



Taller práctico

Evolución y Gestión de la Configuración
Curso 2021/22



¡Antes de empezar!

WIKI OFICIAL DEL PROYECTO

github.com/drorganvidez/evidentia/wiki

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

1. Introducción a Evidentia



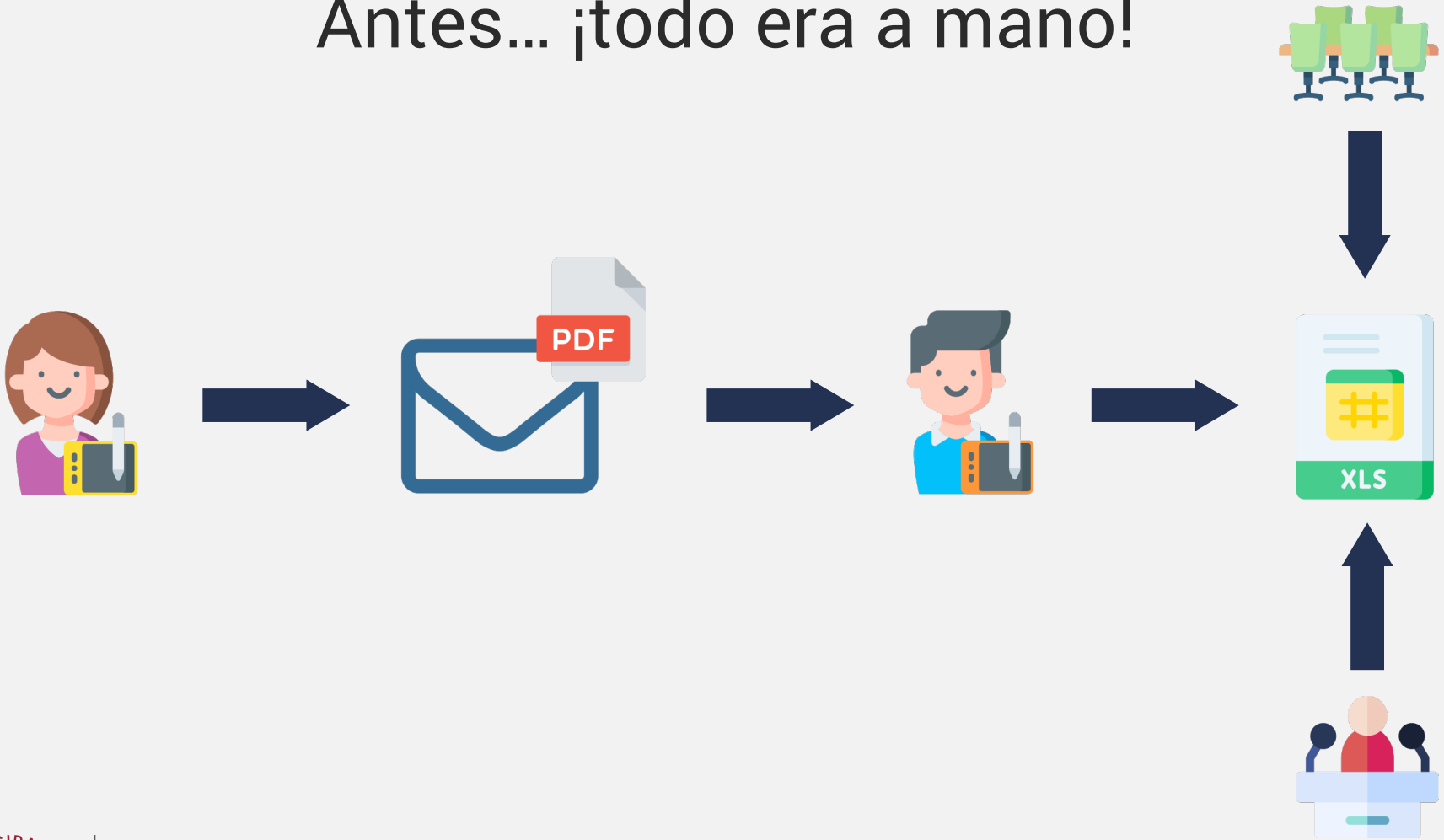
InnoSoft Days

Software libre para la
gestión de evidencias de
trabajo en jornadas
docentes

www.evidentia.cloud

1. Introducción a Evidentia

Antes... ¡todo era a mano!



