**Source**

<http://stackoverflow.com/questions/5886738/setting-up-a-polymorphic-has-many-through-relationship>

<http://railscasts.com/episodes/154-polymorphic-association-revised?view=asciicast>

$ rails g model tagging taggable\_id:integer taggable\_type tag\_id:integer

$ rails g model tag name tag\_type

**Gemfile**

gem "select2-rails", "~> 3.1.1"

**application.js**

//= require select2

**application.css.scss**

//= require select2

**models/tag.rb**

# == Schema Information

#

# Table name: tags

#

# id :integer not null, primary key

# name :string(255)

# created\_at :datetime not null

# updated\_at :datetime not null

# tag\_type :string(255)

#

class Tag < ActiveRecord::Base

# Attributes

attr\_accessible :name, :tag\_type

# Relations

has\_many :taggings, :dependent => :destroy

has\_many :projects, :through => :taggings, :source => :taggable, :source\_type => 'Project'

has\_many :tasks, :through => :taggings, :source => :taggable, :source\_type => 'Task'

#Named scopes

scope :name\_like, lambda { |n| where("name ilike ?", "%#{n}%") } # WARN: Potential DB Change Problem

end

**models/tagging.rb**

# == Schema Information

#

# Table name: taggings

#

# id :integer not null, primary key

# taggable\_id :integer

# taggable\_type :string(255)

# tag\_id :integer

# created\_at :datetime not null

# updated\_at :datetime not null

#

class Tagging < ActiveRecord::Base

attr\_accessible :tag\_id, :taggable\_id, :taggable\_type

belongs\_to :taggable, :polymorphic => true

belongs\_to :tag

end

**model/project.rb**

attr\_accessible: :tag\_names

**attr\_accessor** :tag\_names

after\_save :assign\_tags

has\_many :taggings, :as **=>** :taggable, :dependent **=>** :destroy

has\_many :tags, :through **=>** :taggings

accepts\_nested\_attributes\_for :tags

*# Named Scopes*

scope :name\_like, lambda { **|**n**|** where("name ilike ?", "%#{n}%") } *# WARN: Potential DB Change Problem*

**private**

**def** **assign\_tags**

**if** @tag\_names

self**.**tags **=** @tag\_names**.**split(',')**.**map **do** **|**name**|**

Tag**.**find\_or\_create\_by\_name(name, :tag\_type => 'project')

**end**

**end**

**end**

**views/project/\_form.html\_haml**

- @tags\_array = Tag.find\_all\_by\_tag\_type('project').map(&:name).sort

:javascript

$(document).ready(function() {

$(".typeahead").tokenInput("/users.json", {

minChars: 1,

hintText: "Type name or username...",

theme: "facebook",

prePopulate: $(".typeahead").data("load"),

preventDuplicates: true,

resultsFormatter: function(item) {

return "<li>" + "<img src='" + "/assets/avatar.png" + "' title='" + item.name + "' height='25px' width='25px' />" + "<div style='display: inline-block; padding-left: 10px;'><div class='name'>" + item.name + "</div></div></li>";

}

});

$(".select\_tags").select2({

tags: #{@tags\_array},

tokenSeparators: [','],

placeholder: "Select existing tags or add new ones...",

width: '70%'

});

});

= simple\_form\_for @project do |f|

= f.error\_notification

.pull-right

- if @project.avatar

= image\_tag @project.avatar.image\_url(:thumb).to\_s

= f.fields\_for :avatar, @project.avatar || Avatar.new do |a|

= a.file\_field :image

.form-inputs

= f.input :name

= f.input :title, :input\_html => { :style => 'width:90%' }

= f.input :description, :as => :ckeditor, :input\_html => { :toolbar => 'Basic' }

= f.input :tag\_names, :input\_html => { :class => 'select\_tags', value: @project.tags.map(&:name).join(",") }, :label => 'Add Tags'

%br

= '\* Public Profile Url'

.input-prepend

%span.add-on

http://railsforcharity.org/

- if edit

= f.input :profile\_url, :label => false, :disabled => true

- else

= f.input :profile\_url, :label => false

%br

= f.input :website

= f.input :video

%br

= f.input :collaborator\_tokens, :input\_html => { :class => 'typeahead', data: { load: @project.users } }, :label => 'Add Collaborators'

- if !edit

%br

= f.input :terms, :label => false, :as => :boolean, :inline\_label => 'By clicking on the check box, you agree with the Terms and Conditions of RailsforCharity.'

-# TODO: Make Terms & Conditions Page

.form-actions

= f.button :submit, :class => 'btn btn-primary'

**Specs**

**specs/factories/taggings.rb**

FactoryGirl**.**define **do**

factory :tagging **do**

taggable\_id 1

taggable\_type "MyString"

tag\_id 1

**end**

**end**

**specs/factories/tags.rb**

FactoryGirl**.**define **do**

factory :tag **do**

name { words(1) }

**end**

**end**

**specs/models/project\_spec.rb**

describe 'tagging' **do**

it 'assigns tags' **do**

p **=** create(:project, :tag\_names **=>** 'abc, cde')

p**.**tags**.**size**.**should **==** 2

**end**

**end**

**specs/models/tagging\_spec.rb**

require 'spec\_helper'

describe Tagging **do**

it "assigns one tag" **do**

tag **=** create(:tag)

project **=** create(:project)

project**.**tags **<<** tag

tagging **=** Tagging**.**find\_by\_taggable\_type\_and\_taggable\_id('Project', project**.**id)

tagging**.**should\_not be\_nil

tagging**.**tag\_id**.**should **==** tag**.**id

**end**

**end**

Notes:

1) Whenever you want to delete an item that is linked by a link-model make sure to use "destroy." When you destroy a linked object, it will also destroy the link. This ensures that there are no bad or dead links. This is why we use ':**dependent => :destroy**'

2) When setting up our 'Project' model, which is one of our 'taggable' models, it must be linked using :as. Since in the preceeding example we used 'taggable\_type' and 'taggable\_id' we use :as => :taggable. This helps rails know how to store the values in the database.

3) When linking tags to projects, we use: has\_many :projects, :through => :taggings, :source => :taggable, :source\_type => 'Project' This tells the tag model that it should have many :projects through :taggings. The source is :taggable, for the same reason as above. The source-type is "Project" because a model will automatically set taggable\_type to its own name.

**tags table**

|  |  |  |
| --- | --- | --- |
| id | name | tag\_type |
| x |  | (project) |

has\_many :taggings, :dependent => :destroy

has\_many :projects, :through => :taggings, :source => :taggable, :source\_type => 'Project'

**taggings table**

|  |  |  |  |
| --- | --- | --- | --- |
| id | taggable\_id | taggable\_type | tag\_id |
|  | (project\_id) | (project) | x |

belongs\_to :taggable, :polymorphic => true

belongs\_to :tag

**projects**

has\_many :taggings, :as **=>** :taggable, :dependent **=>** :destroy

has\_many :tags, :through **=>** :taggings