

## Atividade IV

Aluno: Clemente da Silva Sepulveda Neto  
Disciplina: Laboratório de Programação Web II

- Rotas

```
1
2 Route::get('/', function () { return view('index'); });
3
4 Route::prefix('/clientes')->group(function() {
5     Route::get('/index', [ClienteController::class, 'index'])->name('clientes.index');
6     Route::get('/create', [ClienteController::class, 'create'])->name('clientes.create');
7     Route::post('/store', [ClienteController::class, 'store'])->name('clientes.store');
8     Route::get('/edit/{id}', [ClienteController::class, 'edit'])->name('clientes.edit');
9     Route::put('/update/{id}', [ClienteController::class, 'update'])->name('clientes.update');
10    Route::delete('/delete/{id}', [ClienteController::class, 'destroy'])->name('clientes.destroy');
11 });
12
13 Route::prefix('/fornecedores')->group(function() {
14     Route::get('/index', [FornecedorController::class, 'index'])->name('fornecedores.index');
15     Route::get('/create', [FornecedorController::class, 'create'])->name('fornecedores.create');
16     Route::post('/store', [FornecedorController::class, 'store'])->name('fornecedores.store');
17     Route::get('/edit/{id}', [FornecedorController::class, 'edit'])->name('fornecedores.edit');
18     Route::put('/update/{id}', [FornecedorController::class, 'update'])->name('fornecedores.update');
19     Route::delete('/delete/{id}', [FornecedorController::class, 'destroy'])->name('fornecedores.destroy');
20 });
21
22 Route::prefix('/produtos')->group(function() {
23     Route::get('/index', [ProdutoController::class, 'index'])->name('produtos.index');
24     Route::get('/create', [ProdutoController::class, 'create'])->name('produtos.create');
25     Route::post('/store', [ProdutoController::class, 'store'])->name('produtos.store');
26     Route::get('/edit/{id}', [ProdutoController::class, 'edit'])->name('produtos.edit');
27     Route::put('/update/{id}', [ProdutoController::class, 'update'])->name('produtos.update');
28     Route::delete('/delete/{id}', [ProdutoController::class, 'destroy'])->name('produtos.destroy');
29 });
30
31 Route::prefix('/vendas')->group(function() {
32     Route::get('/index', [VendaController::class, 'index'])->name('vendas.index');
33     Route::get('/create', [VendaController::class, 'create'])->name('vendas.create');
34     Route::post('/store', [VendaController::class, 'store'])->name('vendas.store');
35     Route::get('/edit/{id}', [VendaController::class, 'edit'])->name('vendas.edit');
36     Route::put('/update/{id}', [VendaController::class, 'update'])->name('vendas.update');
37     Route::delete('/delete/{id}', [VendaController::class, 'destroy'])->name('vendas.destroy');
38 });
39
40 Route::prefix('/compras')->group(function() {
41     Route::get('/index', [CompraController::class, 'index'])->name('compras.index');
42     Route::get('/create', [CompraController::class, 'create'])->name('compras.create');
43     Route::post('/store', [CompraController::class, 'store'])->name('compras.store');
44     Route::get('/edit/{id}', [CompraController::class, 'edit'])->name('compras.edit');
45     Route::put('/update/{id}', [CompraController::class, 'update'])->name('compras.update');
46     Route::delete('/delete/{id}', [CompraController::class, 'destroy'])->name('compras.destroy');
47 });
```

- Controladores

### *Controller Fornecedor*

```
1
2 class FornecedorController extends Controller
3 {
4     public function index() {
5         return view('fornecedores.index', ['fornecedores' => Fornecedor::all()]);
6     }
7
8     public function create() {
9         return view('fornecedores.create');
10    }
11
12    public function store(Request $request) {
13        $fornecedor = new Fornecedor();
14        $fornecedor->nome = $request->nome;
15        $fornecedor->telefone = $request->telefone;
16        $fornecedor->endereco = $request->endereco;
17        $fornecedor->save();
18
19        return redirect('/fornecedores/index');
20    }
21
22    public function show($id) {
23        //
24    }
25
26    public function edit($id) {
27        $fornecedor = Fornecedor::find($id);
28        return view('fornecedores.edit', ['fornecedor' => $fornecedor]);
29    }
30
31    public function update(Request $request, $id) {
32        Fornecedor::find($id)->update($request->all());
33        return redirect('/fornecedores/index');
34    }
35
36    public function destroy($id) {
37        Fornecedor::find($id)->delete();
38        return redirect('/fornecedores/index');
39    }
40 }
```

## Controller Compra

```
1 class CompraController extends Controller
2 {
3
4     public function index() {
5         return view('compras.index', [
6             'compras' => Compra::all()
7         ]);
8     }
9
10    public function create() {
11        return view('compras.create', [
12            'produtos' => Produto::all(),
13            'fornecedores' => Fornecedor::all()
14        ]);
15    }
16
17    public function store(Request $request) {
18
19        // concertar para varios produtos
20        $compra = new Compra();
21        $compra->fornecedor_id = $request->fornecedor_id;
22        $compra->valor_total = Produto::find($request->produto_id[0])->valor_compra * $request->quantidade[0];
23        $compra->save();
24
25        for ($i = 0; $i < count($request->quantidade); $i++) {
26            $itensCompra = new ItemCompra();
27            $itensCompra->compra_id = $compra->id;
28            $itensCompra->produto_id = $request->produto_id[$i];
29            $itensCompra->quantidade = $request->quantidade[$i];
30            $itensCompra->valor_unitario = Produto::find($request->produto_id[$i])->valor_compra;
31            $itensCompra->valor_total = $itensCompra->valor_unitario * $itensCompra->quantidade;
32            $itensCompra->save();
33        }
34
35        return redirect()->route('compras.index');
36    }
37
38    public function show($id) {
39        //
40    }
41
42    public function edit($id) {
43        //
44    }
45
46    public function update(Request $request, $id) {
47        //
48    }
49
50    public function destroy($id) {
51        Compra::destroy($id);
52        return redirect()->route('compra.index');
53    }
54 }
```

