

¿números romanos ...

MVC MVVM MVP

... o qué?

Guido Marucci Blas

@guidomb



#JSConfAR



zauber

Software Boutique & Labs



20%

JavaScript

.1 Javascript

.2 Ruby

.3 Python

underscore

PhantomJS

Jasmine

Kendo

jQuery

express

socket.io

JavaScript Momentum

Node

Backbone

three.js

d3

Bootstrap

Ember

raphael

Sencha

Code base

100%

JavaScript

Rich Internet Applications

Rich

~~Internet~~ Client

Applications

Responsive
Design

Rich

~~Internet~~ Client

Applications

Responsive
Design

Rich

~~Internet~~ Client

Applications

UX

Responsive
Design

Rich

~~Internet~~ Client

Applications

UX

UI

Responsive
Design

Rich

~~Internet~~ Client

Applications

UX
Performance
UI

Responsive
Design

Rich

Mobile

~~Internet~~

Client

Applications

UX
Performance
UI

Responsive
Design

iOS

Mobile

Rich

~~Internet~~

Client

Applications

UX
Performance
UI

Responsive
Design

Android

iOS

Mobile

Rich

~~Internet~~

Client

Applications

UX
Performance
UI

Responsive
Design

Android

iOS

Webkit

Mobile

Rich

~~Internet~~ Client

Applications

UX
Performance
UI

Responsive
Design

Android

iOS

Webkit

Mobile

ie

Rich

~~Internet~~ Client

Applications

UX
Performance
UI

Responsive
Design

Android

iOS

Webkit

Mobile

7 ie

Rich

~~Internet~~ Client

Applications

UX
Performance
UI

Responsive
Design

Android

iOS

8

Webkit

Mobile

7 ie

Rich

~~Internet~~ Client

Applications

UX
Performance
UI

Responsive
Design

Android

iOS

Webkit

Mobile

8

7 ie

9

Rich

~~Internet~~ Client

Applications

UX
Performance
UI

Responsive
Design

Android

iOS

Webkit

8

7 ie 10

9

Rich

Mobile

Client

~~Internet~~

Applications

UX
Performance
UI

DOM

Android

Responsive
Design

Webkit

iOS

8

Rich

Mobile

7 ie 10

9

~~Internet~~

Client

Applications

UX
Performance
UI

Canvas

Responsive
Design

DOM

Android

Webkit

iOS

8

Rich

Mobile

7 ie 10

9

~~Internet~~

Client

Applications

UX
Performance
UI

WebGL

Canvas

Responsive

DOM

Android

Design

Webkit

iOS

8

Rich

Mobile

7 ie 10

9

~~Internet~~

Client

Applications

UX
Performance
UI

Empiezan a aparecer

Problemas

de ingeniería



¿Por qué?

Empiezan a aparecer

Problemas

de ingeniería

a b a r c a r

t o d a s l a s

p l a t a f o r m a s

responder

rápido al

cambio

desarrollar
rápido

reutilizar código

reutilizar código

reutilizar código

reutilizar

reutilizar código

reutilizar código

reutilizar código

código

reutilizar código

reutilizar código

reutilizar código

¡divertirnos!

“Scripteamos”

la web

jQuery al DOM

“... seguro hay un
plugin de jQuery
que lo hace ...”

```
$(function() {  
  $("input[type=search]").chosen();  
  $("select").change(function(e) {  
    $(".loading").fadeIn();  
    var category = $(this).val();  
    var query = $("input[type=search]").val();  
    $.ajax({  
      url: "http://api.sky.net/terminators",  
      dataType: "jsonp"  
      data: {  
        category: category,  
        query: query  
      },  
      success: function(data) {  
        $(".loading").fadeOut();  
        VIEWS.renderTerminator(data);  
        window.history.pushState("/search?query=" +  
          query + "&category=" + category);  
      }  
      error: function() {  
        $(".loading").fadeOut();  
      }  
    });  
  });  
});
```

... pero ...

spaghetti code

¿testeabilidad &
mantenibilidad?

