

kittinder

minling && seth

some cats are lonely



cats need love, too

Kittinder

Cats

Owners

Log in with:



Cleopatra



Color: White

Gender: female

Description: You be the Dairy Queen and I'll be your Burger King:
You treat me right, and I'll do it your way.

Yes, Please

No, Thanks



demo

cool tech used

ruby



rails



html

HTML



css

CSS



omniauth



implementation

double submission && random cats

```
1 class MatchesController < ApplicationController
2 # before_action :set_user
3
4 def create
5   if params[:commit] == 'Yes'
6     @match = Match.new
7     @match.match_type = true
8     @match.actor_id = session[:cat_id]
9     @match.receiver_id = params[:receiver_id]
10    @match.save
11  elsif params[:commit] == 'No'
12    @match = Match.new
13    @match.match_type = false
14    @match.actor_id = session[:cat_id]
15    @match.receiver_id = params[:receiver_id]
16    @match.save
17  end
18
19  redirect_to root_path
20
21
```

```
1 <div class = "layout">
2   <%= form_tag(create_matches_path) do %>
3     <%= hidden_field_tag 'receiver_id', @cat.id %>
4
5     <%= submit_tag("Yes, Please", id: "button") %>
6     <%= submit_tag("No, Thanks", id: "button") %>
7
8   <%end %>
9 </div>
10
```

```
1 class StaticPagesController < ApplicationController
2
3   def home
4     @cat = Cat.all.sample(1).first
5   end
6
7 end
```

implementation

omniauth && aliases

```
1 class Cat < ActiveRecord::Base
2   belongs_to :owner
3   has_many :matches_as_actor, :class_name => "Match", :foreign_key => :actor_id
4   has_many :matches_as_receiver, :class_name => "Match", :foreign_key => :receiver_id
5
6   validates_presence_of :owner #no strays allowed
7
8
9   def self.find_or_create_from_auth_hash(auth_hash)
10    if Cat.find_by(:name => auth_hash[:info][:name])
11      Cat.find_by(:name => auth_hash[:info][:name])
12    else
13      create! do |cat|
14        cat.provider = auth_hash["provider"]
15        cat.uid = auth_hash["uid"]
16        cat.name = auth_hash["info"]["name"]
17        cat.owner = Owner.find(1)
18      end
19    end
20  end
```

```
1 Rails.application.config.middleware.use OmniAuth::Builder do
2   provider :github, ENV['GITHUB_KEY'], ENV['GITHUB_SECRET']
3 end
4
5 Rails.application.config.middleware.use OmniAuth::Builder do
6   provider :facebook, ENV['FACEBOOK_KEY'], ENV['FACEBOOK_SECRET']
7 end
8
9 Rails.application.config.middleware.use OmniAuth::Builder do
10 provider :twitter, ENV['TWITTER_KEY'], ENV['TWITTER_SECRET']
11 end
12
13 Rails.application.config.middleware.use OmniAuth::Builder do
14 provider :linkedin, ENV['LINKEDIN_KEY'], ENV['LINKEDIN_SECRET']
15 end
```

```
1 class SessionsController < ApplicationController
2
3   def create
4     @cat = Cat.find_or_create_from_auth_hash(auth_hash)
5     session[:cat_id] = @cat.id
6     redirect_to root_path
7   end
8
9   def destroy
10    session[:cat_id] = nil
11    redirect_to root_path, notice: "signed out, sucka!"
12  end
13
14  protected
15  def auth_hash
16    request.env['omniauth.auth']
17  end
18
19 end
```

```
29   create_table "matches", force: :cascade do |t|
30     t.integer "actor_id"
31     t.integer "receiver_id"
32     t.boolean "match_type"
33   end
```



moving forward

notification for matches

**edit profile page integrated with owner
page**

geo-selection

javascript/jquery to animate more activity



the end