

Componentes

Componentes

La funcionalidad de los controladores es limitada.

Symfony ofrece servicios adicionales a través de **componentes**.

Para usarlos, hay que:

1. Instalarlo, en caso de que no sea nativo.
2. En la clase controladora que lo va a usar, añadir el “`use ...`” al principio del fichero.
3. Pasar al método controlador un objeto (a veces).
4. Leer la documentación.

Ejemplos:

- `Session` para gestionar sesiones.
- `Request` para para manipular parámetros GET, POST, cookies, etc.
- [Muchísimos otros.](#)

Componentes: envío de emails

Se usa el componente nativo Mailer:

1. Configura `config/packages/messenger.yaml` para que los emails se envíen directamente, sin necesidad de configurar otros componentes:

```
routing:
    # Symfony\Component\Mailer\Messenger\SendEmailMessage: async
    Symfony\Component\Notifier\Message\ChatMessage: async
    Symfony\Component\Notifier\Message\SmsMessage: async
```

2. El servidor SMTP se indica en `/.env`
`MAILER_DSN=smtp://usuario:pass@servidor:2525`
3. Prueba el ejemplo.

Componentes: bases de datos

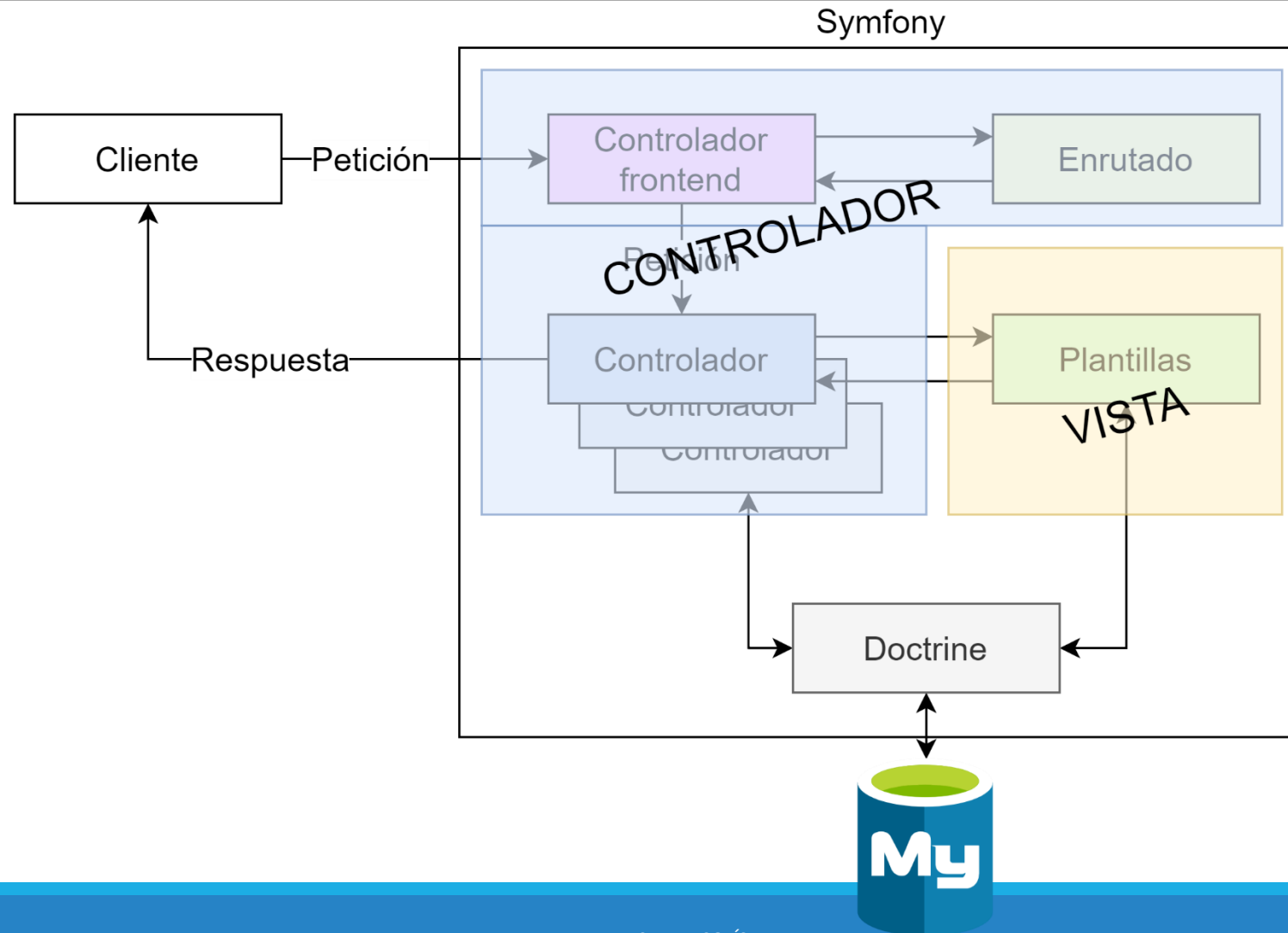
Usamos el componente nativo Doctrine:

