

# Desarrollo de aplicaciones Web con

## Ruby

*“El mejor amigo del programador”*

On

## Rails

*“El desarrollo web que no molesta”*

Por: Ruben Aléxis Dávila Santos

# Ruby

**Creador:** Yukihiro “matz” Matsumoto

**Primera version:** 1995

Lenguaje de scripting verdaderamente orientado a objetos, dinámicamente tipado, muy flexible y con lo mejor de otros lenguajes como: Smalltalk, Perl, Python, Ada, Lisp, Eiffel.

Es open source como todas las buenas cosas de este mundo, y tiene una comunidad que crece de forma impresionante.

Tiene el soporte de empresas como Sun, que ha aportado con proyectos como JRuby y la maquina para el servidor de control se versiones (Subversion), actualmente bajo Solaris 10.

# Rails

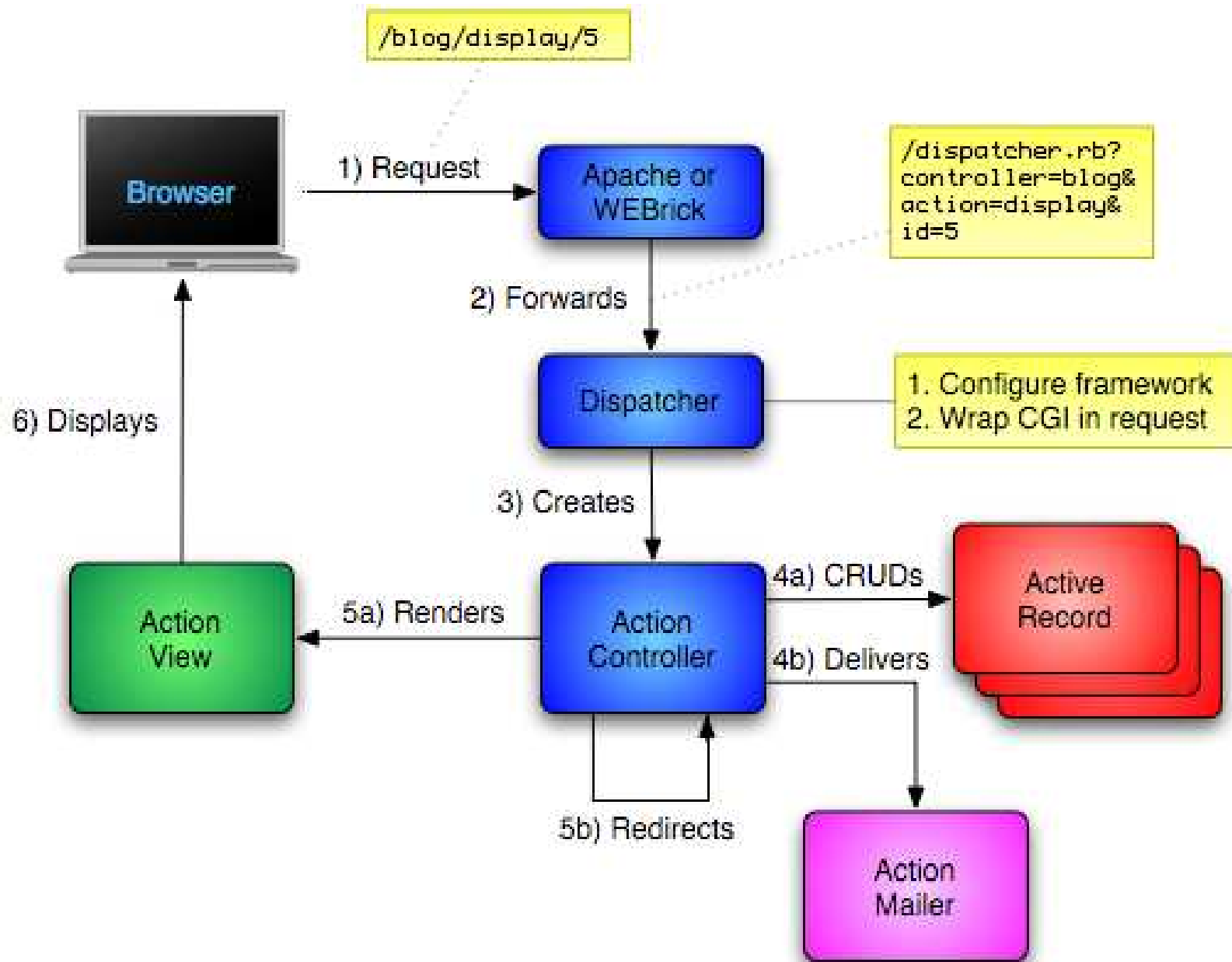
**Creador:** David Heinemeier Hansson

**Primera version:** 2004

Es un framework para el desarrollo de aplicaciones web, muy completo y escalable.

