



Catalyst: Framework para el desarrollo de aplicaciones Web

Esteban Saavedra Lopez

email: estebansaavedra@yahoo.com

<http://jesaavedra.opentelematics.org>

Agenda

- Introducción
- Patron MVC
- Catalyst
- Demostración
- Conclusiones

Que es un Framework

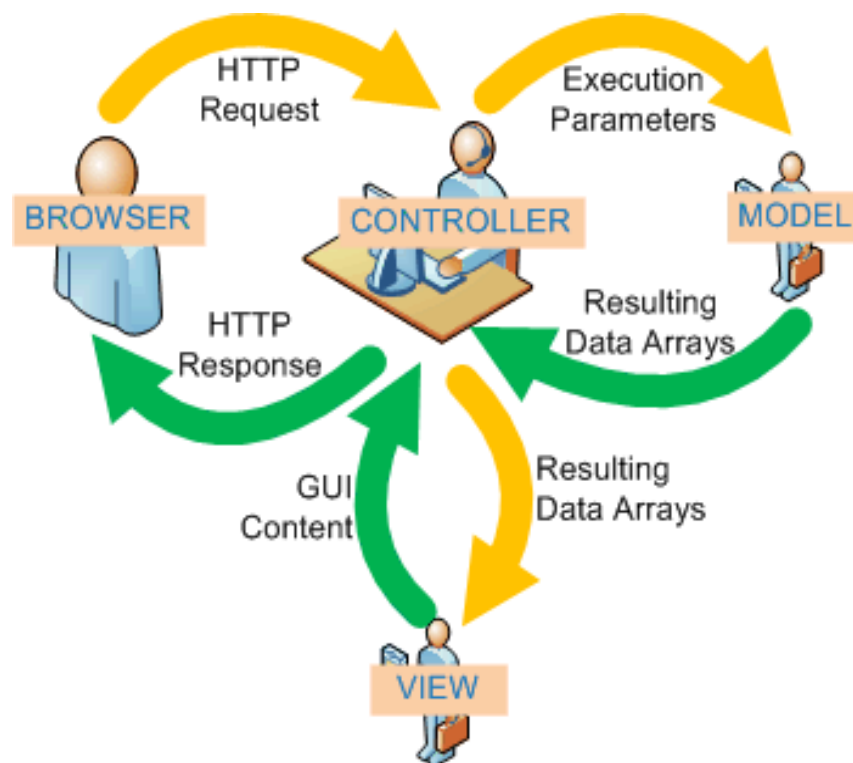
- Es una **estructura** de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado.
- Se han convertido en la piedra angular de la moderna ingeniería del software
- Es una estructura software compuesta de **componentes personalizables** e intercambiables para el desarrollo de una aplicación.

Cual el objetivo de un framework?

- Desarrollo **rápido** y **estructurado**
- **Reutilización** de código
- **Disminuir el esfuerzo** en el desarrollo (Nos concentramos directamente en la solución del problema)
- Aprovechamos las **funcionalidades** ya **implementadas**.
- Tener como aliado a las metodología de desarrollo Ágiles (XP, Scrum, AD,...)

Patrón MVC

- Modelo
- Vista
- Controlador



Que es Catalyst?

- Es un framework para el desarrollo de aplicaciones web.
- Es la respuesta de Perl para Ruby on Rails.
- Permite el desarrollo de aplicaciones web de forma rápida.

Filosofía de Catalayst

- Reutilización de código.
- Desarrollar aplicaciones web de forma rápida, **sencilla y divertida**.
- Utilizar los **mejores componentes** actualmente disponibles.
- Combinar todo en un paquete fácil de usar.

Por que Catalyst?

