

TECNÓLOGO EM ANÁLISE E DESENVOLVIMENTO

Disciplina de Desenvolvimento Web II

Aula 09: Framework Laravel - Blade, ORM e REQUEST

Gil Eduardo de Andrade

Conceitos Preliminares

(<https://laravel.com/>)

Views e Blade Templates

(<https://laravel.com/docs/13.x/views>)

As “**views**” são responsáveis por separar a lógica da sua aplicação da sua lógica de apresentação. Elas são armazenadas no diretório “**resources/views**” e utilizam a poderosa engine de templates Blade.

Criando e Retornando Views

Uma visualização pode ser criada como um arquivo “**.blade.php**” (dentro da pasta “**resources/views**”) e retornada por uma rota ou *controller*.

PHP

```
// Em uma rota ou controller  
return view('greeting', ['name' => 'James']);
```

Blade Templates

O Blade é a engine de templates do Laravel. Diferente de outras, ela não restringe o uso de código PHP puro. Todos os templates Blade são compilados para código PHP puro e cacheados, o que significa que não há sobrecarga de performance.

PHP

```
Hello, {{ $name }}.
```

Diretivas Blade

O Blade oferece diretivas para estruturas de controle comuns:

- **Condicionais:** @if, @elseif, @else, @endif
- **Loops:** @for, @foreach, @while
- **Inclusão de Subviews:** @include('shared.errors')

Layouts com Blade

O Blade facilita a criação de layouts consistentes através da herança de templates:

Layout Principal (layouts/app.blade.php)

```
PHP
<html>
  <head>
    <title>App Name - @yield('title')</title>
  </head>
  <body>
    <div class="container">
      @yield('content')
    </div>
  </body>
</html>
```

View - Filha

```
PHP
@extends('layouts.app')

@section('title', 'Page Title')

@section('content')
  <p>This is my body content.</p>
@endsection
```

Models e Eloquent ORM (<https://laravel.com/docs/13.x/eloquent>)

O Eloquent é o ORM (Object-Relational Mapper) incluído no Laravel. Ele fornece uma simples implementação do padrão ActiveRecord para trabalhar com seu banco de dados. Cada tabela do banco de dados tem um "Model" correspondente que é usado para interagir com essa tabela.

Criando Models

Para criar um model, use o comando Artisan `"make:model"`:

Shell

```
php artisan make:model Flight
```

Convenções do Eloquent

- **Nome da Tabela:** por padrão, o Eloquent assume que o nome da tabela é o plural do nome da **"Model"** em "snake case". Ex: **Flight** model > **flights** table.
- **Chave Primária:** o Eloquent assume que a chave primária é uma coluna com nome **"id"** auto-incrementável.
- **Timestamps:** o Eloquent espera que as colunas **"created_at"** e **"updated_at"** existam. Você pode desabilitar este comportamento com a propriedade `$timestamps = false`, dentro da sua "Model", ou removendo os dois campos dentro da migração.

Operações CRUD Básicas

Recuperando Dados:

PHP

```
$flights = App\Models\Flight::all();  
$flight = App\Models\Flight::find(1);
```

Inserindo Dados:

PHP

```
$flight = new App\Models\Flight;
```

```
$flight->name = 'New Flight';  
$flight->save();
```

Atualizando Dados:

```
PHP  
$flight = App\Models\Flight::find(1);  
$flight->name = 'Updated Flight Name';  
$flight->save();
```

Deletando Dados:

```
PHP  
$flight = App\Models\Flight::find(1);  
$flight->delete();
```

Relacionamentos Eloquent (<https://laravel.com/docs/12.x/eloquent-relationships>)

As tabelas do banco de dados geralmente estão relacionadas umas às outras. Por exemplo, um post de blog pode ter muitos comentários, ou um pedido pode estar relacionado ao usuário que o fez. O Eloquent facilita o gerenciamento e trabalho com esses relacionamentos, suportando vários tipos comuns.

One to One (Um para Um)

Um relacionamento um-para-um é um tipo muito básico de relacionamento de banco de dados. Por exemplo, um modelo User pode estar associado a um modelo Phone.

Definindo o Relacionamento:

Classe de Modelo - User (hasOne)

```
PHP  
<?php
```

```
namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Relations\HasOne;

class User extends Model
{
    /**
     * Obter o telefone associado ao usuário.
     */
    public function phone(): HasOne
    {
        return $this->hasOne(Phone::class);
    }
}
```

Classe de Modelo - Phone (belongsTo)

```
PHP
class Phone extends Model
{
    /**
     * Obter o usuário que possui o telefone.
     */
    public function user(): BelongsTo
    {
        return $this->belongsTo(User::class);
    }
}
```

Usando o Relacionamento:

```
PHP
$phone = User::find(1)->phone;
$user = Phone::find(1)->user;
```

ou

PHP

```
$phone = User::with(['phone'])->>find(1);  
$user = Phone::with(['user'])->>find(1);
```

One to Many (Um para Muitos)

Um relacionamento um-para-muitos é usado para definir relacionamentos onde um único modelo é pai de um ou mais modelos filhos. Por exemplo, um post de blog pode ter um número infinito de comentários.

Definindo o Relacionamento:

Classe de Modelo - Post (hasMany)

PHP

```
class Post extends Model  
{  
    /**  
     * Obter os comentários do post.  
     */  
    public function comments(): HasMany  
    {  
        return $this->hasMany(Comment::class);  
    }  
}
```

Classe de Modelo - Comment (belongsTo)

PHP

```
class Comment extends Model  
{  
    /**  
     * Obter o post que possui o comentário.  
     */  
    public function post(): BelongsTo  
    {  
        return $this->belongsTo(Post::class);  
    }  
}
```

Usando o Relacionamento:

```
PHP
$comments = Post::find(1)->comments;

foreach ($comments as $comment) {
    echo $comment->body;
}

// Acessar o post de um comentário
$post = Comment::find(1)->post;
```

Many to Many (Muitos para Muitos)

Relacionamentos muitos-para-muitos são mais complexos que relacionamentos hasOne e hasMany. Um exemplo seria usuários que têm muitos papéis (roles), e esses papéis também são compartilhados por outros usuários.

Estrutura de Tabelas:

- users (id, name)
- roles (id, name)
- role_user (user_id, role_id) - Tabela pivot

Definindo o Relacionamento:

Classes de Modelo - User e Role (belongsToMany)

```
PHP
class User extends Model
{
    /**
     * Os papéis que pertencem ao usuário.
     */
    public function roles(): BelongsToMany
    {
        return $this->belongsToMany(Role::class);
    }
}
```

```
class Role extends Model
{
    /**
     * Os usuários que pertencem ao papel.
     */
    public function users(): BelongsToMany
    {
        return $this->belongsToMany(User::class);
    }
}
```

Classes de Modelo - User e Role (belongsTo)

```
PHP
class RoleUser extends Model
{
    public function role() {
        return $this->belongsTo(Role::class);
    }

    public function user() {
        return $this->belongsTo(User::class);
    }
}
```

Usando o Relacionamento:

```
PHP
$user = User::find(1);

foreach ($user->roles as $role) {
    echo $role->name;
}

// Anexar papéis a um usuário
```

```
$user->roles()->attach($roleId);  
  
// Remover papéis  
$user->roles()->detach($roleId);  
  
// Sincronizar papéis (remover todos e adicionar novos)  
$user->roles()->sync([1, 2, 3]);
```

APLICANDO OS CONCEITOS BLADE | ORM | FORMREQUEST

Continuação da Aplicação Criada na Aula Anterior

1) Criando | Codificando: Modelos e Migrações (Disciplina, Aluno, Matricula) (Utilizando SoftDelete)

Comando

(no terminal de comando - dentro da pasta da aplicação)

