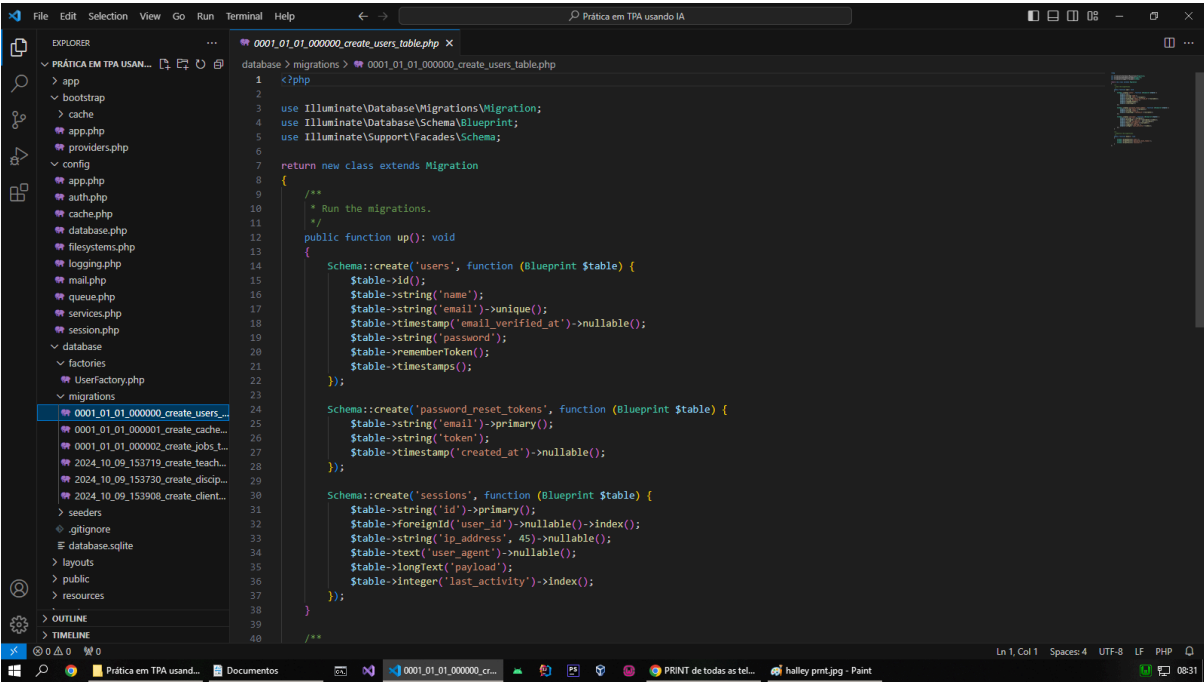


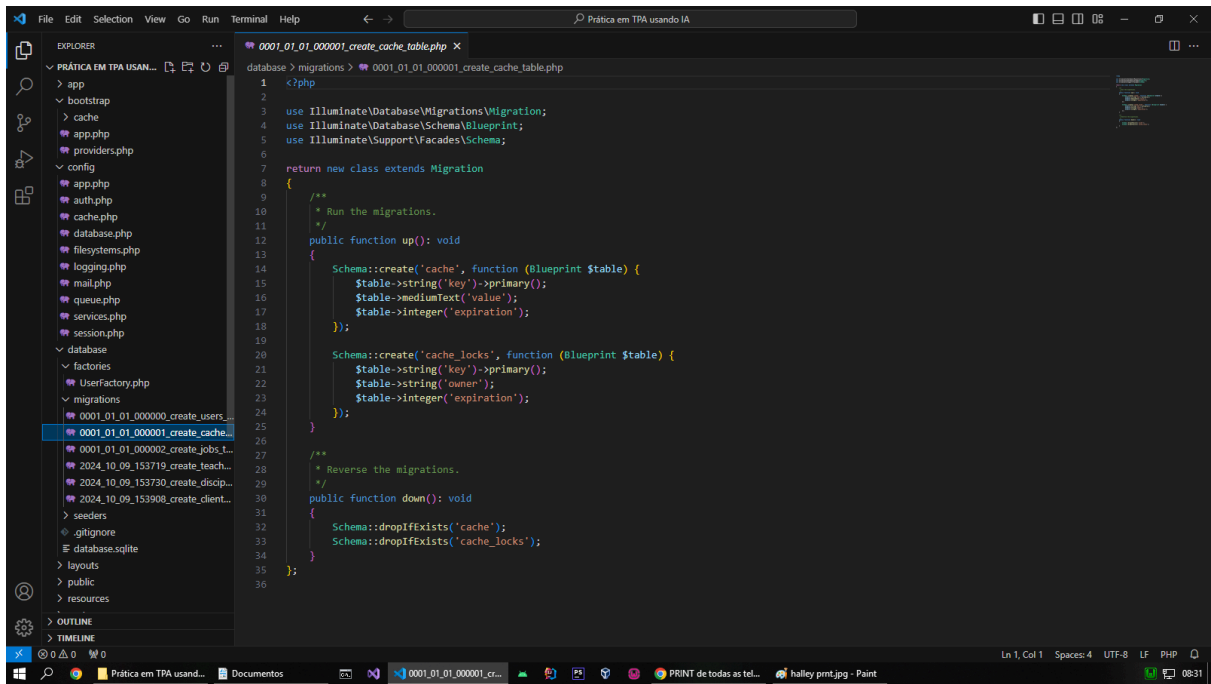
Professores