Lógica de presentación
mezclada con lógica de
negocios

Entonces

.

.

¿Qué hacemos?

TDD

BDD

Continuous Deployment

BDD

Continuous Deployment

BDD

Tests de aceptación

¿números romanos ...

MVC MVVM MVP

... o qué?

MVC

Model

Model

Objeto de **dominio**

Model

Encapsula la
lógica de negocio

Model

Agnóstico de la vista

Model

Maneja o dialoga con
los servicios de
persistencia

View

View

Representación visual de los modelos

View

Layout

View

DOM

HTML

CSS

View

touchup

zoom

swipe

keypress

Mapeo de gestos

mousedown

hover

blur

click

focus

View

Observa al modelo

View

Reacciona ante
cambios del modelo y
renderiza la vista

Controller

Controller

Observa a la vista

Controller

Reacciona ante
eventos de la vista y
modifica el modelo

—— Llamada a un método

----- Notificación de evento

Controller

Actualiza
el estado
del modelo

Actualiza
elementos
de la vista

Notifica gestos

Model

View

Notifica cambios

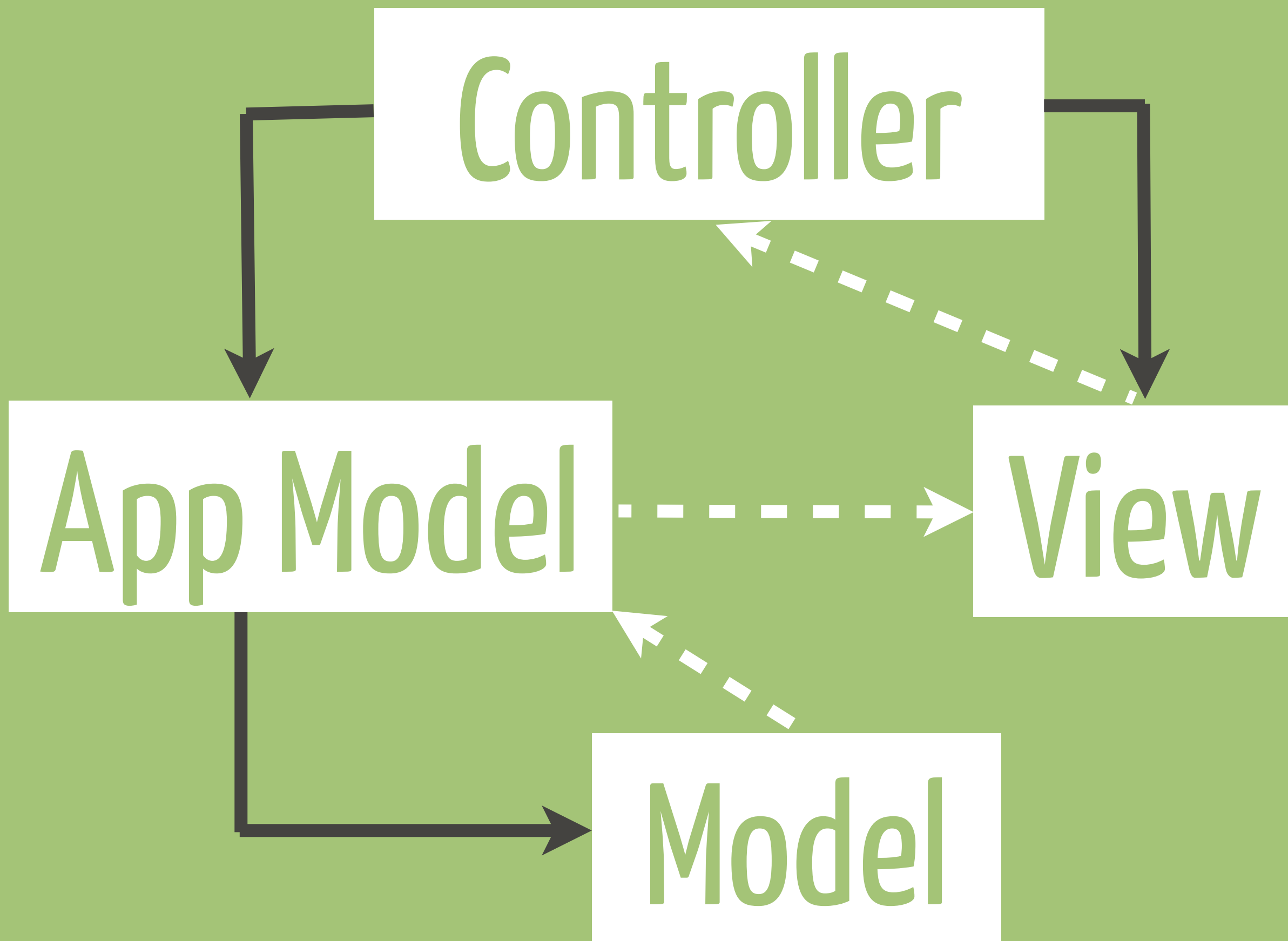
a la vista para ser renderizados



Application Model

—— Llamada a un método

----- Notificación de evento



MVP

View

View

Las vistas son
“tontas”

View

Expone una **interfaz**

View

Delega todo al
presenter

View

Depende del
presenter para
actualizarse

View

No observa al modelo

Presenter

Presenter

Encapsula lógica de presentación

Presenter

Observa al modelo

Presenter

Modifica la vista
mediante su interfaz

Presenter

Desconoce la
implementación
concreta de la vista

Presenter

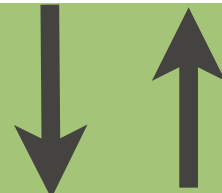
Mantiene estado
de la vista