Su objetivo es tener desarrolladores ágiles, pragmáticos, felices y hacer que el desarrollo de aplicaciones web sea rápido y sencillo.

# MVC



# Configuracion de Base de datos

## config/database.yml

```
1 development:
2   adapter: mysql
3   database: tienda_desarrollo
4   username: root
5   password: secreto
6   host: 127.0.0.1
7
8 test:
9   adapter: mysql
10  database: tienda_test
11  username: root
12  password: secreto
13  host: 127.0.0.1
14
15 production:
16  adapter: mysql
17  database: tienda
18  username: root
19  password: secreto
20  host: 127.0.0.1
```

# Migraciones

```
1 class CreateClientes < ActiveRecord::Migration
2   def self.up
3     create_table :clientes do |t|
4       t.column :nombres, :string, :limit => 30
5       t.column :nombres, :string, :limit => 30
6       t.column :nombres, :string, :limit => 30
7     end
8   end
9
10  def self.down
11    drop_table :clientes
12  end
13 end
```

**rake db:migrate**

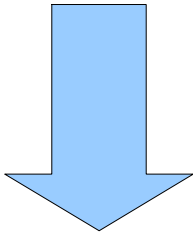


## Clientes

id	?
nombres	?
apellidos	?
direccion	?

# Modelo

```
1 class Cliente < ActiveRecord::Base |  
2 end
```



## Cientes

id

nombres

apellidos

direccion

```
#Creando un nuevo cliente  
cliente=Cliente.new  
cliente.nombres="Jose Carlos"  
cliente.apellidos="Garcia Toledo"  
cliente.direccion="Jr. El Dinero 568"  
cliente.save
```

# Controlador

```
1 class ClientesController < ApplicationController
2   def lista
3     @clientes = Cliente.find(:all)
4   end
5   def nuevo
6     @cliente = Cliente.new
7   end
8   def editar
9     @cliente = Cliente.find(params[:id])
10  end
11  def crear
12    @cliente = Cliente.new(params[:cliente])
13    if @cliente.save
14      redirect_to :action => :lista
15    else
16      render :action => :nuevo
17    end
18  end
end
```

<http://123.com/clientes/nuevo>

<http://123.com/clientes/editar/1>

Ingrediente que se encarga de la iteración con el usuario, de la comunicación con el modelo (Active Record), y de la respuesta al usuario.



# Vistas

## nuevo.rhtml

```
<%= form_tag :action=>:crear %>  
<% text_field :cliente,:nombres %>  
<% text_field :cliente,:apellidos %>  
<% text_field :cliente,:direccion %>  
<%= end_form_tag %>
```

## editar.rhtml

```
<%= form_tag :action=>:actualizar,:id=>@cliente %>  
<% text_field :cliente,:nombres %>  
<% text_field :cliente,:apellidos %>  
<% text_field :cliente,:direccion %>  
<%= end_form_tag %>
```

**DRY al rescate!!!**

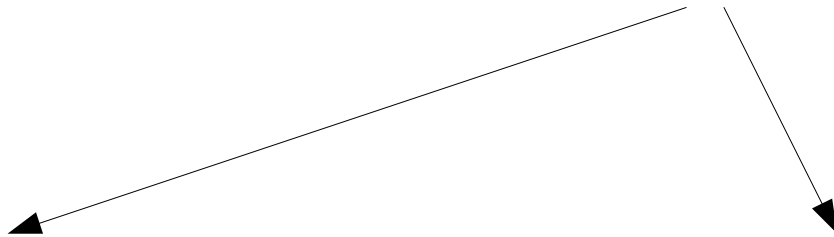
# Partials

## **\_formulario.rhtml**

<% **text\_field** :cliente,:nombres %>

<% **text\_field** :cliente,:apellidos %>

<% **text\_field** :cliente,:direccion %>



## **nuevo.rhtml**

<%= **form\_tag** :action=>:nuevo %>

<%= **render** :partial=>'formulario' %>

<%= **end\_form\_tag** %>

## **editar.rhtml**

<%= **form\_tag** :action=>:actualizar,:id=>@cliente %>

<%= **render** :partial=>'formulario' %>

<%= **end\_form\_tag** %>

# Validaciones

```
1 class Cliente < ActiveRecord::Base
2   validates_presence_of :direccion, :message => 'esta vacia.', :on => :create
3   validates_uniqueness_of :email, :message => "ya existe"
4   validates_format_of :email, :with => /^[^@\s]+@((?:[-a-z0-9]+\.)+[a-z]{2,})$/i
5 end
```