Adicionar Professor

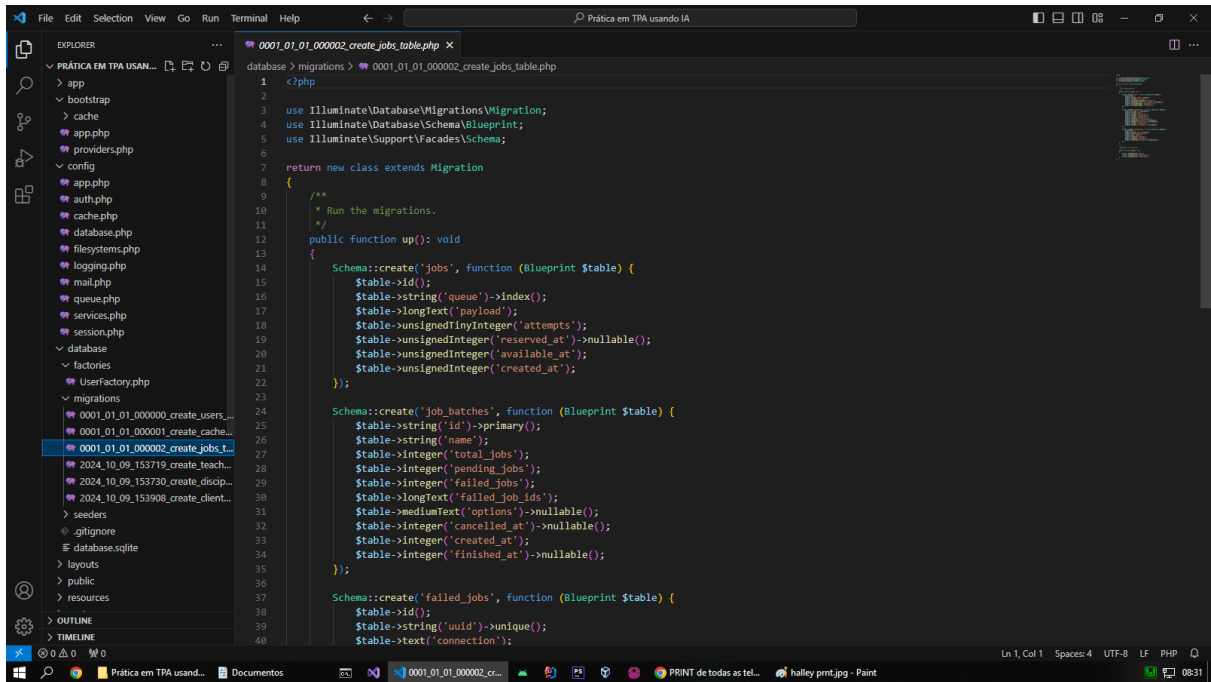
Professor Magro criado com sucesso.