- Se basa en el patrón MVC.
- Tiene todo lo necesario para desarrollar aplicaciones Web.
- Soporta Ajax.
- Posee buena documentación.
- Es multiplataforma.
- Posee una numerosa comunidad de desarrolladores y usuarios
- Inspirado en Ruby on Rails (Padrino de la mayoría de los Frameworks)

Componentes de Catalyst

- Todos sus componentes se encuentran en CPAN

Instalación de Catalyst

- Debian, Ubuntu, ...
 - > `$ apt-get install libcatalyst-perl`
- CPAN
 - > `$ cpan Catalyst::Runtime Catalyst::Devel`

Estructura de un proyecto de Catalyst

- Catalyst proporciona una estructura jerárquica:
 - > Proyecto
 - > Aplicación
 - > Módulo
 - > Acción
 - > Vista

Creación de un proyecto

- Creamos proyecto
 - > `$ catalyst.pl proyecto`
- Ingresamos al directorio del proyecto.
 - > `$ cd proyecto`
- Ejecutamos el servidor de pruebas
 - > `$ perl script/proyecto_server.pl`

Que generó esto?

- Una estructura jerárquica de directorios

```
.
|-- Changes
|-- Makefile.PL
|-- README
|-- lib
|   |-- proyecto
|   |   |-- Controller
|   |   |   |-- Root.pm
|   |   |-- Model
|   |   |-- View
|   |-- proyecto.pm
|-- proyecto.yml
|-- root
|   |-- favicon.ico
|   |-- static
|   |   |-- images
|   |   |   |-- btn_120x50_built.png
|   |   |   |-- btn_120x50_built_shadow.png
|   |   |   |-- btn_120x50_powered.png
|   |   |   |-- btn_120x50_powered_shadow.png
|   |   |   |-- btn_88x31_built.png
|   |   |   |-- btn_88x31_built_shadow.png
|   |   |   |-- btn_88x31_powered.png
|   |   |   |-- btn_88x31_powered_shadow.png
|   |   |   |-- catalyst_logo.png
|-- script
|   |-- proyecto_cgi.pl
|   |-- proyecto_create.pl
|   |-- proyecto_fastcgi.pl
|   |-- proyecto_server.pl
|   |-- proyecto_test.pl
|-- t
|   |-- 01app.t
|   |-- 02pod.t
|   |-- 03podcoverage.t
```

Pasos para crear un proyecto

- 1 Crear el proyecto
- 2 Crear los dominios del proyecto
- 3 Generar los CRUDs
- 4 Adecuar los templates (CSS y/o JavaScript)
- 5 Publicar la Aplicación

Adecuaciones

- Una vez generada gran parte del proyecto (forma muy general), se puede proceder a realizar adecuaciones en:
 - > Controlador (actions)
 - > Vistas (templates)
 - > Formas de acceso en las URL (routing)
 - > Validación de formularios
 - > Efectos especiales



Demostración

Crear un proyecto

- `/usr/bin/catalyst.pl` Gestion

Crear una vista genérica

- `perl script/gestion_create.pl view HTML TTSite`

Crear la BD

```
CREATE TABLE expositor(id INTEGER NOT NULL PRIMARY KEY,paterno VARCHAR(50),  
materno VARCHAR(50), nombre VARCHAR(50),  
pais VARCHAR(20), email VARCHAR(50) );
```

```
CREATE TABLE conferencia (id INTEGER not null PRIMARY KEY,  
expositor_id INTEGER NOT NULL,  
|titulo VARCHAR(50), resumen TEXT);
```