# Relaciones

```
1 class Cliente < ActiveRecord::Base
2   has_many :telefonos
3 end
```

```
1 class Telefono < ActiveRecord::Base
2   belongs_to :cliente
3 end
```

```
>> cliente=Cliente.find(1)
=> #<Cliente:0xb74401d0 @attributes={"nombres"=>"Juan", "id"=>"1",
"direccion"=>nil, "apellidos"=>"Perez"}>
>>
?> cliente.telefonos.each{ |c|
?> puts c.numero
>> }
5632565
3421453
```

# Generadores

## Modelos

```
ruben@bmw:~/ror_proyectos/ejemplo$ ./script/generate model cliente  
exists app/models/  
exists test/unit/  
exists test/fixtures/  
create app/models/cliente.rb  
create test/unit/cliente test.rb  
create test/fixtures/clientes.yml  
create db/migrate  
create db/migrate/001_create_clientes.rb
```

## Controladores

```
ruben@bmw:~/ror_proyectos/ejemplo$ ./script/generate controller clientes  
exists app/controllers/  
exists app/helpers/  
create app/views/clientes  
exists test/functional/  
create app/controllers/clientes controller.rb  
create test/functional/clientes controller test.rb  
create app/helpers/clientes helper.rb
```

# Mapeo de URL

## config/routes.rb

```
1 ActionController::Routing::Routes.draw do |map|
2   map.connect ':controller/service.wSDL', :action => 'wsdl'
3   # Install the default route as the lowest priority.
4   map.connect ':controller/:action/:id.:format'
5   map.connect ':controller/:action/:id'
6
7   #Mapeo para pagina principal
8   map.connect '', :controller=>'home',:action=>'index'
9
10  map.connect 'clientes',:controller=>'clientes',:action=>'list'
11 end
```

**http://123.com/**

**http://123.com/clientes**

# Antes

**find**      **create**      **update**      **destroy**  
**SELECT INSERT UPDATE DELETE**

```
>> Cliente.create({:nombres=>'Ruben',:apellidos=>'Davila Santos',:email=>'ruben@tiperu.com'})  
=> #<Cliente:0xb74bae08 @new_record=false, @errors=#<ActiveRecord::Errors:0xb74b9e68 @errors={}, @base=#<Cliente:0xb74bae08 ...>, @attributes={"nombres"=>"Ruben", "id"=>5, "email"=>"ruben@tiperu.com", "apellidos"=>"Davila Santos"}>
```

```
?> c=Cliente.find(5)  
=> #<Cliente:0xb74b5660 @attributes={"nombres"=>"Ruben", "id"=>"5", "email"=>"ruben@tiperu.com", "apellidos"=>"Davila Santos"}>  
>> c.email  
=> "ruben@tiperu.com"
```

```
?> Cliente.update(5,{:nombres=>'Ruben Alexis'})  
=> #<Cliente:0xb74af6e8 @errors=#<ActiveRecord::Errors:0xb74aec48 @errors={}, @base=#<Cliente:0xb74af6e8 ...>, @attributes={"nombres"=>"Ruben Alexis", "id"=>"5", "email"=>"ruben@tiperu.com", "apellidos"=>"Davila Santos"}>
```

```
?> Cliente.destroy(5)
=> #<Cliente:0xb74a4fe0 @attributes={"nombres"=>"Ruben Alexis", "id"=>"5", "email"=>"ruben@tiperu.com", "apellidos"=>"Davila Santos"}>
>>
?> c=Cliente.find(5)
ActiveRecord::RecordNotFound: Couldn't find Cliente with ID=5
```

**POST** <http://123.com/clientes/crear>

**GET** <http://123.com/clientes/ver/1>

**POST** <http://123.com/clientes/actualizar/1>

**POST** <http://123.com/clientes/eliminar/1>

# Ahora

<b>GET</b>	<b>POST</b>	<b>PUT</b>	<b>DELETE</b>
<b>find</b>	<b>create</b>	<b>update</b>	<b>destroy</b>
<b>SELECT</b>	<b>INSERT</b>	<b>UPDATE</b>	<b>DELETE</b>