ID	Professor	Ações
1	Gordo	<div>EditarExcluir</div>
3	Magro	<div>EditarExcluir</div>

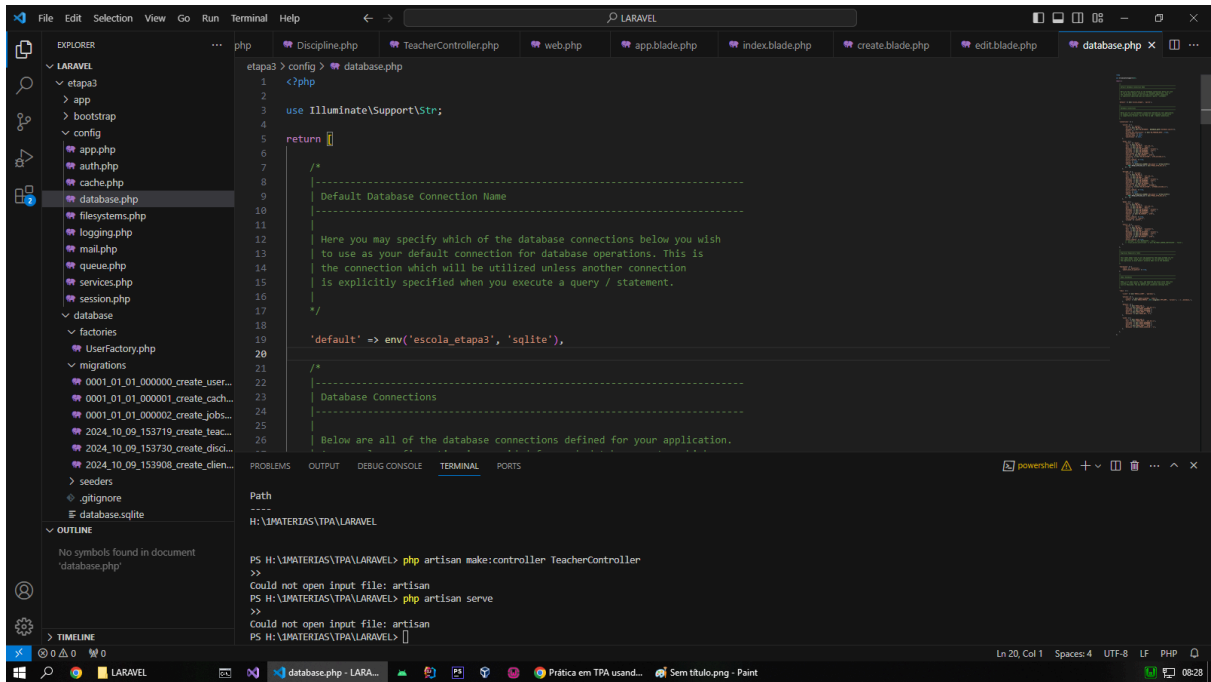
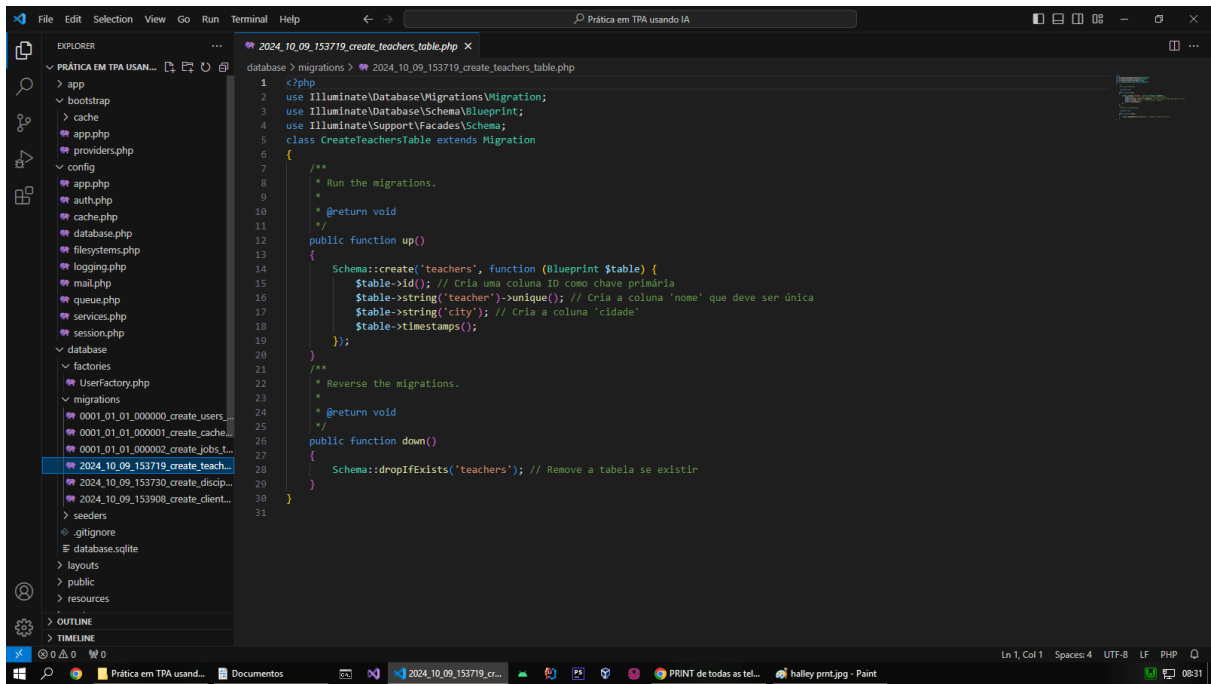


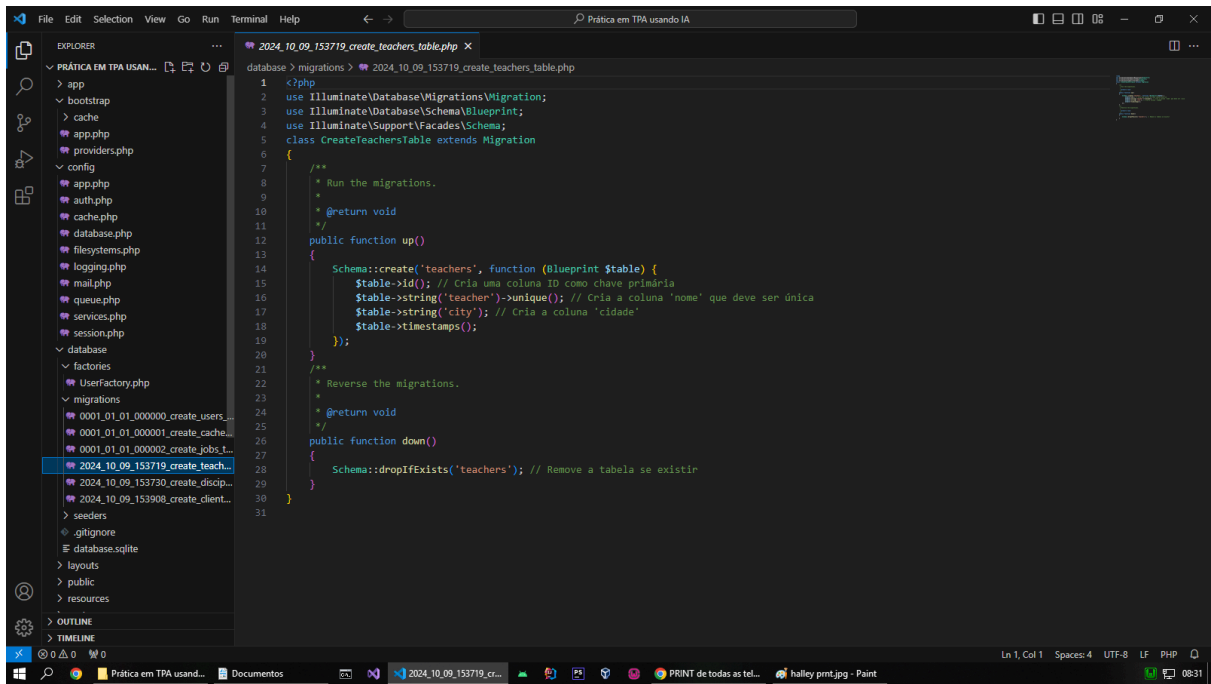


```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::create('cache', function (Blueprint $table) {
15             $table->string('key')->primary();
16             $table->mediumText('value');
17             $table->integer('expiration');
18         });
19
20         Schema::create('cache_locks', function (Blueprint $table) {
21             $table->string('key')->primary();
22             $table->string('owner');
23             $table->integer('expiration');
24         });
25     }
26
27     /**
28      * Reverse the migrations.