—— Llamada a un método

----- Notificación de evento

View

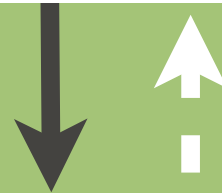
Notifica interacción



Actualiza la vista

Presenter

Actualiza el estado del modelo



Notifica cambios de propiedades

Model

M V V M

View Model

View Model

Recibe comandos
de la vista

View Model

Observa al modelo

View Model

Expone propiedades
a la vista

View Model

Depende fuertemente
del data binding

—— Llamada a un método

----- Notificación de evento

View

Envío de comandos



Notifica cambios para renderiar

View Model

Actualiza el estado del modelo



Notifica cambios de propiedades

Model

¿Cuál usamos?

MVC



MVVM

MVP



JavaScript

MVC

X

X

MVVM

MVP

X

Knockout

Backbone

Spine

Ember

Knockout

Backbone

Spine

Ember

Knockout

Implementa MVVM

Knockout

La vista es el **HTML**

Knockout

Los **bindings** se
hacen en el **HTML**

```
<div class="user-view">
```

```
  Nombre:
```

```
  <strong data-bind="text:firstName">
```

```
  </strong>
```

```
</div>
```

Knockout

Cambios en el
view model actualizan
automáticamente la vista

Knockout

```
var UserViewModel = function(firstName,
                                lastName) {
    this.firstName = ko.observable(firstName);
    this.lastName = ko.observable(lastName);
}
var vm = new UserViewModel("juan", "pérez");
ko.applyBindings(vm, "div.user-view");

...

vm.firstName("pablo");

...
```

Knockout

Incluye su propio
template engine

Pero se puede usar otro

Knockout

Es una librería **simple**
y **fácil** de usar

Knockout

... pero ...

Knockout

No tiene manejo de rutas ...