**POST** <http://123.com/clientes>

**GET** <http://123.com/clientes/1>

**PUT** <http://123.com/clientes/1>

**DELETE** <http://123.com/clientes/1>



```
1 class ClientesController < ApplicationController
2
3   def index
4     @clientes = Cliente.find(:all)
5
6     respond_to do |format|
7       format.html # index.rhtml
8       format.js #index.rjs
9       format.xml { render :xml => @clientes.to_xml }
10    end
11
12  end
```

```
ruben@bmw:~$ curl -i http://localhost:3000/clientes
HTTP/1.1 200 OK
Connection: close
Date: Mon, 19 Mar 2007 03:19:20 GMT
Set-Cookie: _service_session_id=866495701dd94ac95ab1317a13db3e4b; path=/
Status: 200 OK
Cache-Control: no-cache
Server: Mongrel 1.0.1
Content-Type: text/html; charset=utf-8
Content-Length: 1113

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
  <title>Clientes: index</title>
  <link href="/stylesheets/scaffold.css?1174266170" media="screen" rel="Stylesheet" type="text/css" />
</head>
<body>

<p style="color: green;"></p>

<h1>Listing clientes</h1>

<table>
  <tr>
  </tr>

  <tr>
    <td><a href="/clientes/2">Show</a></td>
    <td><a href="/clientes/2;edit">Edit</a></td>
  </tr>
</table>
```

```
ruben@bmw:~$ curl -i http://localhost:3000/clientes.xml  
HTTP/1.1 200 OK  
Connection: close  
Date: Mon, 19 Mar 2007 03:25:38 GMT  
Set-Cookie: _service_session_id=90a072b6833d1c5966e3e3488160c092; path=/  
Status: 200 OK  
Cache-Control: no-cache  
Server: Mongrel 1.0.1  
Content-Type: application/xml; charset=utf-8  
Content-Length: 216
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<clientes>  
  <cliente>  
    <apellidos>Davila</apellidos>  
    <email>ruru@hotmail.com</email>  
    <id type="integer">2</id>  
    <nombres>Ruben</nombres>  
  </cliente>  
</clientes>  
ruben@bmw:~$ █
```

```
1 require 'net/http'
2 Net::HTTP.start('localhost','3000') do |http|
3   response=http.get('/clientes.xml','Accept' => 'text/xml')
4
5   #hacer algo con la respuesta
6
7   puts "Codigo: #{response.code}"
8   puts "Mensaje: #{response.message}"
9   puts "Cuerpo:\n #{response.body}"
10
11 end
```

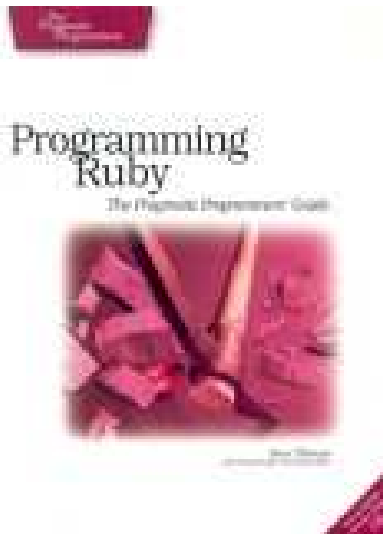
---

```
ruben@bmw:~$ ruby clientes.rb
Codigo: 200
Mensaje: OK
Cuerpo:
<?xml version="1.0" encoding="UTF-8"?>
<clientes>
  <cliente>
    <apellidos>Davila</apellidos>
    <email>ruru@hotmail.com</email>
    <id type="integer">2</id>
    <nombres>Ruben</nombres>
  </cliente>
</clientes>
ruben@bmw:~$ █
```

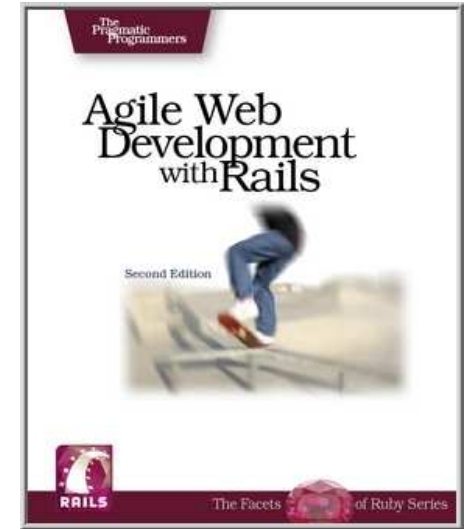
# Tu Decides ;)



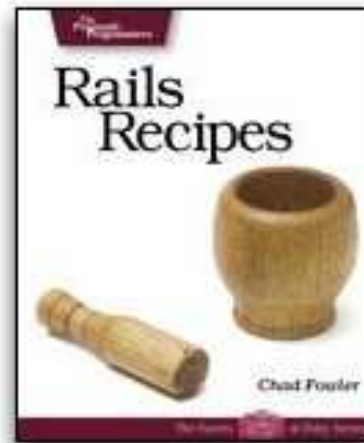
# Libros Recomendados



**Programing Ruby**



**Agile Web development  
with Ruby on Rails**



**Rails Recipes**

# Gracias



<http://ruby-lang.org>



<http://rubyonrails.org>

**Ruben Alexis Davila Santos**

[ruben@tiperu.com](mailto:ruben@tiperu.com)