1. Introducción a Evidentia

Solución: automatización

Evidencias

Convocatorias
y actas

Eventos y
asistencias

Bonos de
horas

EVIDENTIA

LARAVEL V8

ÍNDICE

1. Introducción a Evidentia
2. **Laravel, un framework PHP**
3. En este taller...
4. UML
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

2. Laravel, un framework PHP



Framework PHP para
artesanos de la web

Documentación oficial de la versión usada

<https://laravel.com/docs/8.x/releases>

2. Laravel, un framework PHP



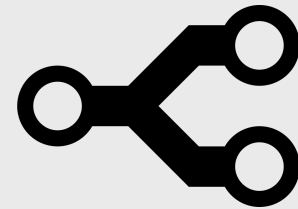
Taylor Otwell



Creado en
2011



Basado en
Symfony



21K forks
en Github

2. Laravel, un framework PHP

¿Por qué Laravel?



```
// Variables
session_start();
$username = $_SESSION["username"]
$email = $_POST["email"];

// Conexión con la base
$link = mysql_connect("localhost", "tu_user", "tu_password");

// Selección de la base de datos
mysql_select_db("mi_base_datos");

// Sentencia SQL y ejecución
$sql = "UPDATE Users SET email = '$email' WHERE username = '$username'";
mysql_query($sql);

// Cierre de la conexión
mysql_close($link);
```

2. Laravel, un framework PHP

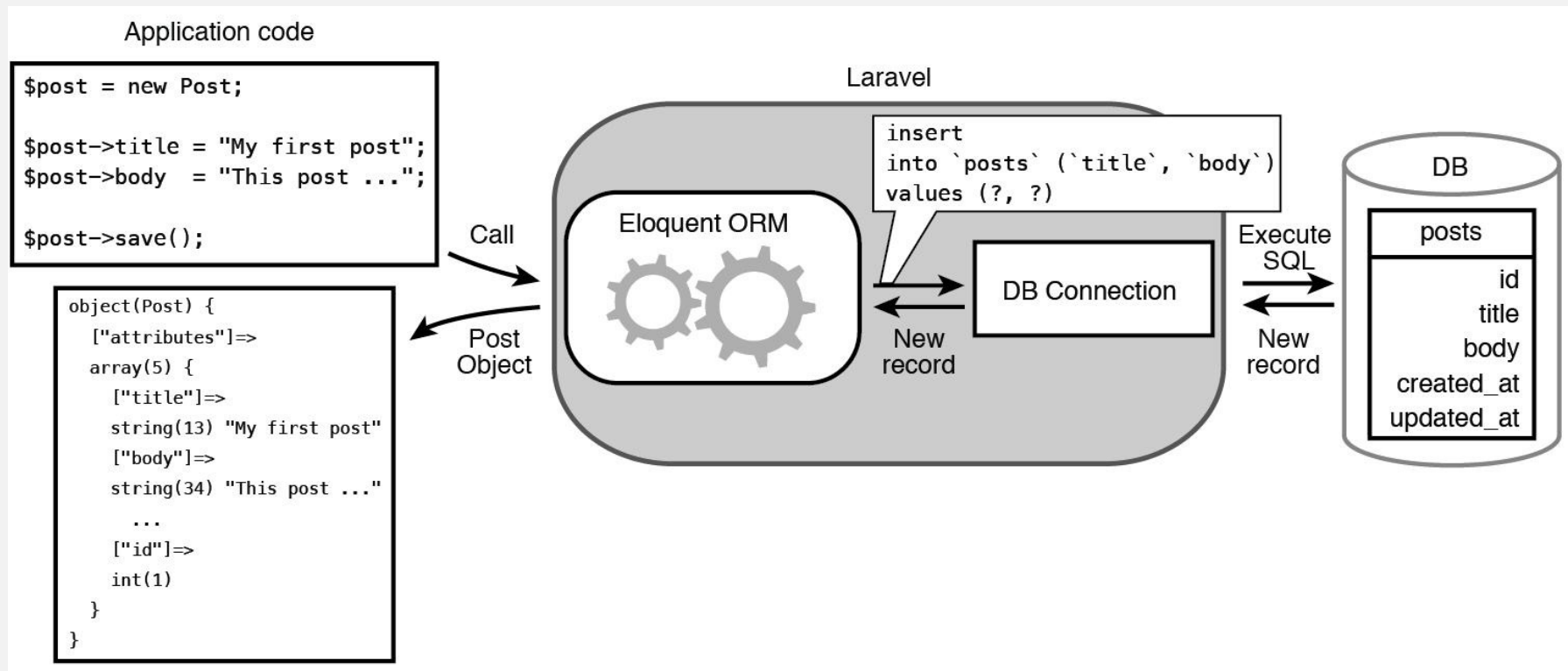
¿Por qué Laravel?



```
$user = Auth::user();  
  
$user->email = request->input("email");  
  
$user->save();
```

2. Laravel, un framework PHP

Eloquent ORM (Object Relational Mapper)



2. Laravel, un framework PHP

Composer



- Gestor de dependencias
- Similar a npm, pip
- Heredero de PEAR
- Repositorio PACKAGIST