Shell

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:model Disciplina -m  
  
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:model Aluno -m  
  
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:model Matricula -m
```

Arquivo: "database/migrations/..._create_cursos_table.php"

PHP

```
Schema::create('cursos', function (Blueprint $table) {  
    $table->id();  
    $table->string('nome');  
    $table->integer('duracao');  
    $table->softDeletes(); // Linha Adicionada  
    $table->timestamps();  
});
```

Arquivo: “database/migrations/..._create_disciplinas_table.php”

PHP

```
Schema::create('disciplinas', function (Blueprint $table) {
    $table->id();
    $table->string('nome');
    $table->integer('carga_horaria');
    $table->unsignedBigInteger('curso_id');
    $table->foreign('curso_id')->references('id')->on('cursos');
    $table->softDeletes();
    $table->timestamps();
});
```

Arquivo: “database/migrations/..._create_alunos_table.php”

PHP

```
Schema::create('alunos', function (Blueprint $table) {
    $table->id();
    $table->string('nome');
    $table->integer('turma');
    $table->unsignedBigInteger('curso_id');
    $table->foreign('curso_id')->references('id')->on('cursos');
    $table->softDeletes();
    $table->timestamps();
});
```

Arquivo: “database/migrations/..._create_matriculas_table.php”

PHP

```
Schema::create('matriculas', function (Blueprint $table) {
    $table->unsignedBigInteger('disciplina_id');
    $table->foreign('disciplina_id')->references('id')->on('disciplinas');
    $table->unsignedBigInteger('aluno_id');
    $table->foreign('aluno_id')->references('id')->on('alunos');
    $table->primary(['disciplina_id', 'aluno_id']);
    $table->timestamps();
});
```

OBS.: a ordem de criação das migrações é importante para garantir que o processo de criação das tabelas no banco aconteça corretamente, considerando as relações entre as tabelas.

Arquivo: "app/Models/Curso.php"

```
PHP
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use \Illuminate\Database\Eloquent\SoftDeletes;

class Curso extends Model
{
    use SoftDeletes;

    protected $fillable = [
        'nome',
        'duracao',
    ];

    public function disciplina() {
        return $this->hasMany('\App\Models\Disciplina');
    }

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

Arquivo: "app/Models/Disciplina.php"

```
PHP
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use \Illuminate\Database\Eloquent\SoftDeletes;

class Disciplina extends Model
{
    use SoftDeletes;
```

```
protected $fillable = [
    'nome',
    'carga_horaria',
    'curso_id',
];

public function curso() {
    return $this->belongsTo('\App\Models\Curso');
}

public function aluno() {
    return $this->belongsToMany('\App\Models\Aluno', 'matriculas');
}
}
```

Arquivo: “app/Models/Aluno.php”

```
PHP
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use \Illuminate\Database\Eloquent\SoftDeletes;

class Aluno extends Model
{
    use SoftDeletes;

    protected $fillable = [
        'nome',
        'turma',
        'curso_id',
    ];

    public function curso() {
        return $this->belongsTo('\App\Models\Curso');
    }

    public function disciplina() {
        return $this->belongsToMany('\App\Models\Disciplina', 'matriculas');
    }
}
```

```
}  
}
```

Arquivo: "app/Models/Matricula.php"

```
PHP  
<?php  
  
namespace App\Models;  
  
use Illuminate\Database\Eloquent\Model;  
  
class Matricula extends Model  
{  
    protected $fillable = [  
        'disciplina_id',  
        'aluno_id',  
    ];  
  
    public function disciplina() {  
        return $this->belongsTo('\App\Models\Disciplina');  
    }  
  
    public function aluno() {  
        return $this->belongsTo('\App\Models\Aluno');  
    }  
}
```

2) Criando | Codificando: Seeders (Curso, Disciplina, Aluno e Matricula)

Comando

(no terminal de comando - dentro da pasta da aplicação)

```
Shell  
  
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:seeder CursoSeeder  
  
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:seeder DisciplinaSeeder
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:seeder AlunoSeeder
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:seeder MatriculaSeeder
```

Arquivo: "database/seeders/CursoSeeder.php"

```
PHP  
<?php  
  
namespace Database\Seeders;  
  
use Illuminate\Database\Console\Seeds\WithoutModelEvents;  
use Illuminate\Database\Seeder;  
use Illuminate\Support\Facades\DB;  
  
class CursoSeeder extends Seeder  
{  
    public function run(): void  
    {  
        $data = [  
            ["nome" => "TÉCNICO EM INFORMÁTICA", "duracao" => 4],  
            ["nome" => "TECNÓLOGO EM DESENVOLVIMENTO", "duracao" => 3],  
        ];  
        DB::table('cursos')->insert($data);  
    }  
}
```

Arquivo: "database/seeders/DisciplinaSeeder.php"

```
PHP  
<?php  
  
namespace Database\Seeders;  
  
use Illuminate\Database\Console\Seeds\WithoutModelEvents;  
use Illuminate\Database\Seeder;  
use Illuminate\Support\Facades\DB;
```

```
class DisciplinaSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        $data = [
            ["nome" => "MATEMÁTICA", "carga_horaria" => 3, "curso_id" => 1],
            ["nome" => "FÍSICA", "carga_horaria" => 2, "curso_id" => 1],
            ["nome" => "PROGRAMAÇÃO", "carga_horaria" => 4, "curso_id" => 2],
            ["nome" => "BANCO DE DADOS", "carga_horaria" => 2, "curso_id" => 2],
        ];
        DB::table('disciplinas')->insert($data);
    }
}
```

Arquivo: "database/seeders/AlunoSeeder.php"

```
PHP
<?php

namespace Database\Seeders;

use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\DB;

class AlunoSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        $data = [
            ["nome" => "JOÃO SILVA", "turma" => 2024, "curso_id" => 1],
            ["nome" => "ANA SANTOS", "turma" => 2025, "curso_id" => 1],
            ["nome" => "MARCOS SOUZA", "turma" => 2025, "curso_id" => 2],
            ["nome" => "BRUNA GOMES", "turma" => 2024, "curso_id" => 2],
        ];
    }
}
```

```
        DB::table('alunos')->insert($data);
    }
}
```

Arquivo: "database/seeders/MatriculaSeeder.php"

```
PHP
<?php

namespace Database\Seeders;

use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\DB;

class MatriculaSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        $data = [
            ["disciplina_id" => 1, "aluno_id" => 1],
            ["disciplina_id" => 2, "aluno_id" => 1],
            ["disciplina_id" => 2, "aluno_id" => 2],
            ["disciplina_id" => 3, "aluno_id" => 3],
            ["disciplina_id" => 3, "aluno_id" => 4],
            ["disciplina_id" => 4, "aluno_id" => 4],
        ];
        DB::table('matriculas')->insert($data);
    }
}
```

Arquivo: "database/seeders/DatabaseSeeder.php"

```
PHP
public function run(): void {
    $this->call(CursoSeeder::class);
    $this->call(DisciplinaSeeder::class);
    $this->call(AlunoSeeder::class);
}
```

```
$this->call(MatriculaSeeder::class);
}
```

3) Executando as Migrations/Seeders na Base de Dados

Comando

(no terminal de comando - dentro da pasta da aplicação)

```
Shell
docker-compose exec app php artisan migrate:fresh --seed
```

Tabela de Cursos





		id	nome	duracao
<input type="checkbox"/>	 Editar  Copiar  Remover	1	TÉCNICO EM INFORMÁTICA	4
<input type="checkbox"/>	 Editar  Copiar  Remover	2	TECNÓLOGO EM DESENVOLVIMENTO	3

