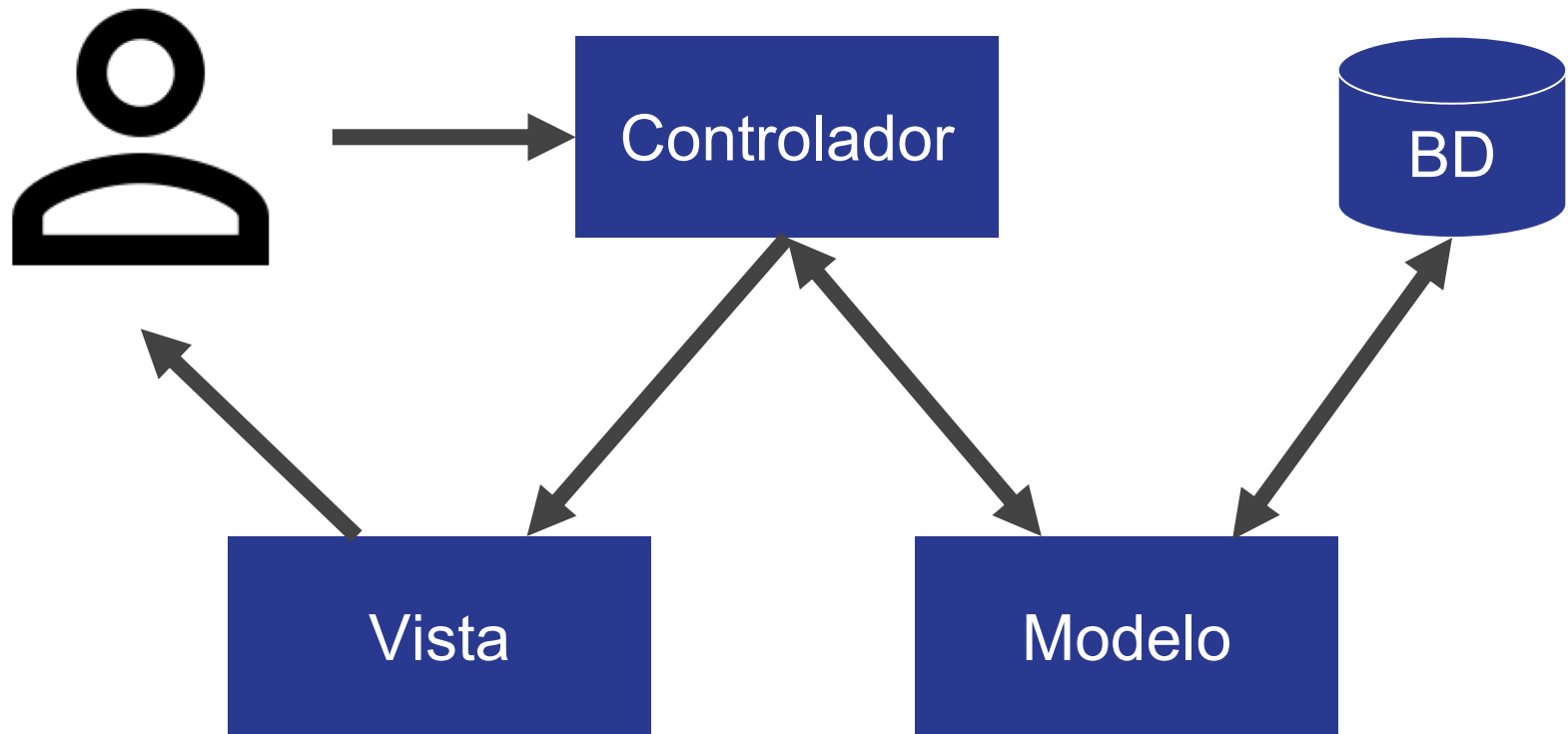


Ayudantía 3

Rails CRUD y Routing

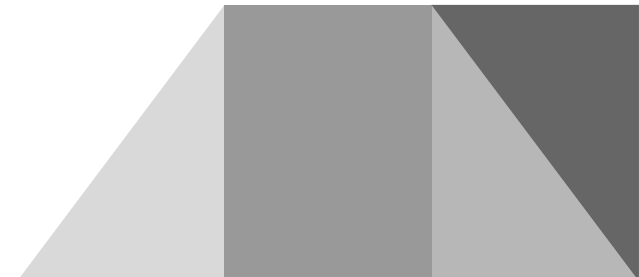
Modelo-Vista-Controlador (MVC)



Rails CRUD

Create, Read, Update y Delete

			Método HTTP
Create	New	Create	POST
Read	Index	Show	GET
Update	Edit	Update	PUT/PATCH
Delete	Delete	Destroy	DELETE



Create, Read, Update y Delete

			Método HTTP
Create	New	Create	POST
Read	Index	Show	GET
Update	Edit	Update	PUT/PATCH
Delete	Delete	Destroy	DELETE

