

Desarrollo Rápido de Aplicaciones Web con Ruby y Rails

Edgardo Hames



Agenda

- Presentación del patrón MVC
- Un pantallazo de Ruby
- Introducción a Rails
- Un caso real

Problema frecuente (1)

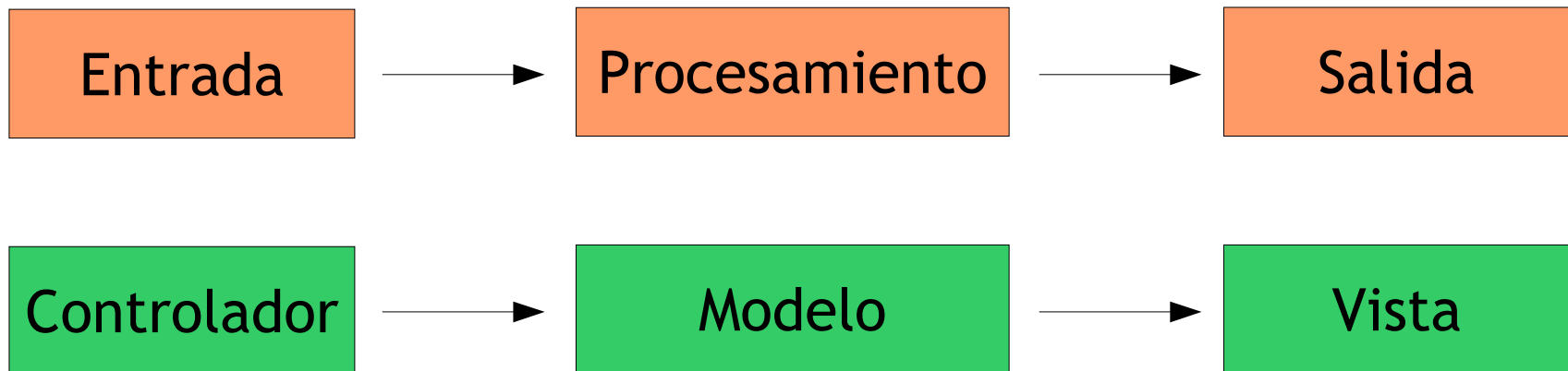
- Las aplicaciones contienen una mezcla de código de acceso a datos, lógica del dominio y código de presentación
 - Una base de datos
 - Las operaciones permitidas sobre los distintos objetos que han sido modelados (personas, productos, etc)
 - Una interfaz con el usuario

Problema Frecuente (2)