Tabela de Disciplinas













		id	nome	carga_horaria
<input type="checkbox"/>	 Editar  Copiar  Remover	1	MATEMÁTICA	3
<input type="checkbox"/>	 Editar  Copiar  Remover	2	FÍSICA	2
<input type="checkbox"/>	 Editar  Copiar  Remover	3	PROGRAMAÇÃO	4
<input type="checkbox"/>	 Editar  Copiar  Remover	4	BANCO DE DADOS	2

Tabela de Alunos































		id	nome	turma	curso_id
<input type="checkbox"/>	 Editar  Copiar  Remover	1	JOÃO SILVA	2024	1
<input type="checkbox"/>	 Editar  Copiar  Remover	2	ANA SANTOS	2025	1
<input type="checkbox"/>	 Editar  Copiar  Remover	3	MARCOS SOUZA	2025	2
<input type="checkbox"/>	 Editar  Copiar  Remover	4	BRUNA GOMES	2024	2

Tabela de Matrículas

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	disciplina_id
<input type="checkbox"/>	 Editar	 Copiar	 Remover	1
<input type="checkbox"/>	 Editar	 Copiar	 Remover	2
<input type="checkbox"/>	 Editar	 Copiar	 Remover	2
<input type="checkbox"/>	 Editar	 Copiar	 Remover	3
<input type="checkbox"/>	 Editar	 Copiar	 Remover	3
<input type="checkbox"/>	 Editar	 Copiar	 Remover	4

4) Criando | Codificando: FormRequest (Curso, Disciplina, Aluno e Matricula)

Comando

(no terminal de comando - dentro da pasta da aplicação)

Shell

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:request CursoRequest
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:request DisciplinaRequest
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:request AlunoRequest
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:request MatriculaRequest
```

Arquivo: "app/Http/Requests/CursoRequest.php"

PHP

```
<?php
```

```
namespace App\Http\Requests;
```

```
use Illuminate\Contracts\Validation\ValidationRule;
```

```
use Illuminate\Foundation\Http\FormRequest;
```

```
use Illuminate\Support\Str;
```

```
class CursoRequest extends FormRequest
```

```
{
```

```
/**
 * Determine if the user is authorized to make this request.
 */
public function authorize(): bool
{
    return true;
}

protected function prepareForValidation() {

    $this->merge(
        collect($this->all())->map(function ($value) {
            return is_string($value) ? Str::upper($value) : $value;
        })->toArray()
    );
}

public function messages(): array {
    return [
        "required" => "O preenchimento deste campo é obrigatório!",
        "max" => "Este campo possui tamanho máximo de [:max] caracteres!",
        "min" => "Este campo possui tamanho mínimo de [:min] caracteres!",
        "nome.unique" => "Já existe um curso cadastrado com este nome!",
    ];
}

/**
 * Get the validation rules that apply to the request.
 *
 * @return array<string, ValidationRule|array<mixed>|string>
 */
public function rules(): array
{
    $id = $this->route('curso');

    return [
        'nome' => "required|max:100|min:8|unique:cursos,nome,{ $id }",
        'duracao' => "required",
    ];
}
}
```

Arquivo: "app/Http/Requests/DisciplinaRequest.php"

```
PHP
<?php

namespace App\Http\Requests;

use Illuminate\Contracts\Validation\ValidationRule;
use Illuminate\Foundation\Http\FormRequest;
use Illuminate\Support\Str;

class DisciplinaRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     */
    public function authorize(): bool
    {
        return true;
    }

    protected function prepareForValidation() {

        $this->merge(
            collect($this->all())->map(function ($value) {
                return is_string($value) ? Str::upper($value) : $value;
            })->toArray()
        );
    }

    public function messages(): array {
        return [
            "required" => "O preenchimento deste campo é obrigatório!",
            "max" => "Este campo possui tamanho máximo de [:max] caracteres!",
            "min" => "Este campo possui tamanho mínimo de [:min] caracteres!",
            "nome.unique" => "Já existe uma disciplina cadastrada com este nome!",
        ];
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array<string, ValidationRule|array<mixed>|string>
     */
}
```

```
*/  
public function rules(): array  
{  
    $id = $this->route('disciplina');  
  
    return [  
        'nome' => "required|max:100|min:8|unique:disciplinas,nome,{ $id }",  
        'carga_horaria' => "required",  
        'curso_id' => "required|exists:cursos,id",  
    ];  
}  
}
```

Arquivo: "app/Http/Requests/AlunoRequest.php"

```
PHP  
<?php  
  
namespace App\Http\Requests;  
  
use Illuminate\Contracts\Validation\ValidationRule;  
use Illuminate\Foundation\Http\FormRequest;  
use Illuminate\Support\Str;  
  
class AlunoRequest extends FormRequest  
{  
    /**  
     * Determine if the user is authorized to make this request.  
     */  
    public function authorize(): bool  
    {  
        return true;  
    }  
  
    protected function prepareForValidation() {  
  
        $this->merge(  
            collect($this->all())->map(function ($value) {  
                return is_string($value) ? Str::upper($value) : $value;  
            })->toArray()  
        );  
    }  
}
```

```
}

public function messages(): array {
    return [
        "required" => "O preenchimento deste campo é obrigatório!",
        "max" => "Este campo possui tamanho máximo de [:max] caracteres!",
        "min" => "Este campo possui tamanho mínimo de [:min] caracteres!",
    ];
}

/**
 * Get the validation rules that apply to the request.
 *
 * @return array<string, ValidationRule|array<mixed>|string>
 */
public function rules(): array
{
    return [
        'nome' => "required|max:100|min:8",
        'turma' => "required",
        'curso_id' => "required|exists:cursos,id",
    ];
}
}
```

Arquivo: "app/Http/Requests/MatriculaRequest.php"

```
PHP
<?php

namespace App\Http\Requests;

use Illuminate\Contracts\Validation\ValidationRule;
use Illuminate\Foundation\Http\FormRequest;

class MatriculaRequest extends FormRequest
{
    /**
     * Determine if the user is authorized to make this request.
     */
    public function authorize(): bool
    {
```

```
        return true;
    }

    public function messages(): array {
        return [
            "required" => "O preenchimento deste campo é obrigatório!",
        ];
    }

    /**
     * Get the validation rules that apply to the request.
     *
     * @return array<string, ValidationRule|array<mixed>|string>
     */
    public function rules(): array
    {
        return [
            'disciplina_id' => "required|exists:disciplinas,id",
            'aluno_id' => "required|exists:alunos,id",
        ];
    }
}
```

5) Adaptando: Curso Controller

Arquivo: "app/Http/Controllers/CursoController.php"

```
PHP
<?php

namespace App\Http\Controllers;

use App\Models\Curso;
use App\Http\Requests\CursoRequest; // Alterado

class CursoController extends Controller
{
    /**
     * Display a listing of the resource.
     */
}
```

```
public function index() {
    $data = Curso::with(['disciplina', 'aluno'])->orderBy('nome')->get();
    return view('curso.index', compact(['data']));
}

/**
 * Show the form for creating a new resource.
 */
public function create()
{
    return view('curso.create');
}

/**
 * Store a newly created resource in storage.
 */
public function store(CursoRequest $request) // Alterado
{
    Curso::create($request->validated());
    return redirect()->route('curso.index');
}

/**
 * Display the specified resource.
 */
public function show(string $id)
{
    $curso = Curso::find($id);

    if(isset($curso)) {
        return view('curso.show', compact(['curso']));
    }

    return "<h1>Curso não encontrado!</h1>";
}

/**
 * Show the form for editing the specified resource.
 */
public function edit(string $id)
{
    $curso = Curso::find($id);
```