composer install



composer.json



composer.lock

2. Laravel, un framework PHP

Consola ARTISAN

```
> artisan
Laravel Framework 8.15.0

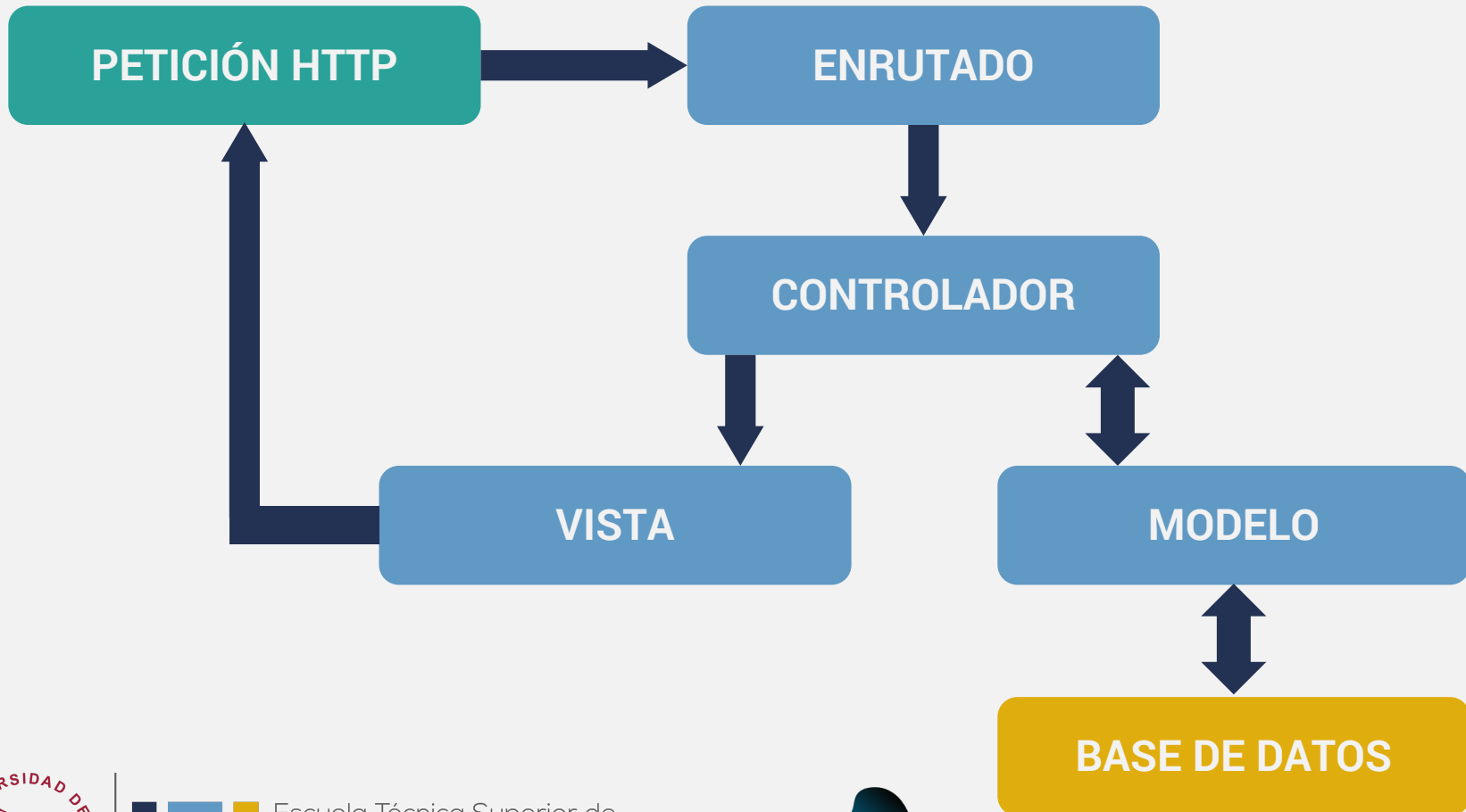
Usage:
  command [options] [arguments]

Options:
  -h, --help                Display this help message
  -q, --quiet               Do not output any message
  -V, --version             Display this application version
      --ansi                Force ANSI output
      --no-ansi             Disable ANSI output
  -n, --no-interaction      Do not ask any interactive question
      --env[=ENV]           The environment the command should run under
  -v|vv|vvv, --verbose      Increase the verbosity of messages: 1 for normal output, 2 for more verbose output and 3 for debug

Available commands:
  clear-compiled      Remove the compiled class file
  down                Put the application into maintenance / demo mode
  env                 Display the current framework environment
  help                Displays help for a command
  inspire             Display an inspiring quote
  list                Lists commands
  migrate             Run the database migrations
  optimize            Cache the framework bootstrap files
  serve               Serve the application on the PHP development server
  test                Run the application tests
  tinker              Interact with your application
  ui                  Swap the front-end scaffolding for the application
  up                  Bring the application out of maintenance mode
```

2. Laravel, un framework PHP

Estructura de Laravel



ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. **En este taller...**
4. UML
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

3. En este taller...

Funcionalidad completa: Notas en Evidentia

Nos piden implementar notas simples

Requisito de información

El sistema deberá gestionar notas. Por cada nota, el sistema guarda un título, la fecha de creación y un contenido.

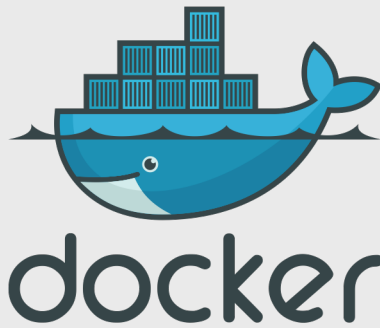
Requisito funcional

Los coordinadores deben poder crear, listar, editar y borrar sus notas.

3. En este taller...

¿Cómo puedo probarlo?