Insertamos datos

```
INSERT INTO expositor VALUES (1, 'Saavedra', 'Lopez', 'Esteban', 'Bolivia', 'esteban@gmail.com');  
INSERT INTO expositor VALUES (2, 'Saavedra', 'Ayarde', 'Kathleen', 'Mexico', 'kathleen@gmail.com');  
INSERT INTO expositor VALUES (3, 'Saavedra', 'Ayarde', 'Stephanie', 'Brasil', 'stephanie@gmail.com');  
INSERT INTO expositor VALUES (4, 'Saavedra', 'Ayarde', 'Scarlet', 'España', 'scarlet@gmail.com');  
INSERT INTO expositor VALUES (5, 'Saavedra', 'Lopez', 'Jenny', 'Venezuela', 'jenny@gmail.com');  
INSERT INTO expositor VALUES (6, 'Saavedra', 'Lopez', 'Jeanneth', 'Colombia', 'jeanneth@gmail.com');
```

```
INSERT INTO conferencia VALUES (1,1, 'Primera Conferencia', 'Primer Resumen');  
INSERT INTO conferencia VALUES (2,1, 'Segunda Conferencia', 'Segundo Resumen');  
INSERT INTO conferencia VALUES (3,2, 'Tercer Conferencia', 'Tercero Resumen');  
INSERT INTO conferencia VALUES (4,3, 'Cuarta Conferencia', 'Cuarto Resumen');  
INSERT INTO conferencia VALUES (5,4, 'Quinta Conferencia', 'Quinto Resumen');  
INSERT INTO conferencia VALUES (6,5, 'Sexta Conferencia', 'Sexto Resumen');  
INSERT INTO conferencia VALUES (7,6, 'Septima Conferencia', 'Septimo Resumen');  
INSERT INTO conferencia VALUES (8,6, 'Octava Conferencia', 'Octavo Resumen');  
INSERT INTO conferencia VALUES (9,1, 'Novena Conferencia', 'Noveno Resumen');
```

Creamos el modelo

- ```
perl script/gestion_create.pl model
GestionDB DBIC::Schema
Gestion::Schema::GestionDB create=static
dbi:SQLite:gestion
```

# Incluir las relaciones

- En Expositor

- (lib/Gestion/Schema/GestionDB/Expositor.pm)

```
__PACKAGE__->has_many(
 conferencia => 'Gestion::Schema::GestionDB::Conferencia',
 'expositor_id',{cascading_delete => 1});
```

- En conferencia

- (lib/Gestion/Schema/GestionDB/Conferencia.pm)

```
__PACKAGE__->belongs_to(
 expositor_id => 'Gestion::Schema::GestionDB::Expositor');
```

# Incluir métodos

- En Expositor

```
sub nombrecompleto {
```

```
 my $self = shift;
```

```
 return $self->paterno. ' '. $self->nombre;
```

```
}
```