Create, Read, Update y Delete

			Método HTTP
Create	New	Create	POST
Read	Index	Show	GET
Update	Edit	Update	PUT/PATCH
Delete	Delete	Destroy	DELETE

Create, Read, Update y Delete

			Método HTTP
Create	New	Create	POST
Read	Index	Show	GET
Update	Edit	Update	PUT/PATCH
Delete	Delete	Destroy	DELETE

Create, Read, Update y Delete

			Método HTTP
Create	New	Create	POST
Read	Index	Show	GET
Update	Edit	Update	PUT/PATCH
Delete	Delete	Destroy	DELETE

Create, Read, Update y Delete

			Método HTTP
Create	New	Create	POST
Read	Index	Show	GET
Update	Edit	Update	PUT/PATCH
Delete	Delete	Destroy	DELETE



Ejemplo en código

Routing

Routing

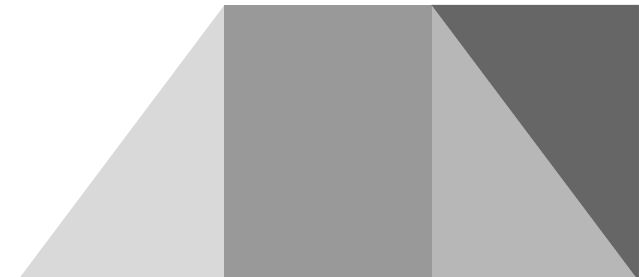
- Dado el path de una request, determina qué controlador la procesará.

GET /forums/4860

POST /comments

PUT /users/13637542

DESTROY /planets/51





Continuación ejemplo en
código

Apéndice: Comandos y Código usados en la ayudantía

Creación de modelos, vistas y controladores

Scaffold (crea modelo, migración, controladores y vistas):

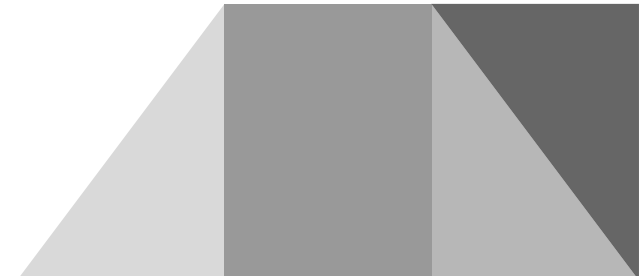
```
docker-compose run web rails g scaffold Forum title:string description:text
```

Model (crea solo modelo y migración):

```
docker-compose run web rails g model Post title:string content:text forum:references
```

Controller (crea controladores y vistas):

```
docker-compose run web rails g controller Posts new create edit update show index destroy
```



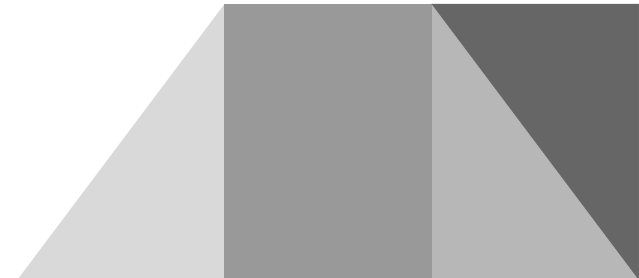
Definición de asociaciones en los modelos

- En `app/models/forum.rb`

```
has_many :posts, dependent: :destroy
```

- En `app/models/post.rb`

```
belongs_to :forum
```

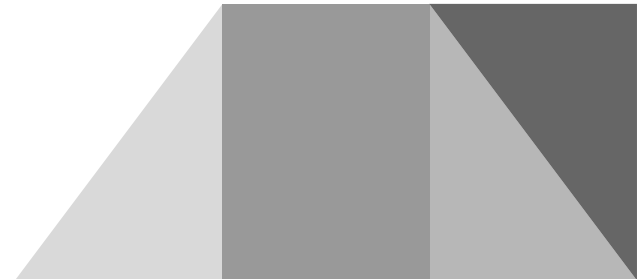


Correr migraciones

```
docker-compose run web rails db:migrate
```

Ver estado:

```
docker-compose run web rails db:migrate:status
```



Métodos en controladores

Acciones que quiero sean ejecutadas antes de x método

```
before_action :set_post, only: %i[show edit update destroy]
```

Métodos Comunes

```
def index
  @posts = Post.all
end

def create
  @post = Post.new(post_params)
  if @post.save
    redirect_to @post
  else
    render 'new'
  end
end

def destroy
  @post.destroy
  redirect_to posts_path
end
```

Métodos en controladores

Métodos Privados

```
private
  def set_forum
    @forum = Forum.find(params[:forum_id])
  end

  def set_post
    @post = Post.find(params[:id])
  end

  def post_params
    params.require(:post).permit(:title, :content, :forum_id)
  end
```



Routing

Forma default:

```
get '/forums/:id', to: 'forums#show'
```

Forma recomendada:

```
resources :forums
```

Forma recomendada para rutas anidadas:

```
resources :forums do  
  resources :posts  
end
```

