class ListsController < ApplicationController

#before\_action :require\_login

def index

@list = List.new

@lists = List.all

end

def show

@list = List.find(params[:id])

@task = Task.new

end

def new

@list = List.new

@list.tasks.build

end

def create

@list = List.new(list\_params)

if @list.save

redirect\_to @list

else

render :new

end

end

def destroy

@list = List.find(params[:id])

@task = Task.find\_by(params[:id])

@task.destroy

redirect\_to @list

end

private

def list\_params

params.require(:list).permit(:name, tasks\_attributes: [:name, :status])

end

def require\_login

return head(:forbidden) unless session.include? :user\_id

end

end

class TasksController < ApplicationController

def index

@task = Task.all

end

def create

@list = List.find(params[:list\_id])

@task = @list.tasks.build(task\_params)

@task.save

redirect\_to @list

end

def edit

@task = Task.find(params[:id])

@task.update(task\_params)

redirect\_to list\_path(@task.list)

end

def destroy

@task = Task.find(params[:id])

@task.destroy

redirect\_to list\_path(@task.list)

end

private

def task\_params

params.require(:task).permit(:name, :status)

end

end

require File.expand\_path('../boot', \_\_FILE\_\_)

require 'rails/all'

# Require the gems listed in Gemfile, including any gems

# you've limited to :test, :development, or :production.

Bundler.require(\*Rails.groups)

module RailsGroupTaskManager

class Application < Rails::Application

# Settings in config/environments/\* take precedence over those specified here.

# Application configuration should go into files in config/initializers

# -- all .rb files in that directory are automatically loaded.

# Set Time.zone default to the specified zone and make Active Record auto-convert to this zone.

# Run "rake -D time" for a list of tasks for finding time zone names. Default is UTC.

# config.time\_zone = 'Central Time (US & Canada)'

# The default locale is :en and all translations from config/locales/\*.rb,yml are auto loaded.

# config.i18n.load\_path += Dir[Rails.root.join('my', 'locales', '\*.{rb,yml}').to\_s]

# config.i18n.default\_locale = :de

# Do not swallow errors in after\_commit/after\_rollback callbacks.

config.active\_record.raise\_in\_transactional\_callbacks = true

end

end

# SQLite version 3.x

# gem install sqlite3

#

# Ensure the SQLite 3 gem is defined in your Gemfile

# gem 'sqlite3'

#

default: &default

adapter: sqlite3

pool: 5

timeout: 5000

development:

<<: \*default

database: db/development.sqlite3

# Warning: The database defined as "test" will be erased and

# re-generated from your development database when you run "rake".

# Do not set this db to the same as development or production.

test:

<<: \*default

database: db/test.sqlite3

production:

<<: \*default

database: db/production.sqlite3

class List < ActiveRecord::Base

has\_many :tasks

has\_many :user\_lists

has\_many :users, through: :user\_lists

validates\_presence\_of :name, uniqueness: true

accepts\_nested\_attributes\_for :tasks

end

class User < ActiveRecord::Base

# Include default devise modules. Others available are:

# :confirmable, :lockable, :timeoutable and :omniauthable

devise :database\_authenticatable, :registerable,

:recoverable, :rememberable, :trackable, :validatable

devise :omniauthable, :omniauth\_providers => [:facebook]

has\_many :user\_lists

has\_many :lists, through: :user\_lists

def self.from\_omniauth(auth)

where(provider: auth.provider, uid: auth.uid).first\_or\_create do |user|

user.email = auth.info.email

user.password = Devise.friendly\_token[0,20]

end

end

end

<h2>Resend confirmation instructions</h2>

<%= form\_for(resource, as: resource\_name, url: confirmation\_path(resource\_name), html: { method: :post }) do |f| %>

<%= devise\_error\_messages! %>

<div class="field">

<%= f.label :email %><br />

<%= f.email\_field :email, autofocus: true, value: (resource.pending\_reconfirmation? ? resource.unconfirmed\_email : resource.email) %>

</div>

<div class="actions">

<%= f.submit "Resend confirmation instructions" %>

</div>

<% end %>

<%= render "devise/shared/links" %>

class List < ActiveRecord::Base

has\_many :tasks

has\_many :user\_lists

has\_many :users, through: :user\_lists

validates\_presence\_of :name, uniqueness: true

accepts\_nested\_attributes\_for :tasks

end

Rails.application.routes.draw do

root 'lists#index'

resources :lists do

resources :tasks

end

devise\_for :users, :controllers => { :omniauth\_callbacks => "users/omniauth\_callbacks" }

# The priority is based upon order of creation: first created -> highest priority.

# See how all your routes lay out with "rake routes".

# You can have the root of your site routed with "root"

# Example of regular route:

# get 'products/:id' => 'catalog#view'

# Example of named route that can be invoked with purchase\_url(id: product.id)

# get 'products/:id/purchase' => 'catalog#purchase', as: :purchase

# Example resource route (maps HTTP verbs to controller actions automatically):

# resources :products

# Example resource route with options:

# resources :products do

# member do

# get 'short'

# post 'toggle'

# end

#

# collection do

# get 'sold'

# end

# end

# Example resource route with sub-resources:

# resources :products do

# resources :comments, :sales

# resource :seller

# end

# Example resource route with more complex sub-resources:

# resources :products do

# resources :comments

# resources :sales do

# get 'recent', on: :collection

# end

# end

# Example resource route with concerns:

# concern :toggleable do

# post 'toggle'

# end

# resources :posts, concerns: :toggleable

# resources :photos, concerns: :toggleable

# Example resource route within a namespace:

# namespace :admin do

# # Directs /admin/products/\* to Admin::ProductsController

# # (app/controllers/admin/products\_controller.rb)

# resources :products

# end

end