# Manejo de plantillas

- en
- `/lib/Gestion/View/HTML`
- adicionar
- `TEMPLATE_EXTENSION => '.tt2'`,



# Crear una nueva pagina de inicio

- `/root/src/index.tt2`

```
[% META title = "Bienvenidos a gestion de eventos" %]
<p>Algunas funciones iniciales de nuestro sistema.</p>
<p>Opciones:

Listado de expositores

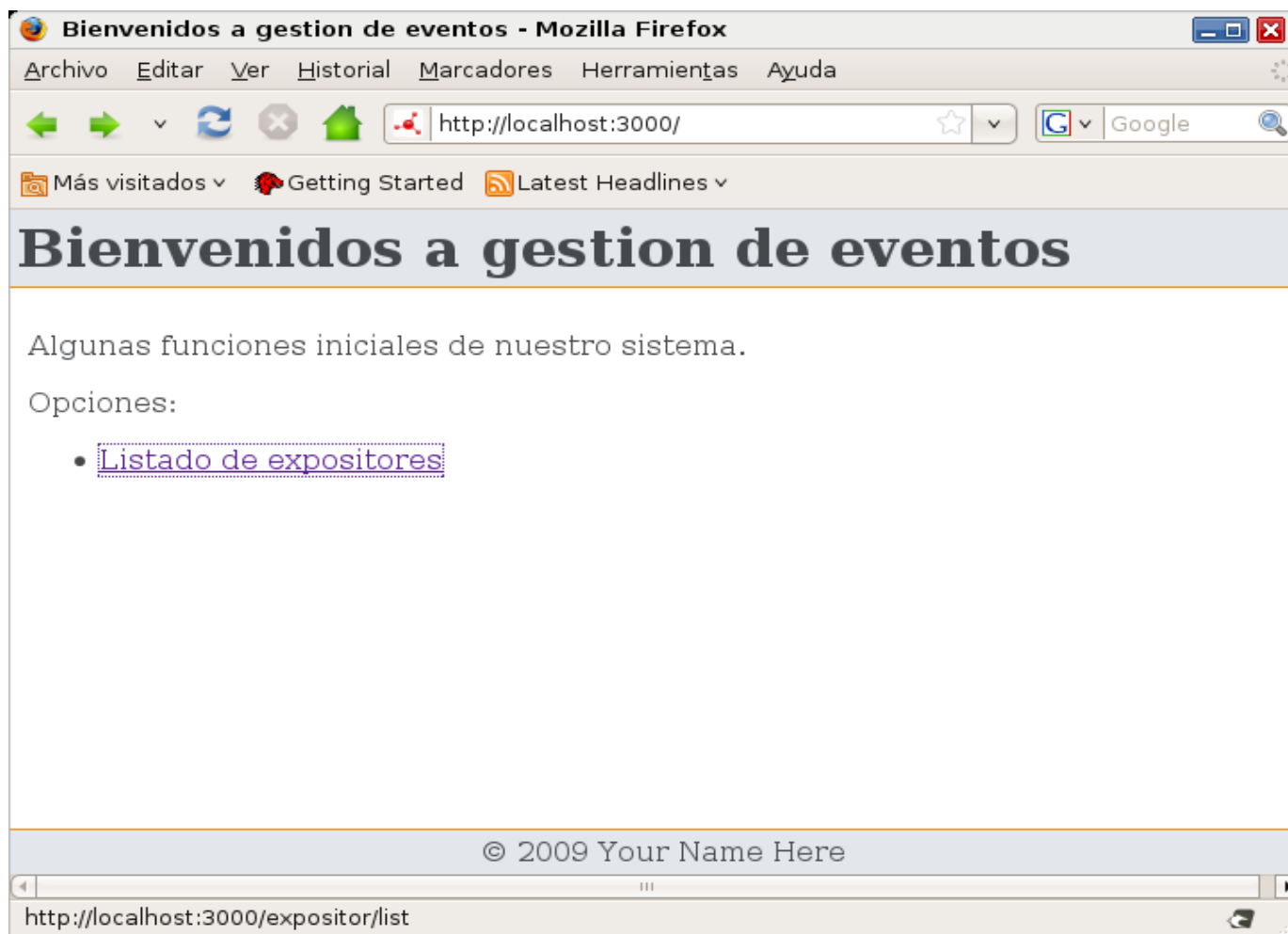
</p>
```

# Hacer que actue la pagina de inicio

- lib/Gestion/Controller/Root.pm

```
package Gestion::Controller::Root;
use strict;
use warnings;
use base 'Catalyst::Controller';
Sets the actions in this controller to be registered with no prefix
so they function identically to actions created in MyApp.pm
__PACKAGE__->config->{namespace} = '';
=head1 NAME
Gestion::Controller::Root - Root Controller for Gestion
=head1 DESCRIPTION
[enter your description here]
=head1 METHODS
=cut
=head2 default
=cut
sub default : Private {
 my ($self, $c) = @_;
 $c->response->status('404');
 $c->stash->{template} = 'not_found.tt2';
}
sub index : Private {};
=head2 end
Attempt to render a view, if needed.
=cut
sub end : ActionClass('RenderView') {}
=head1 AUTHOR
Esteban Saavedra Lopez,,,
=head1 LICENSE
This library is free software, you can redistribute it and/or modify
it under the same terms as Perl itself.
=cut
```

# Apariencia de la nueva página de inicio



# Crear una acción listar Expositores

- lib/Gestion/Controller/Expositor.pm

```
package Gestion::Controller::Expositor;
use strict;
use warnings;
use base 'Catalyst::Controller';
#use base qw(Catalyst::Controller::FormBuilder);
=head1 NAME
Gestion::Controller::Expositor - Catalyst Controller
=head1 DESCRIPTION
Catalyst Controller.
=head1 METHODS
=cut
=head2 list
Recupera todos los expositores y los envia a la platilla expositor/list.tt2 dentro del stash para ser desplegado
=cut

sub list : Local {
 my ($self, $c) = @_;
 #my $expositores : Stashed = $c->model('GestionDB::Expositor');
 $c->stash->{expositores} = [$c->model('GestionDB::Expositor')->all];
 $c->stash->{template} = 'expositor/list.tt2';
}
=head1 AUTHOR
Esteban Saavedra Lopez,,,
=head1 LICENSE
This library is free software, you can redistribute it and/or modify
it under the same terms as Perl itself.
=cut
1;
```