Puedes desplegar en local
usando



o



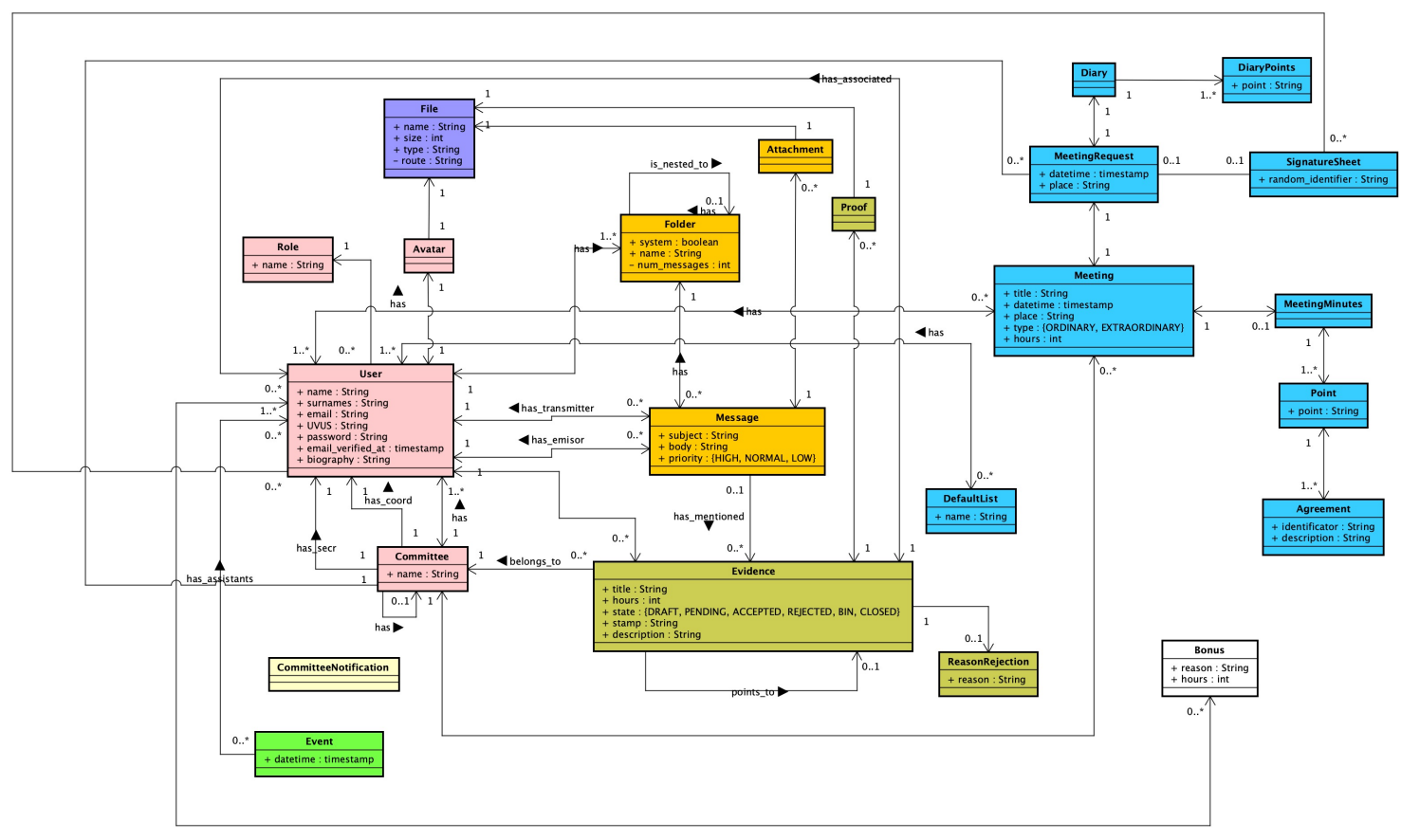
<https://github.com/drorganvidez/evidentia/wiki>

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
- 4. UML**
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