... se puede usar **sammy.js**

Knockout

No

hay herencia de
vistas

Knockout

El estado de los modelos
debe ser **persistido** de
forma “**manual**”

Knockout

Totalmente
acoplado al DOM

Knockout

Backbone

Spine

Ember

Backbone

Implementa ... MV# ...

Backbone

Collection Model
View Router

A vertical dashed line separates the classes into two groups. On the left, 'Collection' and 'View' are listed. On the right, 'Model' and 'Router' are listed. 'Collection' and 'Router' are in white, while 'View' and 'Model' are in blue.

Tiene 4 clases

Backbone

```
var Workspace = Backbone.Router.extend({  
  routes: {  
    "help": "help",  
    "search/:query": "search",  
    "search/:query/p:page": "search"  
  },  
  help: function() { ... },  
  search: function(query, page) { ... }  
});
```

Se encarga de asociar
una ruta a una función

Router

Backbone

```
var User = Backbone.Model.extend({  
  defaults: {  
    firstName: "juan",  
    lastName: "pérez"  
  }  
  fullName: function() {  
    return this.get("firstName")  
      + " " + this.get("lastName");  
  }  
});  
new User().on("change:firstName",  
  function(){alert("Hello!");});
```

Notifica **eventos**

cuando cambia el estado

Model

Backbone

```
user.set("firstName", "Guido");
```

... genera automáticamente ...

```
POST /users/1 HTTP/1.1
```

```
Content-Type: "application/json"
```

```
{"id": "1", "firstName": "Guido"}
```

Cada modificación es
enviada al servidor

Model

Backbone

```
var Users = Backbone.Collection.extend({  
  model: User  
});
```

Es un set ordenado de
objetos de modelo

Collection

Backbone

Create	- POST	/collection
Read	- GET	/collection
Update	- PUT	/collection/id
Delete	- DELETE	/collection/id

Se **sincroniza** con
el **servidor** mediante
operaciones **REST**

Collection

Backbone

add

destroy

remove

sync

reset

error

Notifica eventos

Collection

Backbone

```
UserView = Backbone.View.extend({  
  events: {  
    ".update-btn click": "updateUser"  
  },  
  initialize: function() {  
    this.model.on("change", this.render,  
    this);  
  },  
  render: function() {  
    // Update DOM  
  }  
});
```

Representa un componente
de UI reusable

View

Backbone

Depende de
Underscore y jQuery o Zepto

Backbone

Se puede usar cualquier
template engine

Backbone

Extender backbone es fácil

Backbone

Buena documentación

Backbone

Chaplin
Thorax
Marionette

Knockout

Backbone

Spine

Ember

Spine

Dice implementar MVC

Spine

Es el primo lejano de backbone

Spine

Implementado en Coffeescript

Spine

```
UserView = Spine.Controller.sub({
  events: {
    ".update-btn click": "updateUser"
  },
  init: function() {
    var render = this.proxy(this.render, this);
    this.item.bind("change", render);
  },
  render: function() {
    // Update DOM
  }
});
```

Spine Controller = Backbone View

No hay colecciones como en backbone

```
var users = User.all();  
var user = User.find(1);  
var user = User.findByAttribute("name", "juan");  
var allowedUsers = User.select(function(user) {  
    return user.age >= 18  
});
```

Spine

Soporta **persistencia** usando
Ajax o **LocalStorage**

Spine

No hay router

```
var MyController = Spine.Controller.sub({  
  init: function() {  
    this.routes({  
      "/user/:email" : function(params) {  
        console.log(params.email)  
      }  
    });  
  }  
});
```

Las **rutas** se declaran en
el **controller**

Spine

```
var PostsShow = Spine.Controller.sub();  
var PostsEdit = Spine.Controller.sub();  
  
var Posts = Spine.Stack.sub({  
  controllers: {  
    show: PostsShow,  
    edit: PostsEdit  
  },  
  default: "show"  
});
```