```
    if(isset($curso)) {
        return view('curso.edit', compact(['curso']));
    }

    return "<h1>Curso não encontrado!</h1>";
}

/**
 * Update the specified resource in storage.
 */
public function update(CursoRequest $request, string $id) // Alterado
{
    $curso = Curso::find($id);

    if(isset($curso)) {
        $curso->update($request->validated());
        return redirect()->route('curso.index');
    }

    return "<h1>Curso não encontrado!</h1>";
}

/**
 * Remove the specified resource from storage.
 */
public function destroy(string $id)
{
    $curso = Curso::find($id);

    if(isset($curso)) {
        $curso->delete();
        return redirect()->route('curso.index');
    }

    return "<h1>Curso não encontrado!</h1>";
}
}
```

6) Criando | Codificando: Controllers (Disciplina, Aluno e Matrícula)

Comando

(no terminal de comando - dentro da pasta da aplicação)

Shell

```
docker-compose exec --user $(id -u):$(id -g) app php artisan
    make:controller DisciplinaController -r

docker-compose exec --user $(id -u):$(id -g) app php artisan
    make:controller AlunoController -r

docker-compose exec --user $(id -u):$(id -g) app php artisan
    make:controller MatriculaController -r
```

Rota

Arquivo: "routes/web.php"

PHP

<?php

```
use App\Http\Controllers\AlunoController; //Adicionado
use App\Http\Controllers\CursoController;
use App\Http\Controllers\DisciplinaController; //Adicionado
use App\Http\Controllers\MatriculaController; //Adicionado
use Illuminate\Support\Facades\Route;

Route::get('/', function () {
    return view('home'); // Alterado
})->name('home'); // Alterado

Route::resource('/curso', CursoController::class);
Route::resource('/disciplina', DisciplinaController::class); //Adicionado
Route::resource('/disciplina', AlunoController::class); //Adicionado
Route::resource('/disciplina', MatriculaController::class); //Adicionado
```

Views

Comando

(no terminal de comando - dentro da pasta da aplicação)

Shell

```
docker-compose exec --user $(id -u):$(id -g) app php artisan
    make:view disciplina/index
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:view disciplina/create
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:view disciplina/show
```

```
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:view disciplina/edit
```

Arquivo: "app/Http/Controllers/DisciplinaController.php"

```
PHP  
<?php  
  
namespace App\Http\Controllers;  
  
use App\Models\Disciplina;  
use App\Models\Curso;  
use App\Http\Requests\DisciplinaRequest;  
  
class DisciplinaController extends Controller  
{  
    /**  
     * Display a listing of the resource.  
     */  
    public function index()  
    {  
        $data = Disciplina::with(['curso'])->orderBy('nome')->get();  
        return view('disciplina.index', compact(['data']));  
    }  
  
    /**  
     * Show the form for creating a new resource.  
     */  
    public function create()  
    {  
        $cursos = Curso::all();  
        return view('disciplina.create', compact(['cursos']));  
    }  
}
```

```
/**
 * Store a newly created resource in storage.
 */
public function store(DisciplinaRequest $request)
{
    Disciplina::create($request->validated());
    return redirect()->route('disciplina.index');
}

/**
 * Display the specified resource.
 */
public function show(string $id)
{
    $disciplina = Disciplina::find($id);

    if(isset($disciplina)) {
        return view('disciplina.show', compact(['disciplina']));
    }

    return "<h1>Disciplina não encontrada!</h1>";
}

/**
 * Show the form for editing the specified resource.
 */
public function edit(string $id)
{
    $disciplina = Disciplina::find($id);
    $cursos = Curso::all();

    if(isset($disciplina)) {
        return view('disciplina.edit', compact(['disciplina', 'cursos']));
    }

    return "<h1>Disciplina não encontrada!</h1>";
}

/**
 * Update the specified resource in storage.
 */
public function update(DisciplinaRequest $request, string $id)
```

```
{
    $disciplina = Disciplina::find($id);

    if(isset($disciplina)) {
        $disciplina->update($request->validated());
        return redirect()->route('disciplina.index');
    }

    return "<h1>Disciplina não encontrada!</h1>";
}

/**
 * Remove the specified resource from storage.
 */
public function destroy(string $id)
{
    $disciplina = Disciplina::find($id);

    if(isset($disciplina)) {
        $disciplina->delete();
        return redirect()->route('disciplina.index');
    }

    return "<h1>Disciplina não encontrada!</h1>";
}
}
```

7) Criando uma View “Template” - Blade e Bootstrap

Comando

(no terminal de comando - dentro da pasta da aplicação)

Shell

```
docker-compose exec --user $(id -u):$(id -g) app php artisan
make:view template/main
```

Arquivo: “resources/views/template/main.blade.php”

PHP

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
```

```

<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/css/bootstrap.min.css">
<title>{{ $titulo }}</title>

<style>
    button {
        border: none;
        outline: none;
        background: none;
        cursor: pointer;
    }
    form {
        display: inline;
    }
</style>
</head>
<body>
    <nav class="navbar sticky-top navbar-expand-md navbar-dark bg-secondary">
        <div class="container-fluid">
            <a href="{{ route('home') }}" class="navbar-brand">
                <svg xmlns="http://www.w3.org/2000/svg" width="48" height="48" fill="#F99" class="bi bi-xbox"
viewBox="0 0 16 16">
                    <path d="M7.202 15.967a8 8 0 1-3.552-1.26c-.898-.585-1.101-.826-1.101-1.306 0-.965 1.062-2.656
2.879-4.583C6.459 7.723 7.897 6.44 8.052 6.475c.302.068 2.718 2.423 3.622 3.531 1.43 1.753 2.088 3.189 1.754
3.829-.254.486-1.83 1.437-2.987 1.802-.954.301-2.207.429-3.239.33m-5.866-3.57C5.89 11.253.212 10.127.03
8.497c-.06-.539-.038-.846.137-1.95.218-1.377 1.002-2.97 1.945-3.95.401-.417.437-.427.926-.263.595.2 1.23.638 2.213
1.5281.574.519-.313.385C4.056 6.553 2.52 9.086 1.94 10.653c-.315.852-.442 1.707-.306
2.063.091.24.007.15-.3-.319Zm13.101.195c.074-.36-.019-1.02-.238-1.687-.473-1.443-2.055-4.128-3.508-5.9531-.457-.575.494
-.454c.646-.593 1.095-.948 1.58-1.25.381-.237.927-.448 1.161-.448.145 0 .654.528 1.065 1.104a8.4 8.4 0 0 1 1.343
3.102c.153.728.166 2.286.024 3.012a9.5 9.5 0 0 1-.6 1.893c-.179.393-.624 1.156-.82 1.404-.1.128-.1.127-.043-.148ZM7.335
1.952c-.67-.34-1.704-.705-2.276-.803a4 4 0 0 0-.759-.043c-.471.024-.45 0 .306-.358A7.8 7.8 0 0 1 6.47.128c.8-.169
2.306-.17 3.094-.005.85.18 1.853.552
2.418.91.168.103-.385-.02c-.766-.038-1.88.27-3.078.853-.361.176-.676.316-.699.312a12 12 0 0 1-.654-.319Z"/>
                </svg>
                <span class="ms-2 fs-4 fw-bold">Sistema Aula</span>
            </a>
            <button class="navbar-toggler" data-bs-toggle="collapse" data-bs-target="#itens">
                <svg xmlns="http://www.w3.org/2000/svg" width="30" height="30" fill="currentColor" class="bi
bi-menu-button-wide" viewBox="0 0 16 16">
                    <path d="M0 1.5A1.5 1.5 0 0 1 1.5 0h13A1.5 1.5 0 0 1 16 1.5v2A1.5 1.5 0 0 1 14.5 5h-13A1.5 1.5 0 0
1 0 3.5v-2zM1.5 1a.5.5 0 0 0-.5.5v2a.5.5 0 0 0 .5.5h13a.5.5 0 0 0 1 0h-3a.5.5 0 0 1-.5-.5v-2a.5.5 0 0 0-.5-.5h-13z"/>
                    <path d="M2 2.5a.5.5 0 0 1 .5-.5h3a.5.5 0 0 1 0 1h-3a.5.5 0 0 1-.5-.5zm10.823.323-.396-.396A.25.25
0 0 1 12.604 2h.792a.25.25 0 0 1 .177.4271-.396.396a.25.25 0 1-.354 0zM8a2 2 0 0 1 2-2h12a2 2 0 0 1 2 2v5a2 2 0 0
1-2 2H2a2 2 0 0 1-2-2V8zm1 3v2a1 1 0 0 1 1h12a1 1 0 0 0 1-1v-2H1zm14-1V8a1 1 0 0 0-1-1H2a1 1 0 0 0-1 1v2h14z" data-bbox="134 612 864 665"/>
                </svg>
            </button>
            <div class="collapse navbar-collapse" id="itens">
                <ul class="navbar-nav ms-auto">
                    <li class="nav-item me-2">
                        <a href="{{ route('aluno.index') }}" class="nav-link">
                            <svg xmlns="http://www.w3.org/2000/svg" width="30" height="30" fill="#FFF" class="bi
bi-mortarboard-fill" viewBox="0 0 16 16">
                                <path d="M8.211 2.047a.5.5 0 0 0-.422 0l-7.5 3.5a.5.5 0 0 0 .025.91717.5 3a.5.5 0 0 0
.372 0L14 7.14V13a1 1 0 0 0-1 1v2h3v-2a1 1 0 0 0-1-1V6.7391.686-.275a.5.5 0 0 0 .025-.917z"/>
                                <path d="M4.176 9.032a.5.5 0 0 0-.656.3271-.5 1.7a.5.5 0 0 0 .294.60514.5 1.8a.5.5 0 0
0 .372 0L14.5-1.8a.5.5 0 0 0 .294-.6051-.5-1.7a.5.5 0 0 0-.656-.32718 10.466z"/>
                            </svg>
                            <span class="ps-1 text-white">Alunos</span>
                        </a>
                    </li>
                    <li class="nav-item me-2">
                        <a href="{{ route('curso.index') }}" class="nav-link">
                            <svg xmlns="http://www.w3.org/2000/svg" width="30" height="30" fill="#FFF" class="bi
bi-easel2-fill" viewBox="0 0 16 16">