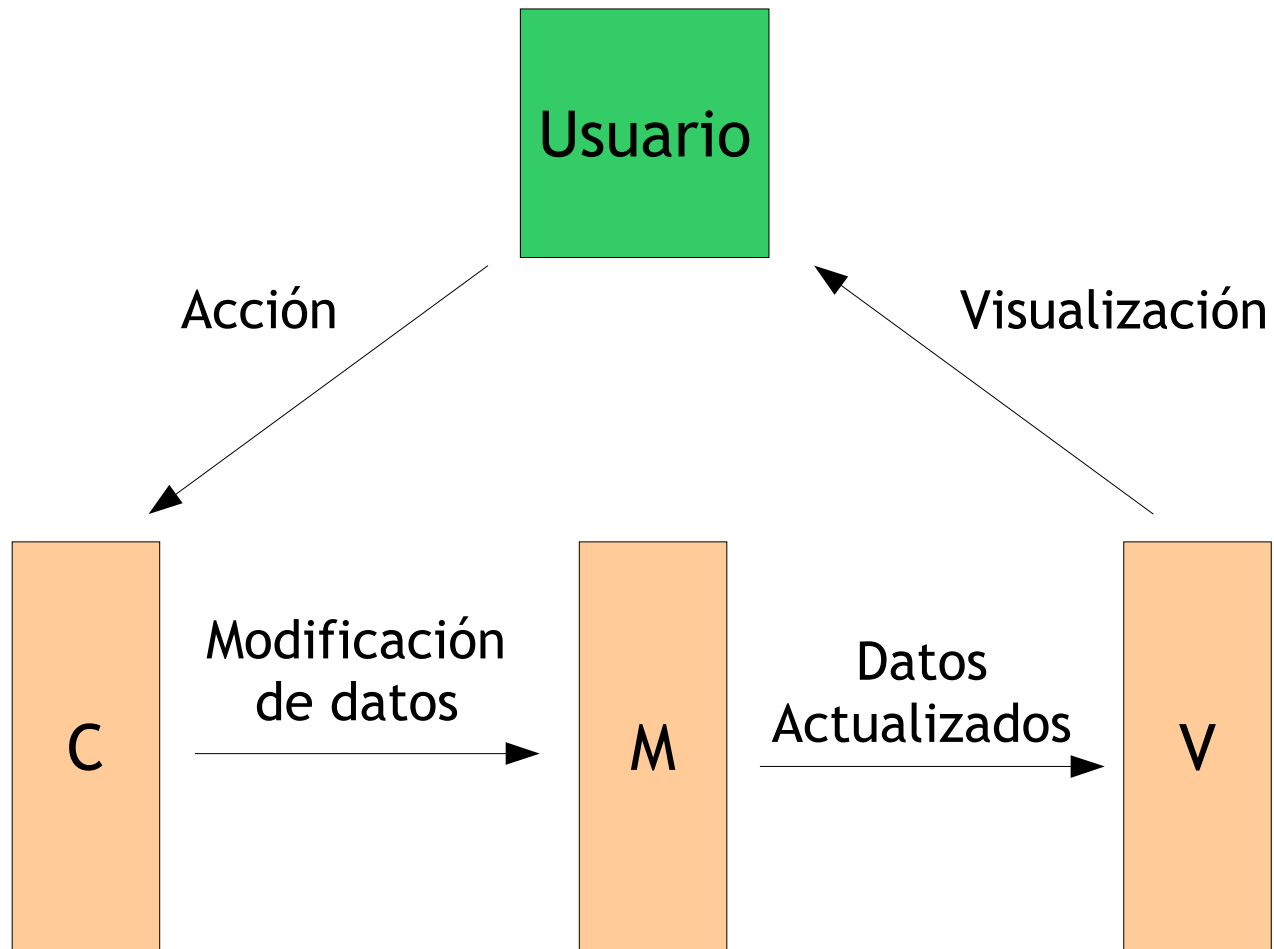
- Alto acoplamiento entre clases o módulos dificulta el mantenimiento y el reuso porque hay una gran inter-dependencia
- Agregar nuevas entidades requiere reimplementar o copiar y pegar lógica de negocio
- Hay que mantener el mismo código en varios lugares y aumenta la inconsistencia

MVC – Modelo Vista Controlador

- Una forma de descomponer una aplicación para manejar potentemente su interfaz reduciendo el acoplamiento entre módulos



Arquitectura MVC



Modelo

- Cómo se organizan los datos
 - Estructura
 - Integridad
 - Manipulación
 - Consulta
- La lógica del modelo de un negocio
 - Flujo de información en una empresa
 - Validaciones

Controlador

- Acepta los comandos del usuario e instruye al modelo y a la vista que realicen distintas acciones:
 - Mapea las acciones del usuario a las respuestas de la aplicación
 - Determina el comportamiento de la aplicación

Vista

- Provee el mecanismo de interacción entre el usuario y el programa
 - Entrada: permite al usuario controlar un sistema
 - Salida: permite al sistema notificar al usuario

Ruby

El mejor amigo del programador

- Lenguaje orientado a objetos creado por Yukihiro Matsumoto en Japón
 - Manejo de excepciones
 - Herencia simple
 - Módulos y Mix-ins
 - Clausuras y Continuaciones
 - Bloques anónimos
 - Administración de memoria dinámica
 - Soporte para hilos en el intérprete
 - Fácil extensión de sus bibliotecas en C

Hola mundo

```
#!/usr/bin/ruby
```

```
# Este es el primer programa que  
# hacemos en Ruby!
```

```
puts "Hola mundo!"
```

Una función

```
#!/usr/bin/ruby
```

```
def saludar(persona)  
  puts "Hola #{persona}"  
end
```

```
saludar("Pedro")  
saludar "Julio"
```

Ruby: todo es un objeto

```
#!/usr/bin/ruby
```

```
puts 5.class
```

```
puts 5 + 2
```

```
puts 5.+(2)
```

```
puts "una cadena".length
```

```
puts "una cadena".class
```

```
puts ["Caro", "Pedro", "Julio"].sort.join(",")
```

```
puts Float.class
```

Declaración de Clases

```
class Persona
  attr_accessor :nombre
  def initialize(nombre)
    @nombre = nombre
  end
end

p = Persona.new "Matz"
puts p.nombre
```

Sintaxis

- Variables y métodos con minúsculas
`i, tmp, calcular_pi, edad`
- Constantes con mayúsculas
`PI`
- Variables de instancia empiezan con `@`
`@nombre, @edad, @fecha_nacimiento`
- Variables de clase empiezan con `@@`
`@@instancia, @@tabla_simbolos`
- Variables globales empiezan con `$`
- Símbolos empiezan con `:`
`:nombre, :edad`

Rails

MVC

+

Ruby

Rails

¿Qué es Rails?