Los **controllers** se pueden
agrupar en **stacks**

Knockout

Backbone

Spine

Ember

Ember

Se puede usar cualquier patrón

MVC MVVM MVP

Ember

Ember.View

Ember.Object

Data Binding

Handlebars

Ember

```
app.user = Ember.Object.create({
  firstName: "Juan",
  lastName: "Pérez",
  age: 19
});
app.userView = Ember.View.create({
  firstNameBinding: "app.user.firstName",
  lastNameBinding: "app.user.lastName",
  textColor: function() {
    return (app.user.get("age")) >= 18 ?
      "green" : "red";
  }.property();
});
```

Incluye un poderoso sistema
de data **binding**

Ember

```
<script type="text/x-handlebars">  
  <div {{bindAttr class="textColor"}}>  
    Nombre: {{app.userView.firstName}}  
  </div>  
</script>
```

Las vistas escuchan
propiedades y se actualizan
solas

Ember

El template engine
es **handlebars**

Ember

Es posible armar una **jeraquía**
de vistas

Ember

No tiene manejo de
persistencia

Ember Data Module

Ember

No tiene manejo
de rutas

Sprout Core Routing
Module

Knockout

Backbone

Spine

Ember



Backbone

Como se implementó la barra de búsqueda

Todas las categorías

Silla escritorio

ORDENAR POR:

Relevancia | Menor precio | Mayor precio

FILTRAR POR:

Nuevos Usados Mejores vendedores

Productos encontrados para "Silla escritorio"

\$690



Nuevo 242 vendidos

Sillon ejecutivo respaldo alto, silla escritorio pc comodo

En Villa Devoto, Capital Federal

\$699



Nuevo 118 vendidos

Sillon ejecutivo gerencial respaldo alto, silla escritorio!!

En Lanus, Buenos Aires

\$689.9



Nuevo 179 vendidos

Sillon ejecutivo gerencial silla de pc / escritorio alto

En Villa Devoto, Capital Federal

\$459.99



Nuevo 53 vendidos

Sillon ejecutivo gerencial pc escritorio oficina entrego ya

En Cap.fed., Buenos Aires

\$199.99



Nuevo 78 vendidos

Silla oficina regulable escritorio entrego hoy garantia ext

En Cap.fed., Buenos Aires

\$439.9



\$399.9



\$439.9



\$439.9



\$179



SearchView

Categoría

Orden

Todas las categorías ▾

Silla escritorio × 🔍

ORDENAR POR:
Relevancia | Menor precio | Mayor precio

FILTRAR POR:
Nuevos Usados Mejores vendedores

Búsqueda

Filtro

¿Qué?

Animales y Mascotas ▼ conejos × 🔍

¿Cuáles?

ORDENAR POR: Relevancia | Menor precio | Mayor precio

FILTRAR POR: Nuevos x Usados

¿Dónde?

Buscar en Buenos Aires ▼

HTML5 History API

Búsqueda

Condición

Categoría

mercado - la nueva forma de comprar el mejor
n/search/conejos?orderBy=price_desc&condition=new&province=AR-B&category=MLA1071

Orden

Provincia

Productos encontrados para "conejos"



\$3000

Nuevo

Jaulon polivalente y 10 conejos juvenes.

En Boulogne, Buenos Aires



\$2250

Nuevo

Conejera 10 jaulas, 3 c/parideras marca le clavier

En Villa Ballester, Buenos Aires, Buenos Aires



\$1498

Nuevo

Huron gato cobayo chinchilla hamster conejo jaula especial

En Temperley, Buenos Aires



\$1498

Nuevo

Huron gato cobayo chinchilla hamster conejo jaula especial

En Temperley, Buenos Aires



\$1498

Nuevo

Huron gato cobayo chinchilla hamster conejo jaula especial

En Temperley, Buenos Aires



\$1498



\$1498



\$1000

Nuevo



\$800



\$500

ArticleCollection

Search

SearchView

URLSearchView



¡Muchas Gracias!

@guidomb

@mrtnslv