```

```
<path d="M8.447.276a.5.5 0 0 0-.894 0L7.19 1H2.5A1.5 1.5 0 0 1 2.5V10h14V2.5A1.5 1.5 0 0 13.5 1H8.809z"/>
<path fill-rule="evenodd" d="M.5 11a.5.5 0 0 0 1h2.861-.845 3.379a.5.5 0 0 0 .97.242L3.89 14h8.221.405 1.621a.5.5 0 0 0 .97-.242L12.64 12h2.86a.5.5 0 0 0 0-1zm3.64 2 .25-1h7.221.25 1z"/>
</svg>
<span class="ps-1 text-white">Cursos</span>
</a>
</li>
<li class="nav-item me-2">
<a href="{ route('disciplina.index') }" class="nav-link">
<svg xmlns="http://www.w3.org/2000/svg" width="30" height="30" fill="#FFF" class="bi bi-stickies-fill" viewBox="0 0 16 16">
<path d="M0 1.5V13a1 1 0 0 1 1V1.5a.5.5 0 0 1 .5-.5H14a1 1 0 0 1-1H1.5A1.5 1.5 0 0 0 0 1.5"/>
<path d="M3.5 2A1.5 1.5 0 0 2 3.5v11A1.5 1.5 0 0 3.5 16h6.086a1.5 1.5 0 0 1 1.06-.4414.915-4.914A1.5 1.5 0 0 16 9.586V3.5A1.5 1.5 0 0 14.5 2zm6 8.5a1 1 0 0 1 1-1h4.396a.25.25 0 0 1 .177.4271-5.146 5.146a.25.25 0 1-.427-.177z"/>
</svg>
<span class="ps-1 text-white">Disciplinas</span>
</a>
</li>
<li class="nav-item me-2">
<a href="{ route('matricula.index') }" class="nav-link">
<svg xmlns="http://www.w3.org/2000/svg" width="30" height="30" fill="#FFF" class="bi bi-ui-checks" viewBox="0 0 16 16">
<path d="M7 2.5a.5.5 0 0 1 .5-.5h7a.5.5 0 0 1 .5v1a.5.5 0 0 1-.5h-7a.5.5 0 0 1-.5-.5zM2 1a2 2 0 0 2 2v2a2 2 0 0 2 2h2a2 2 0 0 2-2V3a2 2 0 0 2-2zm0 8a2 2 0 0 2 2v2a2 2 0 0 2 2h2a2 2 0 0 2-2v-2a2 2 0 0 2-2zm.854-3.646a.5.5 0 1-.708 01-1a.5.5 0 1 1 .708-.7081.646.647 1.646-1.647a.5.5 0 1 1 .708.708zm0 8a.5.5 0 1-.708 01-1a.5.5 0 1 1 .708-.7081.646.647 1.646-1.647a.5.5 0 1 1 .708.708zM7 10.5a.5.5 0 0 1 .5-.5h7a.5.5 0 0 1 .5v1a.5.5 0 0 1-.5h-7a.5.5 0 0 1-.5-.5zm0-5a.5.5 0 0 1 .5-.5h5a.5.5 0 0 1 0 1h-5a.5.5 0 0 1-.5-.5"/>
</svg>
<span class="ps-1 text-white">Matrículas</span>
</a>
</li>
<li class="nav-item dropdown pe-3">
<a class="nav-link dropdown-toggle" data-bs-toggle="dropdown" href="#">
<svg xmlns="http://www.w3.org/2000/svg" width="36" height="36" fill="#FFF" class="bi bi-person-circle" viewBox="0 0 16 16">
<path d="M11 6a3 3 0 1 1-6 0 3 3 0 0 1 6 0z"/>
<path fill-rule="evenodd" d="M0 8a8 8 0 1 1 16 0A8 8 0 1 0 8zm8-7a7 7 0 0 0-5.468 11.37C3.242 11.226 4.805 10 8 10s4.757 1.225 5.468 2.37A7 7 0 0 8 1z"/>
</svg>
<span class="ps-1 text-white">Visitante</span>
</a>
<ul class="dropdown-menu">
<li>
<a href="" class="dropdown-item">
<svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="#777" class="bi bi-key-fill" viewBox="0 0 16 16">
<path d="M3.5 11.5a3.5 3.5 0 1 1 3.163-5H14L15.5 8 14 9.51-1-1 1-1-1 1-1-1 1H6.663a3.5 3.5 0 0 1-3.163 2M2.5 9a1 1 0 1 0 0-2 1 1 0 0 0 2"/>
</svg>
<span class="ps-1 text-secondary ">Alterar Senha</span>
</li>
</li>
<li>
<a href="" class="dropdown-item">
<svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="#777" class="bi bi-door-open" viewBox="0 0 16 16">
<path d="M8.5 10c-.276 0-.5-.448-.5-1s.224-1 .5-1.5448.5 1-.224 1-.5 1"/>
<path d="M10.828.122A.5.5 0 0 1 11 .5V1h.5A1.5 1.5 0 0 1 13 2.5V15h1.5a.5.5 0 0 1 0 1h-13a.5.5 0 0 1 0-1H3V1.5a.5.5 0 0 1 .43-.49517-1a.5.5 0 0 1 .398.117M11.5 2H11v13h1V2.5a.5.5 0 0 0-.5-.5M4 1.934V15h6V1.077z"/>
</svg>
<span class="ps-1 text-secondary ">Sair</span>
</li>
</li>
</ul>
```

```

        </li>
      </ul>
    </div>
  </div>
</nav>
<div class="container py-4">
  <div class="row">
    <div class="col">
      @if($cabecalho != '')
        <h3 class="text-secondary"><b>{{ $cabecalho }}</b></h3>
      @endif
    </div>
    <div class="col d-flex justify-content-end">
      @if($rota != '')
        <a href= "{{ route($rota) }}" class="btn btn-secondary ms-2">
          <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="#FFF" class="bi
bi-plus-circle-fill" viewBox="0 0 16 16">
            <path d="M16 8A8 8 0 1 1 0 8a8 8 0 1 16 0zM8.5 4.5a.5.5 0 0 0-1 0v3h-3a.5.5 0 0 0
1h3v3a.5.5 0 0 1 0v-3h3a.5.5 0 0 0 0-1h-3v-3z"/>
          </svg>
        </a>
      @endif
    </div>
  </div>
<hr>
@yield('conteudo')
</div>
<nav class="navbar fixed-bottom navbar-dark bg-secondary">
  <div class="container-fluid">
    <span class="text-white">&copy; Todos os direitos reservados.</span>
  </div>
</nav>
</body>

<div class="modal fade" tabindex="-1" id="removeModal">
  <div class="modal-dialog">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title text-danger">Operação de Remoção</h5>
        <button type="button" class="btn-close" data-bs-dismiss="removeModal" onclick="closeRemoveModal()"
aria-label="Close"></button>
      </div>
      <input type="hidden" id="id_remove">
      <div class="modal-body text-secondary">
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-secondary btn-block align-content-center"
onclick="closeRemoveModal()">
          <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi
bi-arrow-left-square-fill" viewBox="0 0 16 16">
            <path d="M16 14a2 2 0 0 1-2 2H2a2 2 0 0 1-2-2V2a2 2 0 0 1 2-2h12a2 2 0 0 1 2
2v12zm-4.5-6.5H5.70712.147-2.146a.5.5 0 1 0-.708-.708l-3 3a.5.5 0 0 0 .708l3 3a.5.5 0 0 0
.708l3 3a.5.5 0 0 0 .708-.708L5.707
8.5H11.5a.5.5 0 0 0-1z"/>
          </svg>
          &nbsp; Não
        </button>
        <button type="button" class="btn btn-danger" onclick="remove()">
          Sim &nbsp;
          <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi
bi-check-circle-fill" viewBox="0 0 16 16">
            <path d="M16 8A8 8 0 1 1 0 8a8 8 0 1 16 0zm-3.97-3.03a.75.75 0 0 0-1.08.022L7.477 9.417 5.384
7.323a.75.75 0 0 0-1.06 1.06L6.97 11.03a.75.75 0 0 1 1.079-.0213.992-4.99a.75.75 0 0 0-.01-1.05z"/>
          </svg>
        </button>
      </div>
    </div>
  </div>
</div>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/js/bootstrap.bundle.min.js"></script>