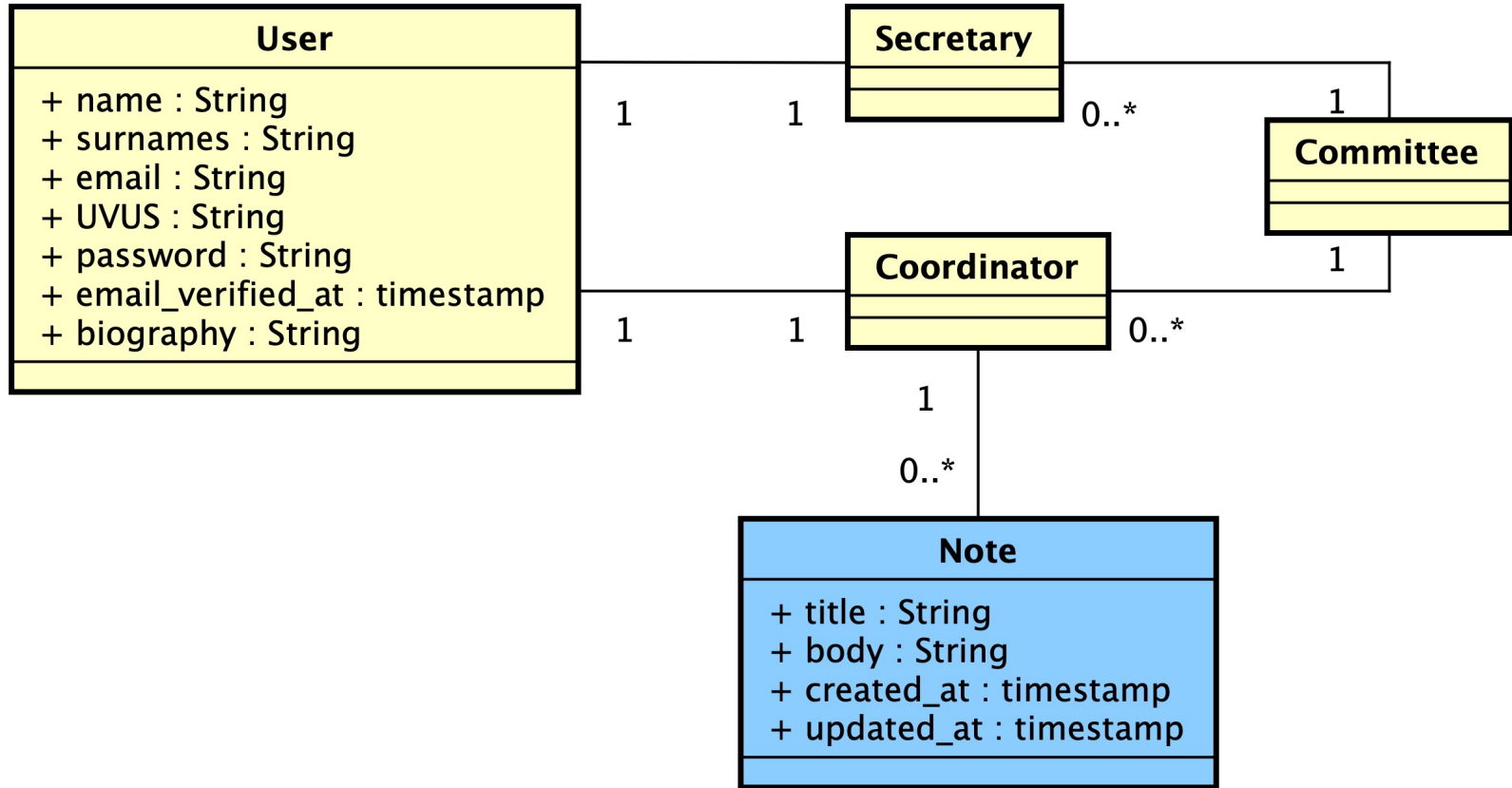
4. UML

Estado actual



4. UML

Nueva entidad: Note



ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. **Migración**
6. Modelo
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

5. Migración

github.com/drorganvidez/taller_laravel

```
php artisan make:migration CreateNotesTable
```

Añadimos...

```
--path database/migrations/develop
```



Ejecutamos la migración

```
php artisan evidencia:reloadinstance
```


5. Migración

github.com/drorganvidez/taller_laravel

```
public function up()
{
    Schema::create('notes', function (Blueprint $table) {
        $table->id();
        $table->string('title');
        $table->text('body');
        $table->foreignId('coordinator_id');
        $table->timestamps();
    });
}

. . .
```

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. **Modelo**
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

6. Modelo

github.com/drorganvidez/taller_laravel

```
php artisan make:model Note
```

app/Models

6. Modelo

github.com/drorganvidez/taller_laravel

```
class Note extends Model
{
    use HasFactory;

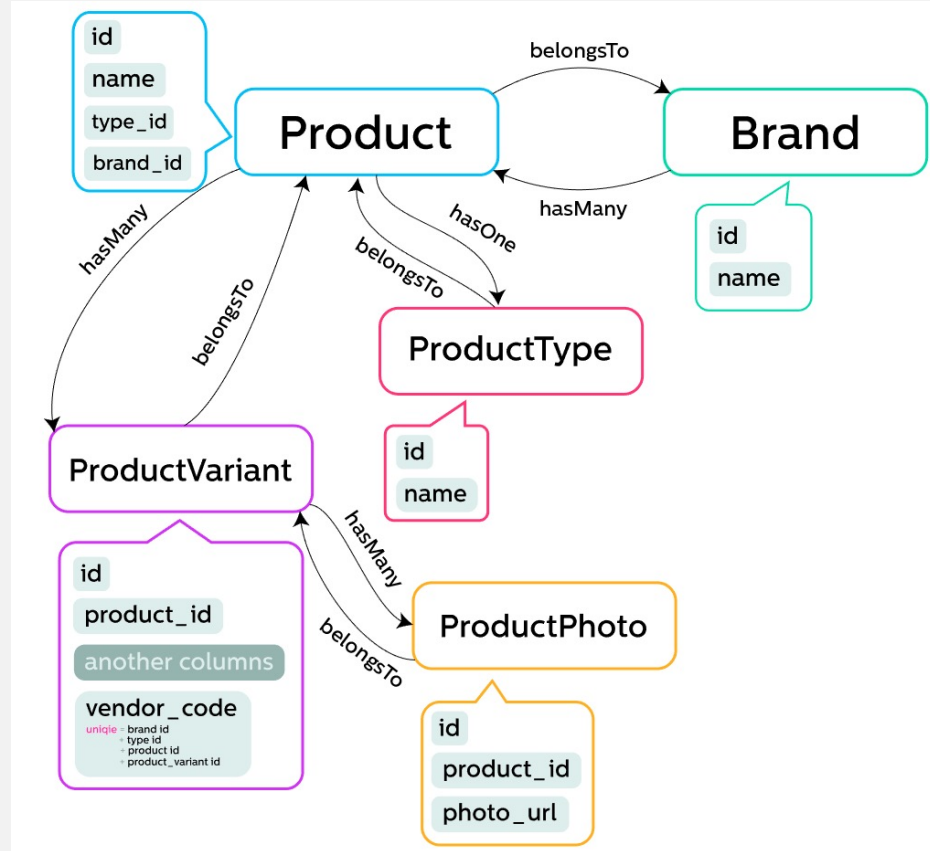
    protected $table = 'notes';

    protected $fillable = [
        'title',
        'body',
        'coordinator_id'
    ];
}
```