- Rails es un entorno de desarrollo escrito en Ruby pensado para aplicaciones web que brinda una implementación completa de las capas del patrón MVC.
 - ActiveRecord: Modelo
 - ActionPack: Controlador y Vista
 - ActionMailer: acciones de control que implican notificaciones por correo electrónico

DRY

- El principio detrás de Ruby y Rails:

DON'T REPEAT YOURSELF

- El código va en un único lugar lo cual facilita el mantenimiento y el agregado de funcionalidades.

ActiveRecord

- Conecta objetos de negocio y tablas de la base de datos para crear un modelo del dominio persistente donde la lógica y los datos están envueltos en una sola capa.
- Provee un lenguaje específico del dominio que describe asociaciones y herencia entre tablas.
- Disminuye la cantidad de código que debe escribir un programador reduciendo la cantidad de errores en las aplicaciones.

ActiveRecord – Un ejemplo (1)

- Creemos una tabla para almacenar productos

```
CREATE TABLE PRODUCTS (  
    id INTEGER NOT NULL,  
    name VARCHAR (30) NOT NULL,  
    price INTEGER NOT NULL  
);
```

ActiveRecord – Un ejemplo (2)

- La clase que mapea la tabla de productos es

```
class Product < ActiveRecord::Base  
end
```

- Rails genera esa clase automáticamente y los atributos (*name*, *price*) se obtienen dinámicamente a partir de las columnas en la tabla.
- Notar que si alteramos la tabla no necesitamos modificar la clase Product: DRY!

ActiveRecord – DSL

- Algunos nombres de los métodos que nos provee AR:
 - `find`, `find_first`, `find_all`
 - `belongs_to`, `has_many`
 - `validates_presence_of`,
`validates_uniqueness_of`,
`validates_confirmation_of`
- Ejemplo
 - `Product.find(:condition => "price = 10")`
 - `validate_confirmation_of :password, :email`

ActiveRecord – Características

- Mapeo automático de clases con tablas y atributos con columnas
- Asociaciones y agregaciones con simples macros
- Reglas de validación
- Observadores durante toda el ciclo de vida de un objeto de negocio
- Transacciones y bitácora de consultas
- Adaptadores sencillos para nuevas bases de datos

ActiveRecord – Filosofía

- Configuración
 - Ningún archivo XML de configuración
 - Mucha reflexión y extensión en tiempo de ejecución
 - *Magia* no es una mala palabra en si misma
- Base de datos
 - Se puede usar SQL por performance o casos raros
 - No intenta duplicar o reemplazar las definiciones de los datos

ActionPack y ActionView

- ActionPack
 - Rutea los pedidos recibidos a través de controladores con un método por cada acción
- ActionView
 - Permite que la vista sea mostrada a través de plantillas de Ruby
 - Las páginas rhtml contienen etiquetas que encierran código Ruby
 - Hay métodos que permiten generar HTML muy fácilmente

ActionMailer

- Herramienta para diseñar capas que requieren el envío de correo electrónico
 - Contraseñas olvidadas
 - Bienvenida a grupos
 - Notificación de asignaciones o acciones pendientes
 - Envío de facturas a un cliente
- También se puede utilizar para procesar correo electrónico entrante
 - Aceptar publicaciones en blogs a través de email

Otras características de Rails

- Soporte para
 - AJAX
 - Depuración con breakpoints
 - WebServices, SOAP, XML-RPC
 - Caché de acciones, páginas o fragmentos
 - Cookies
 - Paginación
 - Verificación
 - Testing

Una aplicación real

- Una aplicación que nos permita asignar actividades a personas dentro de una organización
 - ABM de tareas
 - ABM de personas
 - Asignación de tareas a personas

Referencias

- <http://www.ruby-lang.org>
- <http://www.rubyonrails.com>

¿Preguntas?