```

```
@yield('script')  
</html>
```

8) Criando a View “Home” e Definindo uma Rota

Comando

(no terminal de comando - dentro da pasta da aplicação)

```
Shell  
docker-compose exec --user $(id -u):$(id -g) app php artisan  
make:view home
```

Arquivo: “resources/views/home.blade.php”

```
PHP  
@extends('template/main',  
    [  
        'titulo'=>"Sistema Aula",  
        'cabecalho' => 'Home',  
        'rota' => '',  
    ]  
)  
  
@section('conteudo')  
  
@endsection
```

9) Reescrevendo as Views de Curso (CRUD)

Arquivo: “resources/views/curso/index.blade.php”

```
PHP  
@extends('template/main',  
    [  
        'titulo'=>"Sistema Aula",  
        'cabecalho' => 'Lista de Cursos',  
        'rota' => 'curso.create',  
    ]  
)  
@section('conteudo')  
  
    <table class="table align-middle caption-top table-striped">
```

```

<thead>
  <th class="text-secondary">NOME</th>
  <th class="d-none d-md-table-cell text-secondary">DURAÇÃO</th>
  <th class="text-secondary">AÇÕES</th>
</thead>
<tbody>
  @foreach ($data as $item)
    <tr>
      <td>{{ $item->nome }}</td>
      <td class="d-none d-md-table-cell">{{ $item->nome }}</td>
      <td>
        <a href="{{route('curso.show', $item->id)}}" class="btn
btn-outline-info">
          <svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" fill="#0d6efd" class="bi bi-info-circle-fill" viewBox="0 0 16 16">
            <path d="M8 16A8 8 0 1 0 8 0a8 8 0 0 0 16m.93-9.412-1
4.705c-.07.34.029.533.304.533.194
.487-.07.686-.246l-.088.416c-.287.346-.92.598-1.465.598-.703
0-1.002-.422-.808-1.319l.738-3.468c.064-.293.006-.399-.287-.471-.451-.081.082-.381
2.29-.287zM8 5.5a1 1 0 1 1 0 2 1 1 0 0 1 0 2"/>
          </svg>
        </a>
        <a href="{{route('curso.edit', $item->id)}}" class="btn
btn-outline-success">
          <svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" fill="#5cb85c" class="bi bi-arrow-counterclockwise" viewBox="0 0 16 16">
            <path fill-rule="evenodd" d="M8 3a5 5 0 1 1-4.546
2.914.5.5 0 0 0-.908-.417A6 6 0 1 0 8 2v1z"/>
            <path d="M8 4.466V.534a.25.25 0 0 0-.41-.192L5.23
2.308a.25.25 0 0 0 .384l2.36 1.966A.25.25 0 0 0 8 4.466z"/>
          </svg>
        </a>
        <form action="{{route('curso.destroy', $item->id)}}"
method="POST" id="form_{{ $item->id }}">
          @csrf
          @method('delete')
          <button type="submit" class="btn btn-outline-danger">
            <svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" fill="#d9534f" class="bi bi-trash-fill" viewBox="0 0 16 16">
              <path d="M2.5 1a1 1 0 0 0-1 1v1a1 1 0 0 1 1H3v9a2 2
0 0 0 2 2h6a2 2 0 0 0 2-2V4h.5a1 1 0 0 0 1-1V2a1 1 0 0 0-1-1H10a1 1 0 0 0-1-1H7a1 1 0 0
0-1 1H2.5zm3 4a.5.5 0 0 1 .5.5v7a.5.5 0 0 1-1 0v-7a.5.5 0 0 1-.5-.5zM8 5a.5.5 0 0 1
.5.5v7a.5.5 0 0 1-1 0v-7a.5.5 0 0 1-.5-.5zM8 5a.5.5 0 0 1 1 0v-7a.5.5 0 0 1 1 0z"/>
            </svg>
          </button>
        </form>
      </td>
    </tr>
  @endforeach
</tbody>
</table>

```

@endsection

Arquivo: "resources/views/curso/create.blade.php"

