



AUTHLOGIC AND AUTH_HELPERS

Presenter : Allen Walker

Email: allen@greathills.com

WHAT IS AUTHLOGIC GEM

- Pure authentication plugin
 - No generated code unlike Restful authentication plugin
 - Doesn't litter application with a ton of code
 - Simple and unobtrusive



AUTHLOGIC – SESSION LOGIN

- Define special user session model that inherits from `Authlogic::Session::Base` (instead of `ApplicationController`)

```
class UserSession < Authlogic::Session::Base
  def create
    @user_session = UserSession.new(params[:user_session])
    if @user_session.save
      flash[:notice] = "Successfully logged in."
      redirect_to root_url
    else
      render :action => 'new'
    end
  end
end
```



AUTHLOGIC – SESSION LOGOUT

```
def destroy
  @user_session = UserSession.find
  @user_session.destroy
  flash[:notice] = "Successfully logged out."
  redirect_to root_url
end
```



AUTHLOGIC – USER MODEL

- **User** model (linked to **UserSession** by name)

```
class User < ActiveRecord::Base
  acts_as_authentic
end
```



AUTHLOGIC – USER CONTROLLER

```
def create
  @user = User.new(params[:user])
  if @user.save
    flash[:notice] = "Registration successful."
    redirect_to root_url
  else
    render :action => 'new'
  end
end
```



AUTHLOGIC – USER MODEL

```
def update
  @user = current_user
  if @user.update_attributes(params[:user])
    flash[:notice] = "Successfully updated."
    redirect_to root_url
  else
    render :action => 'edit'
  end
end
```



AUTHLOGIC – HELPERS

- application_controller.rb

```
filter_parameter_logging :password  
helper_method :current_user
```

```
private
```

```
def current_user_session  
  return @current_user_session if defined?  
    (@current_user_session)  
  @current_user_session = UserSession.find  
end
```

```
def current_user  
  return @current_user if defined?(@current_user)  
  @current_user = current_user_session.record  
end
```



AUTH_HELPERS

- Collection of modules to improve your Authlogic models
- Exposes some convenience methods to help on confirmation and passwords management
- Deals automatically notifications (email)
- Makes use of the “**perishable_token**” magic field defined in your user Authlogic model



AUTH_HELPERS – INSTALL

- `gem sources -a http://gems.github.com`
- `gem install josevalim-inherited_resources`
- `gem install josevalim-auth_helpers`



AUTH_HELPERS – CONFIRMABLE MODULE

- Add migrations to User Model

```
t.datetime :confirmed_at
```

```
t.datetime :confirmation_sent_at
```



AUTH_HELPERS – MODEL

```
class User < ActiveRecord::Base
  acts_as_authentic do |a|
    a.require_password_confirmation = false
  end

  include AuthHelpers::Model::Confirmable
  include AuthHelpers::Model::Recoverable
  include AuthHelpers::Model::Updatable

  attr_accessible :username, :time_zone, :city, :state
end
```

- Authlogic already supports password confirmation
- Auth_Helpers overrides this logic so therefore must disable require_password_confirmation from Authlogic



AUTH_HELPERS – NOTIFICATION SETUP

- **environment.rb**

```
AuthHelpers::Notifier.sender =  
  "notifications@fadethepublic.com"  
AuthHelpers::Notifier.template_root =  
  "#{RAILS_ROOT}/app/views"  
AuthHelpers::Notifier.delivery_method = :smtp  
AuthHelpers::Notifier.raise_delivery_errors = true  
AuthHelpers::Notifier.smtp_settings = {  
  :address => "localhost",  
  :port => 25,  
  :domain => "fadethepublic.com",  
  :user_name => "mail@fadethepublic.com",  
  :password => "xxx",  
  :authentication => :login  
}
```



AUTH_HELPERS – VIEWS

- views/auth_helpers/notifier
 - create_confirmation.erb
 - resend_confirmation.erb
 - reset_password.erb
 - update_confirmation.erb

E.g. (create_confirmation.erb)

Your account has been created and you can confirm
your account at

```
<%= user_confirmation_url(@record.id, :user =>  
{ :perishable_token =>  
@record.perishable_token }) %>
```



AUTHHELPERS – ROUTING

```
map.resources :users do |user|
  user.resource :password, :only =>
    [ :new, :edit, :create, :update ], :controller =>
    'users/passwords'
  user.resource :confirmation, :only =>
    [ :new, :show, :create ], :controller => 'users/
    confirmations'
end
```



AUTH_HELPERS CONTROLLERS

- o **users/confirmations_controller.rb**

```
class Users::ConfirmationsController <  
  AuthHelpers::Controller::Confirmable  
end
```



MORE READINGS – QUESTIONS

- http://wiki.github.com/josevalim/auth_helpers
- <http://github.com/binarylogic/authlogic>