app/Models/Note

6. Modelo

github.com/drorganvidez/taller_laravel



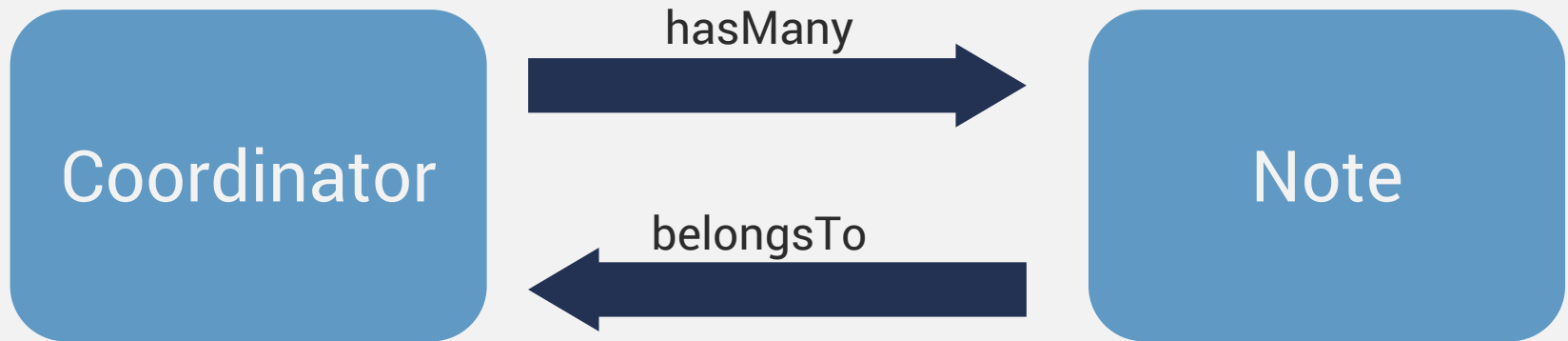
6. Modelo

github.com/drorganvidez/taller_laravel



6. Modelo

github.com/drorganvidez/taller_laravel



Más info en

<https://laravel.com/docs/8.x/eloquent-relationships>

6. Modelo

github.com/drorganvidez/taller_laravel

```
public function coordinator()  
{  
    return $this->belongsTo('App\Models\Coordinator');  
}
```

app/Models/Note

```
public function note()  
{  
    return $this->hasMany('App\Models>Note');  
}
```

app/Models/Coordinator

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. Modelo
7. **Controlador**
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

7. Controlador

github.com/drorganvidez/taller_laravel

```
php artisan make:controller NoteController
```

app/Http

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. Modelo
7. Controlador
8. **Rutas**
9. CRUD
10. Middleware
11. Otras cuestiones

8. Rutas

github.com/drorganvidez/taller_laravel

```
Route::prefix('note')->group(function () {  
  
    Route::get('list', [NoteController::class, 'list']);  
  
    Route::get('view/{id}', [NoteController::class, 'view']);  
  
    Route::get('create', [NoteController::class, 'create']);  
    Route::post('create_p', [NoteController::class, 'create_p']);  
  
    Route::get('edit/{id}', [NoteController::class, 'edit']);  
    Route::post('edit_p', [NoteController::class, 'edit_p']);  
  
    Route::post('remove', [NoteController::class, 'remove']);  
  
});
```

routes/web.php

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. **CRUD**
10. Middleware
11. Otras cuestiones