PHP

```
@extends('template/main',
    [
        'titulo'=>"Sistema Aula",
        'cabecalho' => 'Novo Curso',
        'rota' => '',
    ]
)
@section('conteudo')
    <form action="{{route('curso.store')}}" method="POST">
        @csrf
        <div class="row">
            <div class="col" >
                <div class="form-floating mb-3">
                    <input
                        type="text"
                        class="form-control @error('nome') is-invalid @enderror"
                        name="nome"
                        placeholder="Nome"
                        value="{{old('nome')}}"
                    />
                    <label for="nome">Nome</label>

                    @if($errors->has('nome'))
                        <div class='invalid-feedback'>
                            {{ $errors->first('nome') }}
                        </div>
                    @endif
                </div>
            </div>
            <div class="row">
                <div class="col" >
                    <div class="form-floating mb-3">
                        <input
                            type="number"
                            class="form-control @error('duracao') is-invalid @enderror"
                            name="duracao"
                            placeholder="Tempo de Duração"
                            value="{{old('duracao')}}"
                        />
                        <label for="duracao">Duração</label>

                        @if($errors->has('duracao'))
                            <div class='invalid-feedback'>
                                {{ $errors->first('duracao') }}
                            </div>
                        @endif
                    </div>
                </div>
            </div>
        </div>
    </form>
</div>
```

```

        @endif
    </div>
</div>
</div>

<div class="row mb-5">
    <div class="col">
        <a href="{{route('curso.index')}}" class="btn btn-secondary btn-block align-content-center">
            <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-arrow-left-square-fill" viewBox="0 0 16 16">
                <path d="M16 14a2 2 0 0 1-2 2H2a2 2 0 0 1-2-2V2a2 2 0 0 1 2-2h12a2 2 0 0 1 2 2v12zm-4.5-6.5H5.70712.147-2.146a.5.5 0 1 0-.708-.708l-3 3a.5.5 0 0 0 .708l3 3a.5.5 0 0 0 .708L5.707 8.5H11.5a.5.5 0 0 0 0-1z"/>
            </svg>
            &nbsp; Voltar
        </a>
        <button type="submit" class="btn btn-success btn-block align-content-center">
            Confirmar &nbsp;
            <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-check-circle-fill" viewBox="0 0 16 16">
                <path d="M16 8A8 8 0 1 1 0 8a8 8 0 0 1 16 0zm-3.97-3.03a.75.75 0 0 1-.08.022L7.477 9.417 5.384 7.323a.75.75 0 0 0-1.06 1.06L6.97 11.03a.75.75 0 0 0 1.079-.0213.992-4.99a.75.75 0 0 0-.01-1.05z"/>
            </svg>
        </button>
    </div>
</div>
</form>
@endsection

```

Arquivo: "resources/views/curso/show.blade.php"

```

PHP
@extends('template/main',
    [
        'titulo'=>"Sistema Aula",
        'cabecalho' => 'Detalhes do Curso',
        'rota' => '',
    ]
)
@section('conteudo')

<div class="row">
    <div class="col" >
        <div class="form-floating mb-3">
            <input
                type="text"
                class="form-control"
                value="{{ $curso->nome }}"
            >
        </div>
    </div>
</div>

```

```

        disabled
    />
<label for="nome">Nome</label>

@if($errors->has('nome'))
    <div class='invalid-feedback'>
        {{ $errors->first('nome') }}
    </div>
@endif
</div>
</div>
</div>
<div class="row">
    <div class="col" >
        <div class="form-floating mb-3">
            <input
                type="number"
                class="form-control"
                value="{{ $curso->duracao }}"
                disabled
            />
            <label for="duracao">Duração</label>
        </div>
    </div>
</div>

<div class="row mb-5">
    <div class="col">
        <a href="{{route('curso.index')}}" class="btn btn-secondary btn-block align-content-center">
            <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24" fill="currentColor" class="bi bi-arrow-left-square-fill" viewBox="0 0 16 16">
                <path d="M16 14a2 2 0 0 1-2 2H2a2 2 0 0 1-2-2V2a2 2 0 0 1 2-2h12a2 2 0 0 1 2 2v12zm-4.5-6.5H5.707l2.147-2.146a.5.5 0 1 0-.708-.708l-3 3a.5.5 0 0 0 .708l3 3a.5.5 0 0 0 .708-.708L5.707 8.5H11.5a.5.5 0 0 0-.12"/>
            </svg>
            &nbsp; Voltar
        </a>
    </div>
</div>
@endsection

```

Arquivo: "resources/views/curso/edit.blade.php"

```

PHP
@extends('template/main',
    [
        'titulo'=>"Sistema Aula",
        'cabecalho' => 'Alterar Curso',
    ]
)

```

```
        'rota' => '',
    ]
)
@section('conteudo')
    <form action="{{route('curso.update', $curso->id)}}" method="POST">
        @csrf
        @method('PUT')
        <div class="row">
            <div class="col" >
                <div class="form-floating mb-3">
                    <input
                        type="text"
                        class="form-control @error('nome') is-invalid @enderror"
                        name="nome"
                        placeholder="Nome"
                        value="{{ $curso->nome }}"
                    />
                    <label for="nome">Nome</label>

                    @if($errors->has('nome'))
                        <div class='invalid-feedback'>
                            {{ $errors->first('nome') }}
                        </div>
                    @endif
                </div>
            </div>
            <div class="row">
                <div class="col" >
                    <div class="form-floating mb-3">
                        <input
                            type="number"
                            class="form-control @error('duracao') is-invalid @enderror"
                            name="duracao"
                            placeholder="Tempo de Duração"
                            value="{{ $curso->duracao }}"
                        />
                        <label for="duracao">Duração</label>

                        @if($errors->has('duracao'))
                            <div class='invalid-feedback'>
                                {{ $errors->first('duracao') }}
                            </div>
                        @endif
                    </div>
                </div>
            </div>
            <div class="row mb-5">
                <div class="col">
                    <a href="{{route('curso.index')}}" class="btn btn-secondary btn-block align-content-center">
```

```

                <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24"
fill="currentColor" class="bi bi-arrow-left-square-fill" viewBox="0 0 16 16">
                    <path d="M16 14a2 2 0 0 1-2 2H2a2 2 0 0 1-2-2V2a2 2 0 0 1
2-2h12a2 2 0 0 1 2 2v12zm-4.5-6.5H5.70712.147-2.146a.5.5 0 1 0-.708-.7081-3 3a.5.5 0 0 0
.70813 3a.5.5 0 0 0 .708-.708L5.707 8.5H11.5a.5.5 0 0 0 0-1z"/>
                </svg>
                &nbsp; Voltar
            </a>

                <button type="submit" class="btn btn-success btn-block
align-content-center">
                    Confirmar &nbsp;
                    <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24"
fill="currentColor" class="bi bi-check-circle-fill" viewBox="0 0 16 16">
                        <path d="M16 8A8 8 0 1 1 0 8a8 8 0 1 16 0zm-3.97-3.03a.75.75 0
0 0-1.08.022L7.477 9.417 5.384 7.323a.75.75 0 0 0-1.06 1.06L6.97 11.03a.75.75 0 0 0
1.079-.0213.992-4.99a.75.75 0 0 0-.01-1.05z"/>
                    </svg>
                </button>
            </div>
        </div>
    </form>
@endsection

```

10) Codificando: Views (Disciplina)

Arquivo: "resources/views/disciplina/index.blade.php"

```

PHP
@extends('template/main',
    [
        'titulo'=>"Sistema Aula",
        'cabecalho' => 'Lista de Disciplinas',
        'rota' => 'disciplina.create',
    ]
)
@section('conteudo')

    <table class="table align-middle caption-top table-striped">
        <thead>
            <th class="text-secondary">NOME</th>
            <th class="d-none d-md-table-cell text-secondary">CARGA HORÁRIA</th>
            <th class="d-none d-md-table-cell text-secondary">CURSO</th>
            <th class="text-secondary">AÇÕES</th>
        </thead>
        <tbody>
            @foreach ($data as $item)
                <tr>
                    <td>{{ $item->nome }}</td>
                    <td class="d-none d-md-table-cell">{{ $item->carga_horaria }}
aula(s)</td>