1. La configuración de la BD va en `/.env`

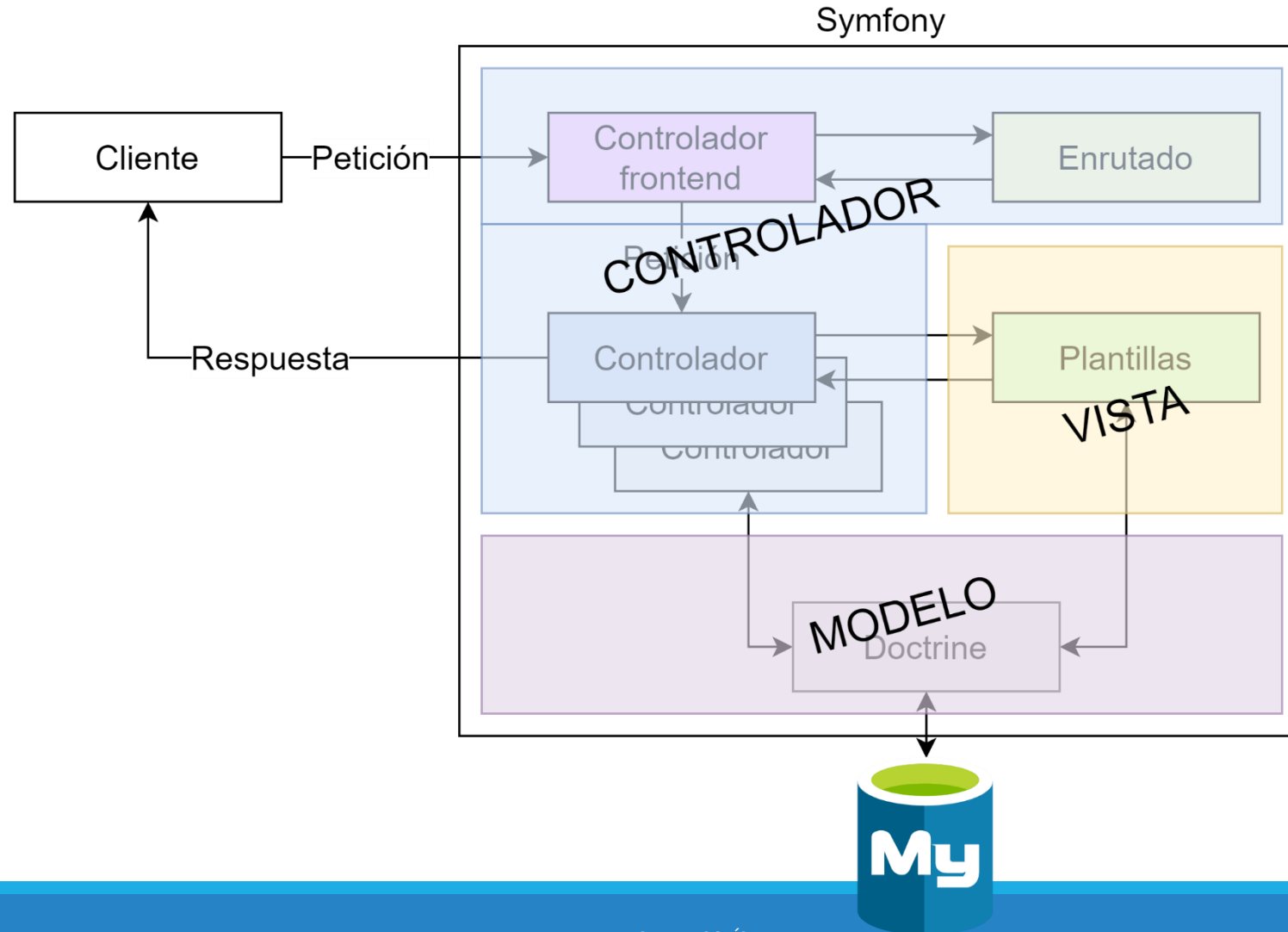
```
DATABASE_URL=mysql://root:@127.0.0.1:3306/doctrine
```

2. Las entidades van en `/src/Entity`
3. Symfony se encarga de arrancar Doctrine por nosotros (ya **NO** hay un `bootstrap.php`).
4. Prueba los ejemplos.

Componentes: bases de datos



Componentes: bases de datos



Componentes: formularios

Escribimos los formularios directamente en las plantillas Twig.

...pero Symfony tiene un componente específico para automatizarlo.

Al principio resulta complejo.

No lo usaremos por falta de tiempo.

<https://symfony.com/doc/current/forms.html>