9. CRUD

github.com/drorganvidez/taller_laravel

Listar notas

```
public function list()
{
    $coordinator = Auth::user()->coordinator;

    $notes = $coordinator->notes;

    return view('notes.list', [
        'instance' => \Instantiation::instance(),
        'notes' => $notes
    ]);
}
```

app/Http/Controllers/NoteController

9. CRUD

github.com/drorganvidez/taller_laravel

Ver una nota

```
public function view($instance,$id)
{
    $note = Note::findOrFail($id);

    return view('notes.view', [
        'instance' => \Instantiation::instance(),
        'note' => $note
    ]);
}
```

app/Http/Controllers/NoteController

9. CRUD

github.com/drorganvidez/taller_laravel

Crear una nota (GET)

```
public function create()
{
    return view('notes.createandedit', [
        'instance' => \Instantiation::instance(),
    ]);
}
```

app/Http/Controllers/NoteController

9. CRUD

github.com/drorganvidez/taller_laravel

Crear una nota (POST)

```
public function create_p(Request $request)
{
    $request->validate([
        'title' => 'required|min:5|max:255',
        'body' => 'required|min:5|max:255'
    ]);

    $coordinator = Auth::user()->coordinator;

    Note::create(
        [
            'title' => $request->input('title'),
            'body' => $request->input('body'),
            'coordinator_id' => $coordinator->id
        ]
    );

    return redirect()->route('note.list', \Instantiation::instance())
        ->with('success', 'Nota creada con éxito');
}
```

app/Http/Controllers/NoteController

9. CRUD

github.com/drorganvidez/taller_laravel

Editar una nota (GET)

```
public function edit($instance, $id)
{
    $note = Note::findOrFail($id);

    return view('notes.createandedit', [
        'instance' => \Instantiation::instance(),
        'note' => $note,
        'edit' => true
    ]);
}
```

app/Http/Controllers/NoteController

9. CRUD

github.com/drorganvidez/taller_laravel

Editar una nota (POST)

```
public function edit_p(Request $request)
{
    $request->validate([
        'title' => 'required|min:5|max:255',
        'body' => 'required|min:5|max:255'
    ]);

    $note = Note::findOrFail($request->input('note_id'));

    $note->title = $request->input('title');
    $note->body = $request->input('body');
    $note->save();

    return redirect()->route('note.list', \Instantiation::instance())
        ->with('success', 'Nota editada con éxito');
}
```

app/Http/Controllers/NoteController

9. CRUD

github.com/drorganvidez/taller_laravel

Eliminar una nota (POST)

```
public function remove(Request $request)
{
    $note = Note::findOrFail($request->input('note_id'));

    $note->delete();

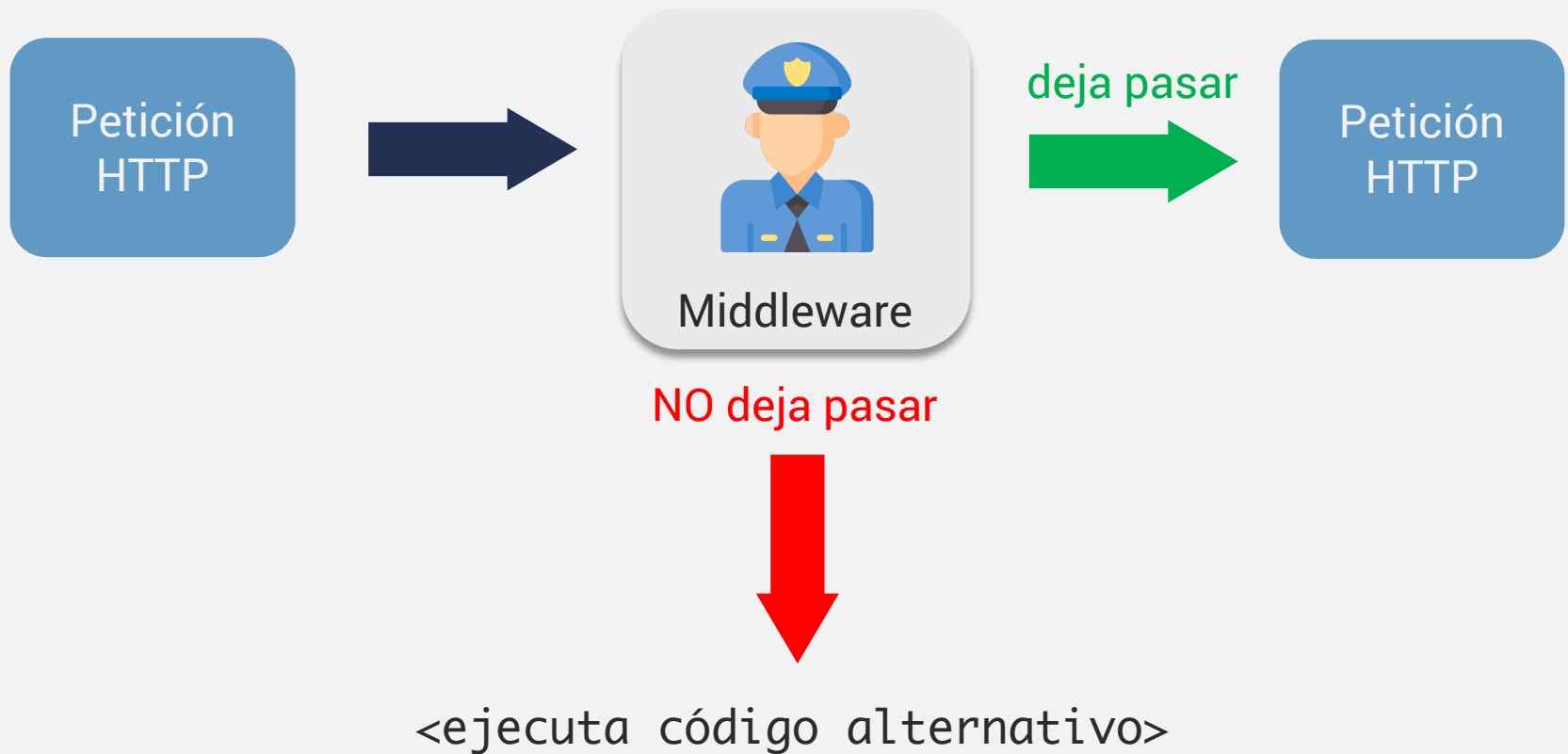
    return redirect()
        ->route('note.list', \Instantiation::instance())
        ->with('success', 'Nota eliminada con éxito');
}
```

