
Autor do Evento: Anônimo

Descrição do evento:  
Criar a Classe (Classe por aprovação) na ontologia **Montadoras Ontologia**

Confirmação do evento:  
☐ Confirmado ☒ Não confirmado

Obersavação do Evento:

Fechar Salvar

Figura 40: Tela de verificação de evento

Na figura 33 podemos ver a área de confirmação de evento, lá há duas opções **Confirmado** e **Não confirmado**, caso a opção não confirmado permaneça selecionada então a ação contida no evento será negado, caso a opção **Confirmado** seja selecionada e em seguida o botão salvar seja clicado então a ação do evento será executada e o evento irá para a aba de eventos verificados.

# Comentários

Os usuários podem realizar comentários sobre o que foi publicado em uma ontologia, uma classe e uma propriedade. Isso tem o intuito de aumentar a colaboração e expressar opiniões sobre algo que foi publicado. As **figuras 41, 42 e 43** mostram a área de comentário de relação, indivíduo e classe respectivamente. Um usuário pode deixar um comentário digitando o que deseja comentar na caixa de texto logo abaixo do texto *leave your comment* e em seguida clicando no botão comentar.

possui   Subclasse de socialdb\_property\_object

Voltar para a página principal da ontologia

---

Identificador

possui\_147878303640

Propriedade de Origem

Propriedade de Destino

Marca

---

Leave your comment

Logged in as root.



Comentar


Figura 41: Comentário em relação

# Comentários

|                                      |  |
|--------------------------------------|--|
| Campo vazio                          | Campo vazio                            |
| <b>possui</b><br>Campo vazio         | <b>valor_de_mercado</b><br>Campo vazio |
| <b>valor_em_dolar</b><br>Campo vazio | <b>valor_em_real</b><br>Campo vazio    |



---

Leave your comment Logged in as root.



Comentar

Figura 42: Comentário em indivíduo


**Nova Classe**   Subclasse Voltar para a página principal da ontologia

---

**Descrição**

**Identificador**

nova-classe\_147879892482


**Propriedades**  Editar propriedades

---

**Indivíduos**

---

Leave your comment Logged in as root.



Comentar

Figura 43: Comentário em classe

# API

A API do Tainacan foi construída a partir de um fork sobre o plugin WP-API, há dois plugins mais conhecidos que permitem que o wordpress seja considerado uma aplicação Restful, o Thermal API e WPI-API. O segundo foi o escolhido para adequar-se ao Tainacan, por ser a API com a maior comunidade, maior número de downloads e principalmente, por possuir uma documentação mais completa.

## Alterações realizadas

Para o funcionamento completo da API abrangendo todo Tainacan, foi necessário modificações em partes do código-fonte. Principalmente pelo fato que o Tainacan implementa uma nova tabela de metadados para taxonomias e um grande conjunto de novos tipos de post e taxonomias.

No arquivo principal do plugin foi alterada a função *json\_api\_default\_filters*, para aceitar os filtros para leitura, adição, edição e exclusão de metadados de taxonomia. Foi incluído a classe *WP\_JSON\_Meta\_Taxonomy* que implementa estes métodos que são utilizados nestes filtros. Esta classe estende a *WP\_JSON\_Meta* que também foi alterada para que pudesse 'enxergar' este novo tipo de metadado. Na classe *WP\_JSON\_Posts* foi alterada a função *get\_posts* para a aceitar a taxonomia *socialdb\_category\_type* quando for realizado uma requisição que utilize o parâmetro *term*.

Devido as alterações, o plugin não pode ser atualizado pois sobre escreveria todas as alterações realizadas, essa é uma versão única que funciona especificamente com o Tainacan.

## Autenticação

Há diferentes métodos de autenticação utilizados nesta API, como é mostrado nesta referência, [<http://wp-api.org/guides/authentication.html>], a mais simples foi criada especialmente para o Tainacan. O usuário deve estar logado diretamente na página e passar seu login e senha, em suas requisições

## Usando a API

### Repositório

- **Listar todas as coleções ativas**
  - Autenticação: não necessária
  - Tipo de requisição: GET
  - Link: [ URL ]/wp-json/posts?type=socialdb\_collection
  - Parâmetros de filtragem: [http://wp-api.org/#posts\\_retrieve-posts](http://wp-api.org/#posts_retrieve-posts)
- **Listar todos os objetos do repositório**
  - Autenticação: não necessária
  - Tipo de requisição: GET
  - Link: [ URL ]/wp-json/posts?type=socialdb\_object
  - Parâmetros de filtragem: [http://wp-api.org/#posts\\_retrieve-posts](http://wp-api.org/#posts_retrieve-posts)
- **Listar metadados do repositório**
  - Autenticação: necessária