29      */
30     public function down(): void
31     {
32         Schema::dropIfExists('cache');
33         Schema::dropIfExists('cache_locks');
34     }
35 };
36
```

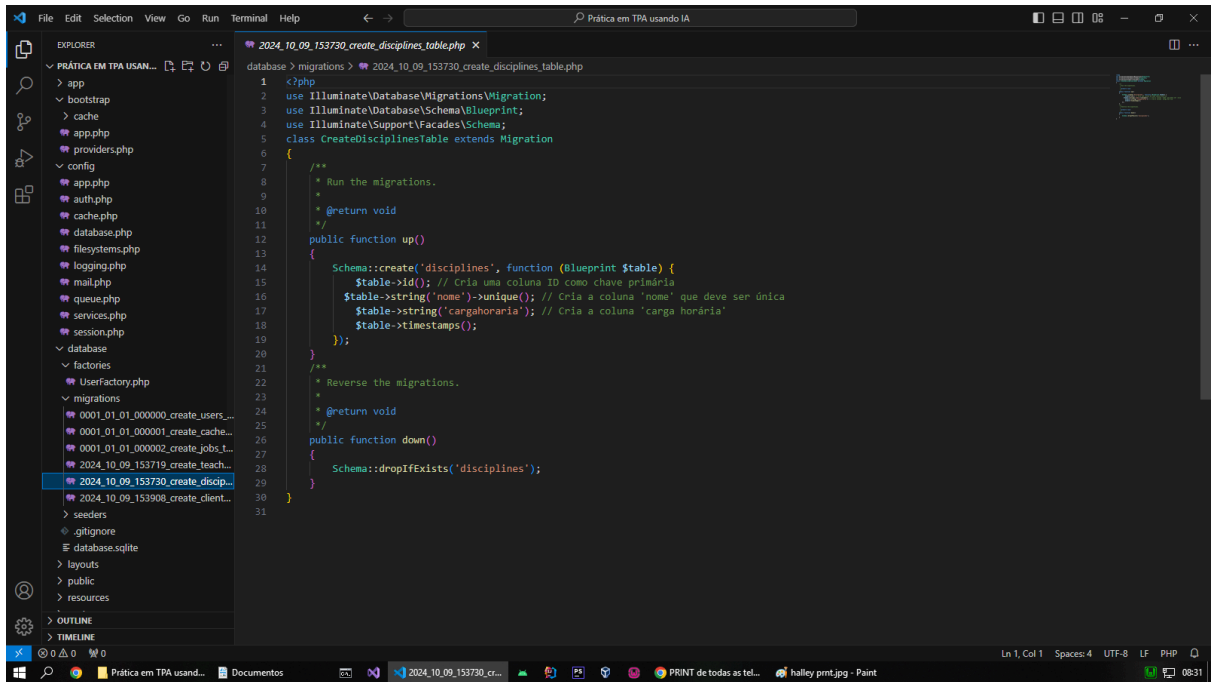


```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::create('jobs', function (Blueprint $table) {
15             $table->id();
16             $table->string('queue')->index();
17             $table->longText('payload');
18             $table->unsignedTinyInteger('attempts');
19             $table->unsignedInteger('reserved_at')->nullable();
20             $table->unsignedInteger('available_at');
21             $table->unsignedInteger('created_at');
22         });
23
24         Schema::create('job_batches', function (Blueprint $table) {
25             $table->string('id')->primary();
26             $table->string('name');
27             $table->integer('total_jobs');
28             $table->integer('pending_jobs');
29             $table->integer('failed_jobs');
30             $table->longText('failed_job_ids');
31             $table->mediumText('options')->nullable();
32             $table->integer('cancelled_at')->nullable();
33             $table->integer('created_at');
34             $table->integer('finished_at')->nullable();
35         });
36
37         Schema::create('failed_jobs', function (Blueprint $table) {
38             $table->id();
39             $table->string('uuid')->unique();
40             $table->text('connection');
41         });
42     }
43
44     /**
45      * Reverse the migrations.