app/Http/Controllers/NoteController

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. CRUD
- 10. Middleware**
11. Otras cuestiones

10. Middleware



10. Middleware

github.com/drorganvidez/taller_laravel

Middleware de acceso a las notas

```
public function __construct()  
{  
    $this->middleware('auth');  
    $this->middleware('checkroles:COORDINATOR');  
}
```

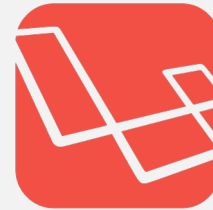
app/Http/Controllers/NoteController

ÍNDICE

1. Introducción a Evidentia
2. Laravel, un framework PHP
3. En este taller...
4. UML
5. Migración
6. Modelo
7. Controlador
8. Rutas
9. CRUD
10. Middleware
11. Otras cuestiones

11. Otras cuestiones

Entorno de desarrollo



Se recomienda el IDE PHPStorm de JetBrains junto con el plugin de Laravel:

<https://www.jetbrains.com/es-es/phpstorm>

Licencia gratuita prorrogable cada año con GitHub

Education Pack: <https://education.github.com/pack>

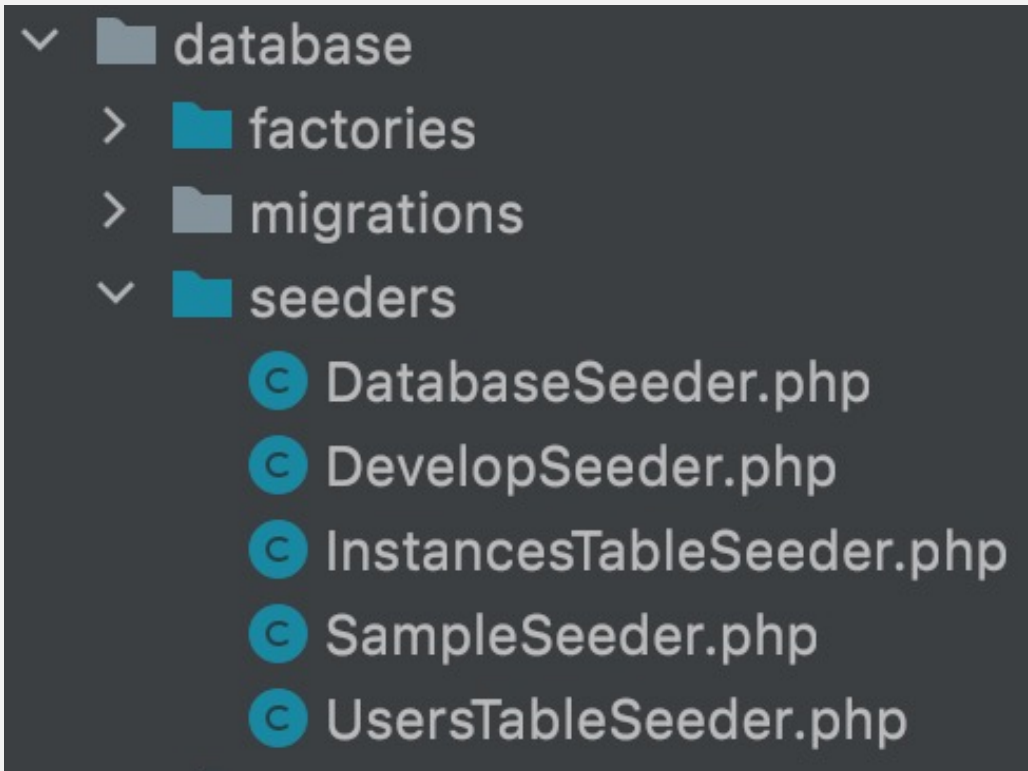
11. Otras cuestiones

Integración continua

GitHub
actions

11. Otras cuestiones

¿Y el populate?



11. Otras cuestiones

Contenedores disponibles en Laradock

¡MÁS DE 60!

<https://laradock.io/introduction/#supported-services>



¡Muchas gracias!

¿Dudas, sugerencias...?