- Tipo de requisição: GET
- Link: :[URL]/wp-json/taxonomies/socialdb\_category\_type/terms/**categoria\_raiz\_repositório**/meta
- Observação: Esta ação exige que seja recuperado a categoria raiz de todo o repositório , esta categoria é o parent da categoria raiz das coleções, nas próximas seções é mostrado como recuperar uma categoria raiz de uma coleção. Para achar seu parent basta realizar uma requisicao GET [URL]/wp-json/taxonomies/socialdb\_category\_type/terms/**categoria\_raiz\_colecao**

### Metadados

- **Busca o metadado, trazendo seu nome e seu slug**
  - Autenticação: não necessária
  - Tipo Requisição: GET
  - Link: [URL]/wp-json/taxonomies/socialdb\_property\_type/terms/metadado\_id/
- **Lista os metadados de um metadado**
  - Autenticação: Necessária
  - Tipo Requisição: GET
  - Link: [URL]/wp-json/taxonomies/socialdb\_property\_type/terms/metadado\_id/meta

### Coleção

- **Busca apenas uma coleção**
  - Autenticação: não necessária
  - Tipo Requisição: GET
  - Link: [ URL ]/wp-json/posts/ID
- **Busca os metadados de uma coleção, importante para buscar o ID da categoria raiz** ( meta: socialdb\_collection\_object\_type)
  - Autenticação: Necessária
  - Tipo Requisição: GET
  - Link: [ URL ]/wp-json/posts/ID/meta
- **Lista todos os itens de uma coleção**
  - Autenticação: Necessária
  - Tipo Requisição: GET
  - Link:[URL]/wp-json/posts?type=socialdb\_object&filter[term]=Categoria\_Raiz\_Coleção&filter[posts\_per\_page]=-1
  - Parâmetros para filtragem: [http://wp-api.org/#posts\\_retrieve-posts](http://wp-api.org/#posts_retrieve-posts)
  - Exemplo:
    - Filtrando pela palavra-chave 'cultura' e ordenando pelo título do item
    - [URL]/wp-json/posts?type=socialdb\_object&filter[term]=Categoria\_Raiz\_Coleção&filter[posts\_per\_page]=-1&filter[s]=cultura&filter[orderby]=title



- **Adicionar um item na coleção**
  - Autenticação: Necessária
  - Tipo Requisição: POST
  - Link: [ URL ]/wp-json/posts/
    - **Content-Type** →application/json
      - Dados no corpo da requisição para inserção
      - {"title":"**Título**","content\_raw":"**Conteúdo**","status":"publish","type":"socialdb\_object","x-categories":["**Categoria\_Raiz\_Coleção**"],"post\_meta":{"key":"socialdb\_object\_dc\_type","value":"**TIPO**"},"key":"socialdb\_object\_content","value":"**LINK\_OU\_TEXTO**"},"key":"socialdb\_object\_dc\_source","value":"**FONTE**"},"key":"socialdb\_object\_from","value":"external"}}

### Item

- Busca um item
  - Autenticação: não necessária
  - Tipo Requisição: GET
  - Link: [ URL ]/wp-json/posts/ID
- Busca os metadados de um item
  - Autenticação: Necessária
  - Tipo Requisição: GET
  - Link: [ URL ]/wp-json/posts/ID/meta

### Categorias

- **Busca uma categoria a partir do seu ID**
  - Autenticação: não necessária
  - Tipo Requisição: GET
  - Link:[URL]/wp-json/taxonomies/socialdb\_category\_type/terms/ID
- **Busca os metadados de uma categoria**
  - Autenticação: Necessária
  - Tipo Requisição: GET
  - Link:[URL]/wp-json/taxonomies/socialdb\_category\_type/terms/ID/meta
- **Lista todos os itens vinculados à uma categoria**
  - Autenticação: Necessária
  - Tipo Requisição: GET
  - Link:[URL]/wp-json/posts?type=socialdb\_object&filter[term]=ID&filter[posts\_per\_page]=-1
- **Lista todas categorias descendentes**
  - Autenticação: Necessária
  - Tipo Requisição: GET

- Link:[URL]/wp-json/taxonomies/socialdb\_category\_type/terms?filter[child\_of]=ID
- **Lista todas categorias filhas (diretas)**
  - Autenticação: Necessária
  - Tipo Requisição: GET
  - Link:[URL]/wp-json/taxonomies/socialdb\_category\_type/terms?filter[parent]=ID
  - Observação: Outros filtros que podem ser utilizados são os mesmos da função `get_terms` nativa do WordPress

# Agradecimentos

## **Gestão e Orientação**

Dalton Lopes Matins  
Marcel Ferrante Silva

## **Desenvolvimento**

Eduardo Humberto Resende Guimarães  
Marcus Bruno Diniz Molinari  
Walison José de Deus  
Rodrigo de Oliveira Guimarães

## **Design**

Ravi Figueiredo Passos  
Sérgio  
Thayrone

## **Desenvolvimento do Manual**

Dalton Lopes Martins  
Marcel Ferrante Silva

## **Realização**

UFG – Universidade Federal de Goiás  
L3P – Laboratório de Políticas Públicas Participativas