# Listado de Expositores y conferencias

root/src/expositor/list.tt2

```
[% META title = 'Listado de Expositores' -%]
<p>Listado de expositores:</p>

[% FOREACH expositor IN expositores -%]

 [% expositor.nombrecompleto | html %]
 Add address
 Edit
 Delete
[% SET conferencias = expositor.conferencia %]

 [% FOREACH conferencia = conferencias %]

 [% conferencia.titulo | html %]
 Edit
 Delete

 <address>
 Resumen: [% conferencia.resumen | html %]

 </address>

 [% END %]

[% END %]

```

# Listado de Expositores y Conferencias



The screenshot shows a Mozilla Firefox browser window with the title "Listado de Expositores - Mozilla Firefox". The address bar displays "http://localhost:3000/expositor/list". The page content includes a main heading "Listado de Expositores" and a sub-heading "Listado de expositores:". Below this, there is a list of exhibitors, each with a set of links for "Add address", "Edit", and "Delete", followed by a list of conferences and their respective summaries.

© 2009 Your Name Here

Listo

# Ejemplos Desarrollados

# Aplicaciones de ejemplo (1)

**Reportes de Clientes**  
[Estado de Cuentas General](#)  
[Estado de Cuentas Especifico](#)

**Reportes de compras**  
[Compras por Fecha](#)  
[Reporte Clientes](#)

**Reportes de Pagos**  
[Pagos por Fechas](#)  
[Pagos Empresas](#)

**Todo LIBROS**

## Listado de Libros

	ID	1	
	Precio	10	
	Nombre	El Perfume	
	Editorial	editorial01	
	Descripcion		
	ID	2	
	Precio	20	
	Nombre	Pensamientos	
	Editorial	editorial02	
	Descripcion		
	ID	4	
	Precio	44	
	Nombre	Picasso	
	Editorial	editorial04	
	Descripcion		
	ID	5	
	Precio	33	
	Nombre	Diseño y decoración	
	Editorial	editorial03	
	Descripcion		
	ID	6	
	Precio	555	
	Nombre	Lecciones de Vida	
	Editorial	editorial05	
	Descripcion		

Listo



# Aplicaciones de ejemplo (2)

The screenshot shows a Mozilla Firefox browser window titled "Venta de Libros - Mozilla Firefox". The address bar displays "http://localhost:3000/clientes/estado". The page content includes a navigation menu with "Libros", "Clientes", "Compras", "Pagos", "Acerca de", and "Contacto". A sidebar on the left lists various reports under "Reportes de Clientes", "Reportes de compras", and "Reportes de Pagos". The main content area is titled "Formulario de estado de Cuentas" and contains the following form fields:

- Cliente:** saavedra lopez jose
- Fecha Inicial:** 2006-09-26
- Fecha Final:** 2007-10-10

Below the form is a button labeled "Obtener Reporte". A date selection calendar is open, showing the month of October 2007. The calendar table is as follows:

October, 2007							
Today							
wk	Sun	Mon	Tue	Wed	Thu	Fri	Sat
39		1	2	3	4	5	6
40	7	8	9	10	11	12	13
41	14	15	16	17	18	19	20
42	21	22	23	24	25	26	27
43	28	29	30	31			

The status bar at the bottom of the browser window shows "Listo".

# Frameworks de desarrollo



**django**



GRAILS



# Conclusiones

- Tiene todo lo necesario para desarrollar aplicaciones Web
- Facilitan enormemente el desarrollo de aplicaciones
- Posee buena documentación
- Esta en constante desarrollo (muchas funcionalidades en base plugins)
- Al tener como aliado a las metodologías de desarrollo Ágiles(XP,AD,...), se convierte en un fuerte candidato para desarrollo de aplicaciones web corporativas.

# Consultas