## Controller Produto

```
1  class ProdutoController extends Controller
2  {
3
4      public function index() {
5          return view('produtos.index', ['produtos' => Produto::all()]);
6      }
7
8      public function create() {
9          return view('produtos.create');
10     }
11
12     public function store(Request $request) {
13         $produto = new Produto();
14         $produto->nome = $request->nome;
15         $produto->porcentagem_lucro = $request->lucro;
16         $produto->save();
17
18         return redirect('/produtos/index');
19     }
20
21     public function show($id) {
22         //
23     }
24
25     public function edit($id) {
26         $produto = Produto::find($id);
27         return view('produtos.edit', ['produto' => $produto]);
28     }
29
30     public function update(Request $request, $id) {
31         Produto::find($id)->update($request->all());
32         return redirect('/produtos/index');
33     }
34
35     public function destroy($id) {
36         Produto::find($id)->delete();
37         return redirect('/produtos/index');
38     }
39 }
```

## Controller Venda

```
1 class VendaController extends Controller
2 {
3
4     public function index() {
5         return view('vendas.index', ['vendas' => Venda::all()]);
6     }
7
8     public function create() {
9         return view('vendas.create', [
10             'produtos' => Produto::all(),
11             'clientes' => Cliente::all()
12         ]);
13     }
14
15     public function store(Request $request) {
16
17         // concertar para varios produtos
18         $venda = new Venda();
19         $venda->cliente_id = $request->cliente_id;
20         $venda->valor_total = Produto::find($request->produto_id[0])->valor_venda * $request->quantidade[0];
21         $venda->save();
22
23         for ($i = 0; $i < count($request->quantidade); $i++) {
24             $itensVenda = new ItemVenda();
25             $itensVenda->venda_id = $venda->id;
26             $itensVenda->produto_id = $request->produto_id[$i];
27             $itensVenda->quantidade = $request->quantidade[$i];
28             $itensVenda->valor_unitario = Produto::find($request->produto_id[$i])->valor_venda;
29             $itensVenda->valor_total = $itensVenda->valor_unitario * $itensVenda->quantidade;
30             $itensVenda->save();
31         }
32
33         return redirect()->route('vendas.index');
34     }
35
36     public function show($id) {
37         //
38     }
39
40     public function edit($id) {
41
42     }
43
44     public function update(Request $request, $id) {
45         Venda::find($id)->update($request->all());
46         return redirect()->route('vendas.index');
47     }
48
49     public function destroy($id) {
50         Venda::find($id)->delete();
51         return redirect()->route('vendas.index');
52     }
53 }
```

- Models

### Cliente

```
class Cliente extends Model
{
    use HasFactory;
    protected $table = 'clientes';
    protected $fillable = [
        'nome',
        'debito',
        'endereco',
        'descricao'
    ];

    function venda() {
        return $this->hasMany(Venda::class);
    }
}
```

### Compra

```
class Compra extends Model
{
    use HasFactory;
    protected $table = 'compras';
    protected $fillable = [
        'fornecedor_id',
        'valor_total',
    ];

    public function fornecedor() {
        return $this->belongsTo(Fornecedor::class);
    }

    public function itensCompra() {
        return $this->hasMany(ItemCompra::class);
    }
}
```

### Fornecedor

```
class Fornecedor extends Model
{
    use HasFactory;
    protected $table = 'fornecedores';
    protected $fillable = [
        'nome',
        'telefone',
        'endereco'
    ];

    public function compra() {
        return $this->hasMany(Compra::class);
    }
}
```

### ItemCompra

```
class ItemCompra extends Model
{
    use HasFactory;
    protected $table = 'itens_compra';
    protected $fillable = [
        'compra_id',
        'produto_id',
        'quantidade',
        'valor_unitario',
        'valor_total'
    ];

    public function produto() {
        return $this->belongsTo(Produto::class);
    }

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

### ItemVenda

```
class ItemVenda extends Model
{
    use HasFactory;
    protected $table = 'itens_venda';
    protected $fillable = [
        'venda_id',
        'produto_id',
        'quantidade',
        'valor_unitario',
        'valor_total'
    ];

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

### Produto

```
class Produto extends Model
{
    use HasFactory;
    protected $table = 'produtos';
    protected $fillable = [
        'nome',
        'valor_compra',
        'valor_venda',
        'porcentagem_lucro',
        'quantidade'
    ];
}
```

### Venda

```
class Venda extends Model
{
    use HasFactory;
    protected $table = 'vendas';
    protected $fillable = [
        'cliente_id',
        'valor_total'
    ];

    public function cliente() {
        return $this->belongsTo(Cliente::class);
    }

    public function itensVenda() {
        return $this->hasMany(ItemVenda::class);
    }
}
```