```

```

<td class="d-none d-md-table-cell">{{ $item->curso->nome }}</td>
<td>
    <a href="{{route('disciplina.show', $item->id)}}" class="btn
btn-outline-info">
        <svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" fill="#0d6efd" class="bi bi-info-circle-fill" viewBox="0 0 16 16">
            <path d="M8 16A8 8 0 1 0 8 0a8 8 0 0 0 16m.93-9.412-1
4.705c-.07.34.029.533.304.533.194
.487-.07.686-.246l-.088.416c-.287.346-.92.598-1.465.598-.703
0-1.002-.422-.808-1.319l.738-3.468c.064-.293.006-.399-.287-.471-.451-.081.082-.381
2.29-.287zM8 5.5a1 1 0 1 1 0 2 1 1 0 0 1 0 2"/>
        </svg>
    </a>
    <a href="{{route('disciplina.edit', $item->id)}}" class="btn
btn-outline-success">
        <svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" fill="#5cb85c" class="bi bi-arrow-counterclockwise" viewBox="0 0 16 16">
            <path fill-rule="evenodd" d="M8 3a5 5 0 1 1-4.546
2.914.5.5 0 0 0-.908-.417A6 6 0 1 0 8 2v1z"/>
            <path d="M8 4.466V.534a.25.25 0 0 0-.41-.192L5.23
2.308a.25.25 0 0 0 .384l2.36 1.966A.25.25 0 0 0 8 4.466z"/>
        </svg>
    </a>
    <form action="{{route('disciplina.destroy', $item->id)}}"
method="POST" id="form_{{ $item->id }}">
        @csrf
        @method('delete')
        <button type="submit" class="btn btn-outline-danger">
            <svg xmlns="http://www.w3.org/2000/svg" width="24"
height="24" fill="#d9534f" class="bi bi-trash-fill" viewBox="0 0 16 16">
                <path d="M2.5 1a1 1 0 0 0-1 1v1a1 1 0 0 1 1H3v9a2 2
0 0 0 2 2h6a2 2 0 0 0 2-2V4h.5a1 1 0 0 0 1-1V2a1 1 0 0 0-1-1H10a1 1 0 0 0-1-1H7a1 1 0 0
0-1 1H2.5zm3 4a.5.5 0 0 1 .5.5v7a.5.5 0 0 1-1 0v-7a.5.5 0 0 1-.5-.5zM8 5a.5.5 0 0
1 .5.5v7a.5.5 0 0 1-1 0v-7a.5.5 0 0 1-.5-.5zM8 5a.5.5 0 0 1 0 2 1 1 0 0 1 0 2"/>
            </svg>
        </button>
    </form>
</td>
</tr>
@endforeach
</tbody>
</table>

@endsection

```

Arquivo: "resources/views/disciplina/create.blade.php"

```

PHP
@extends('template/main',
[
    'titulo'=>"Sistema Aula",

```

```
'cabecalho' => 'Nova Disciplina',
'rota' => '',
]
)
@section('conteudo')
<form action="{{route('disciplina.store')}}" method="POST">
  @csrf
  <div class="row">
    <div class="col" >
      <div class="form-floating mb-3">
        <input
          type="text"
          class="form-control @error('nome') is-invalid @enderror"
          name="nome"
          placeholder="Nome"
          value="{{old('nome')}}"
        />
        <label for="nome">Nome</label>

        @if($errors->has('nome'))
          <div class='invalid-feedback'>
            {{ $errors->first('nome') }}
          </div>
        @endif
      </div>
    </div>
    <div class="row">
      <div class="col" >
        <div class="form-floating mb-3">
          <input
            type="number"
            class="form-control @error('carga_horaria') is-invalid @enderror"
            name="carga_horaria"
            placeholder="Carga Horária"
            value="{{old('carga_horaria')}}"
          />
          <label for="carga_horaria">Carga Horária</label>

          @if($errors->has('carga_horaria'))
            <div class='invalid-feedback'>
              {{ $errors->first('carga_horaria') }}
            </div>
          @endif
        </div>
      </div>
    </div>
    <div class="row">
      <div class="col">
        <div class="input-group mb-3">
          <span class="input-group-text bg-secondary text-white">Curso</span>
          <select
```



```
[
  'titulo'=>"Sistema Aula",
  'cabecalho' => 'Alterar Disciplina',
  'rota' => '',
]
)
@section('conteudo')
<form action="{{route('disciplina.update', $disciplina->id)}}" method="POST">
  @csrf
  @method('PUT')
  <div class="row">
    <div class="col" >
      <div class="form-floating mb-3">
        <input
          type="text"
          class="form-control @error('nome') is-invalid @enderror"
          name="nome"
          placeholder="Nome"
          value="{{ $disciplina->nome }}"
        />
        <label for="nome">Nome</label>

        @if($errors->has('nome'))
          <div class='invalid-feedback'>
            {{ $errors->first('nome') }}
          </div>
        @endif
      </div>
    </div>
    <div class="row">
      <div class="col" >
        <div class="form-floating mb-3">
          <input
            type="number"
            class="form-control @error('carga_horaria') is-invalid @enderror"
            name="carga_horaria"
            placeholder="Carga Horária"
            value="{{ $disciplina->carga_horaria }}"
          />
          <label for="carga_horaria">Carga Horária</label>

          @if($errors->has('carga_horaria'))
            <div class='invalid-feedback'>
              {{ $errors->first('carga_horaria') }}
            </div>
          @endif
        </div>
      </div>
    </div>
  </div>
  <div class="row">
```


Arquivo: "resources/views/disciplina/edit.blade.php"

PHP

```
@extends('template/main',
[
    'titulo'=>"Sistema Aula",
    'cabecalho' => 'Alterar Disciplina',
    'rota' => '',
])
@section('conteudo')
<form action="{{route('disciplina.update', $disciplina->id)}}" method="POST">
    @csrf
    @method('PUT')
    <div class="row">
        <div class="col" >
            <div class="form-floating mb-3">
                <input
                    type="text"
                    class="form-control @error('nome') is-invalid @enderror"
                    name="nome"
                    placeholder="Nome"
                    value="{{ $disciplina->nome }}"
                />
                <label for="nome">Nome</label>

                @if($errors->has('nome'))
                    <div class='invalid-feedback'>
                        {{ $errors->first('nome') }}
                    </div>
                @endif
            </div>
        </div>
    </div>
    <div class="row">
        <div class="col" >
            <div class="form-floating mb-3">
                <input
                    type="number"
                    class="form-control @error('carga_horaria') is-invalid @enderror"
                    name="carga_horaria"
                    placeholder="Carga Horária"
                    value="{{ $disciplina->carga_horaria }}"
                />
                <label for="carga_horaria">Carga Horária</label>

                @if($errors->has('carga_horaria'))
                    <div class='invalid-feedback'>
                        {{ $errors->first('carga_horaria') }}
                    </div>
                @endif
            </div>
        </div>
    </div>
</div>
```


VISUALIZANDO A APLICAÇÃO CRIADA

Navegador: <http://localhost:15000>

HOME

 **Sistema Aula**  Alunos  Cursos  Disciplinas  Matrículas  Visitante ▾

Home

CURSO - INDEX


 **Sistema Aula**  Alunos  Cursos  Disciplinas  Matrículas  Visitante ▾

Lista de Cursos











| NOME | DURAÇÃO | AÇÕES |
|------------------------------|----------|---|
| TÉCNICO EM INFORMÁTICA | 4 ano(s) |    |
| TECNÓLOGO EM DESENVOLVIMENTO | 3 ano(s) |    |

DISCIPLINA INDEX

 **Sistema Aula**  Alunos  Cursos  Disciplinas  Matrículas  Visitante ▾

Lista de Disciplinas



| NOME | CARGA HORÁRIA | CURSO | AÇÕES |
|-------------------|---------------|------------------------------|---|
| BANCO DE DADOS II | 4 aula(s) | TECNÓLOGO EM DESENVOLVIMENTO |    |
| FÍSICA | 2 aula(s) | TÉCNICO EM INFORMÁTICA |    |
| LÍNGUA PORTUGUESA | 2 aula(s) | TÉCNICO EM INFORMÁTICA |    |