46      */
47     public function down(): void
48     {
49         Schema::dropIfExists('jobs');
50         Schema::dropIfExists('job_batches');
51         Schema::dropIfExists('failed_jobs');
52     }
53 };
54
```

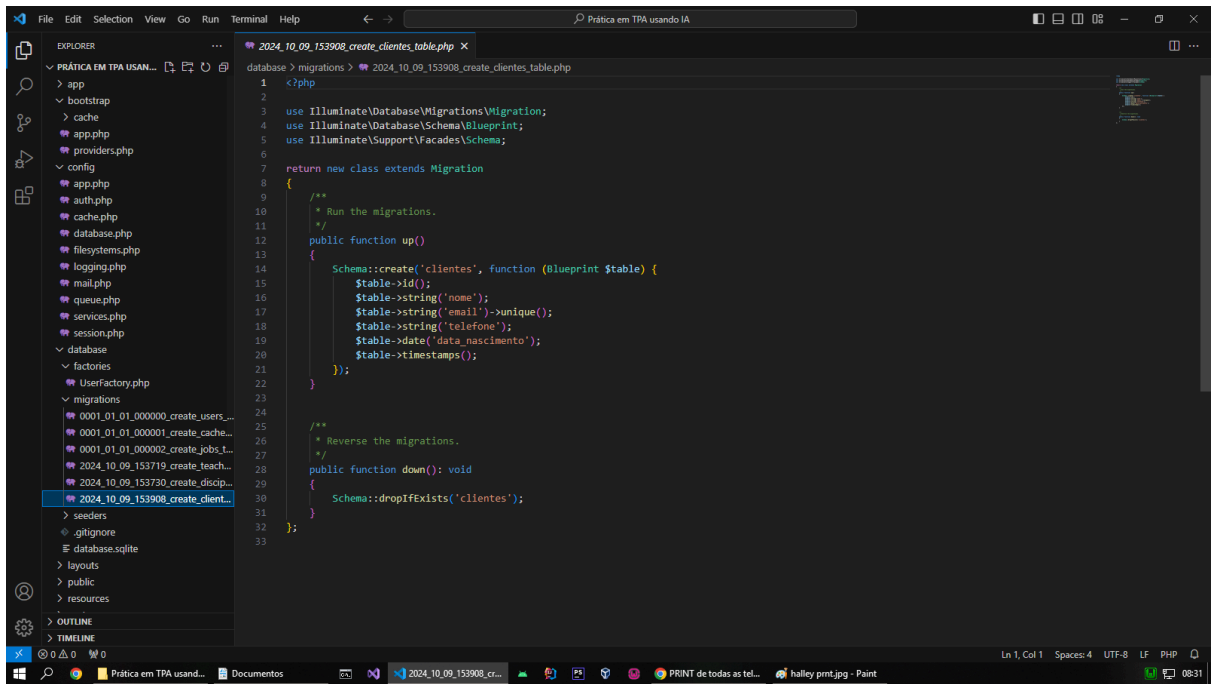




```
1 <?php
2 use Illuminate\Database\Migrations\Migration;
3 use Illuminate\Database\Schema\Blueprint;
4 use Illuminate\Support\Facades\Schema;
5 class CreateTeachersTable extends Migration
6 {
7     /**
8      * Run the migrations.
9      *
10     * @return void
11     */
12     public function up()
13     {
14         Schema::create('teachers', function (Blueprint $table) {
15             $table->id(); // Cria uma coluna ID como chave primária
16             $table->string('teacher')->unique(); // Cria a coluna 'nome' que deve ser única
17             $table->string('city'); // Cria a coluna 'cidade'
18             $table->timestamps();
19         });
20     }
21     /**
22      * Reverse the migrations.
23      *
24      * @return void
25      */
26     public function down()
27     {
28         Schema::dropIfExists('teachers'); // Remove a tabela se existir
29     }
30 }
31
```

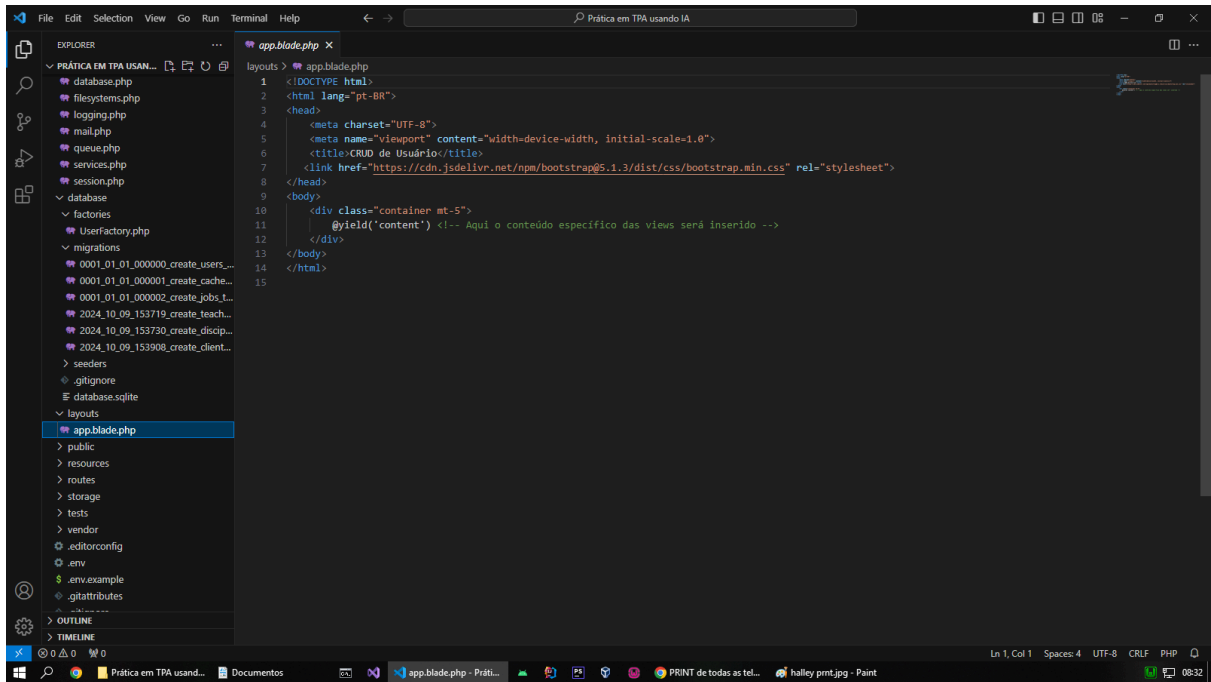


```
1 <?php
2 use Illuminate\Database\Migrations\Migration;
3 use Illuminate\Database\Schema\Blueprint;
4 use Illuminate\Support\Facades\Schema;
5 class CreateDisciplinesTable extends Migration
6 {
7     /**
8      * Run the migrations.
9      *
10     * @return void
11     */
12     public function up()
13     {
14         Schema::create('disciplines', function (Blueprint $table) {
15             $table->id(); // Cria uma coluna ID como chave primária
16             $table->string('nome')->unique(); // Cria a coluna 'nome' que deve ser única
17             $table->string('cargahoraria'); // Cria a coluna 'carga horária'
18             $table->timestamps();
19         });
20     }
21     /**
22      * Reverse the migrations.
23      *
24      * @return void
25      */
26     public function down()
27     {
28         Schema::dropIfExists('disciplines');
29     }
30 }
31
```



This screenshot shows the Visual Studio Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with folders like 'app', 'bootstrap', 'cache', 'database', 'factories', 'migrations', 'seeds', 'storage', 'tests', and 'vendor'. The code editor displays a PHP file named '2024_10_09_153908_create_clientes_table.php' with the following content:

```
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up()
13     {
14         Schema::create('clientes', function (Blueprint $table) {
15             $table->id();
16             $table->string('nome');
17             $table->string('email')->unique();
18             $table->string('telefone');
19             $table->date('data_nascimento');
20             $table->timestamps();
21         });
22     }
23
24     /**
25      * Reverse the migrations.
26      */
27     public function down(): void
28     {
29         Schema::dropIfExists('clientes');
30     }
31
32 };
```



This screenshot shows the Visual Studio Code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with folders like 'app', 'bootstrap', 'cache', 'database', 'factories', 'migrations', 'seeds', 'storage', 'tests', and 'vendor'. The code editor displays a Blade template file named 'app.blade.php' with the following content:

```
1 <DOCTYPE html>
2 <html lang="pt-BR">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>CRUD de Usuário</title>
7     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
8 </head>
9 <body>
10     <div class="container mt-5">
11         @yield('content') <!-- Aqui o conteúdo específico das views será inserido -->
12     </div>
13 </body>
14 </html>
```