

التاريخ: 2024-06-19.



جامعة طرابلس / كلية تقنية المعلومات قسم هندسة البرمجيات.



السيد المحترم / د. عبدالحميد الواعر.

الموضوع:- تقرير عن WAFs البنية على نمط MVC.

بعد التحية والسلام.

في إطار مادة إعادة استخدام البرمجيات (ITSE422)، يطلب إنشاء تطبيق ويب باستخدام إحدى WAFs البنية على MVC pattern، وكتاب تقرير يشرح الـ MVC pattern وكيف سيتم استخدامه عند إنشاء التطبيق. عليه تقدم الى حضرتكم هذا التقرير.

مع خالص التحية،

الطالب: أحمد مصطفى المحجوب.

رقم القيد: 2191804140.

الطالب: علي محمد الغناي.

رقم القيد: 2191804737.

الطالب: انس فتحي المغربي.

رقم القيد: 2191804670.

مقدمة

يوضح هذا التقرير تطوير تطبيق ويب بسيط لإدارة وعرض معلومات الطلاب. يتم بناء التطبيق باستخدام نمط MVC (Model-View-Controller) ومجموعة من إطارات تطبيقات الويب (WAFs). الهدف الرئيسي هو فهم وتنفيذ نمط MVC لتحسين إعادة استخدام البرمجيات وتنظيمها.

الأهداف

1. تطوير تطبيق ويب لعرض وإدارة معلومات الطلاب (الاسم، العنوان، العمر).
2. تنفيذ نمط MVC لتنظيم التطبيق.
3. تقديم تقرير مفصل يشرح نمط MVC واستخدامه في تطبيق الويب.

التقنيات المستخدمة

- **Laravel**: إطار عمل ويب لبناء التطبيق.
- **Vite**: أداة بناء متكاملة شبيهة بويب باك.
- **SQLite**: قاعدة بيانات لتخزين معلومات الطلاب.
- **Axios**: عميل HTTP لإجراء طلبات API.

هيكل المشروع:

```
root
├── app
│   ├── Http/Controllers
│   │   ├── Controller.php
│   │   └── delete-all-student.js
│   ├── Modles
│   │   └── Student.php
│   └── ...
├── resources
│   ├── js
│   │   ├── app.js
│   │   ├── bootstrap.js
│   │   └── deleteAll.js
│   ├── views
│   │   ├── layouts
│   │   │   └── app.blade.php
│   │   └── students
│   │       ├── create.blade.php
│   │       └── index.blade.php
│   └── ...
├── routes
│   ├── console.php
│   └── web.php
└── ...
```

التاريخ: 2024-06-19.

شرح نمط MVC

يقسم نمط MVC التطبيق إلى ثلاث مكونات مترابطة:

1. **النموذج (Model):** يدير البيانات والمنطق التجاري.
2. **العرض (View):** يتولى عرض المعلومات.
3. **التحكم (Controller):** يدير منطق التطبيق وإدخال المستخدم.

يساعد هذا التقسيم في تنظيم قاعدة الكود وتحسين قابلية التوسع وتعزيز إعادة استخدام الكود.

شرح طريقة تنفيذ البرنامج (Implementation):

النموذج (Model):

يعرّف نموذج **Student.php** مخطط الطالب ويتفاعل مع قاعدة البيانات **SQLite**.

```
app > Models > Student.php
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Student extends Model
9  {
10     use HasFactory;
11     protected $fillable = [
12         'name',
13         'address',
14         'age',
15     ];
16 }
17
```

التاريخ: 2024-06-19.

العرض (View)

تم تعريف العروض باستخدام قوالب Pug. تقوم بتقديم معلومات الطالب وتوفر واجهات لإضافة وحذف الطلاب.

app.blade.php:

```
resources > views > layouts > app.blade.php
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <meta charset="utf-8">
5      <title>Reuse MVC Project</title>
6      <meta name="viewport" content="width=device-width, initial-scale=1">
7      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZy76Pj0z9MPlUscQ+Kw9697820u6T9y89CaIOmQ9nnw39du6cv8q/Tw" crossorigin="anonymous">
8  </head>
9  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
10      <div class="container-fluid">
11          <a class="navbar-brand" href="#">Reuse MVC Project</a>
12      </div>
13  </nav>
14  <body>
15      <div class="container">
16          @yield('content')
17      </div>
18  </body>
19  </html>
20
```

التاريخ: 2024-06-19.

:index.blade.php

```
resources > views > students > index.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5  <h4>Dashboard</h4>
6  <div class="mb-3">
7  <a class="btn btn-primary" href="/add-student">Add New Student</a>
8  <button class="btn btn-danger ml-3" onclick="deleteAllStudents()">Delete All Students</button>
9  </div>
10 @if(!is_null($students) && count($students) > 0)
11     @foreach($students as $student)
12         <div class="card mb-3">
13             <div class="card-body">
14                 <h5 class="card-title">{{ $student->name }}</h5>
15                 <p class="card-text">
16                     Age: {{ $student->address }}
17                 </p>
18                 <p class="card-text">
19                     Address: {{ $student->age }}
20                 </p>
21             </div>
22         </div>
23     @endforeach
24 @else
25     <p>No students found.</p>
26 @endif
27 @vite('resources/js/app.js')
28
29 </div>
30 @endsection
```

create.blade.php:

```
resources > views > students > create.blade.php
1  @extends('layouts.app')
2
3  @section('content')
4  <div class="container">
5      <h2>Add Student</h2>
6      <form method="post" action="{{ url('/api/add-student') }}">
7          @csrf
8          <div class="form-group">
9              <label for="name">Name:</label>
10             <input type="text" class="form-control" id="name" name="name">
11         </div>
12         <div class="form-group">
13             <label for="address">Address:</label>
14             <input type="text" class="form-control" id="address" name="address">
15         </div>
16         <div class="form-group">
17             <label for="age">Age:</label>
18             <input type="number" class="form-control" id="age" name="age">
19         </div>
20         <button type="submit" class="btn btn-primary">Submit</button>
21     </form>
22 </div>
23 @endsection
24
```

التحكم (Controller)

المتحكمات تدير منطق التطبيق وإدخال المستخدم، وتدير تدفق البيانات بين النماذج والعروض.

StudentController.php:

التاريخ: 2024-06-19.

```
app > Http > Controllers > StudentController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Models\Student;
7
8  class StudentController extends Controller
9  {
10     public function index()
11     {
12         $students = Student::all();
13         return view('students.index', compact('students'));
14     }
15
16     public function create()
17     {
18         return view('students.create');
19     }
20
21     public function store(Request $request)
22     {
23         $request->validate([
24             'name' => 'required',
25             'address' => 'required',
26             'age' => 'required|integer',
27         ]);
28
29         Student::create($request->only(['name', 'address', 'age']));
30
31         return redirect('/')->with('success', 'Student added successfully.');
```

المسارات (Routes):

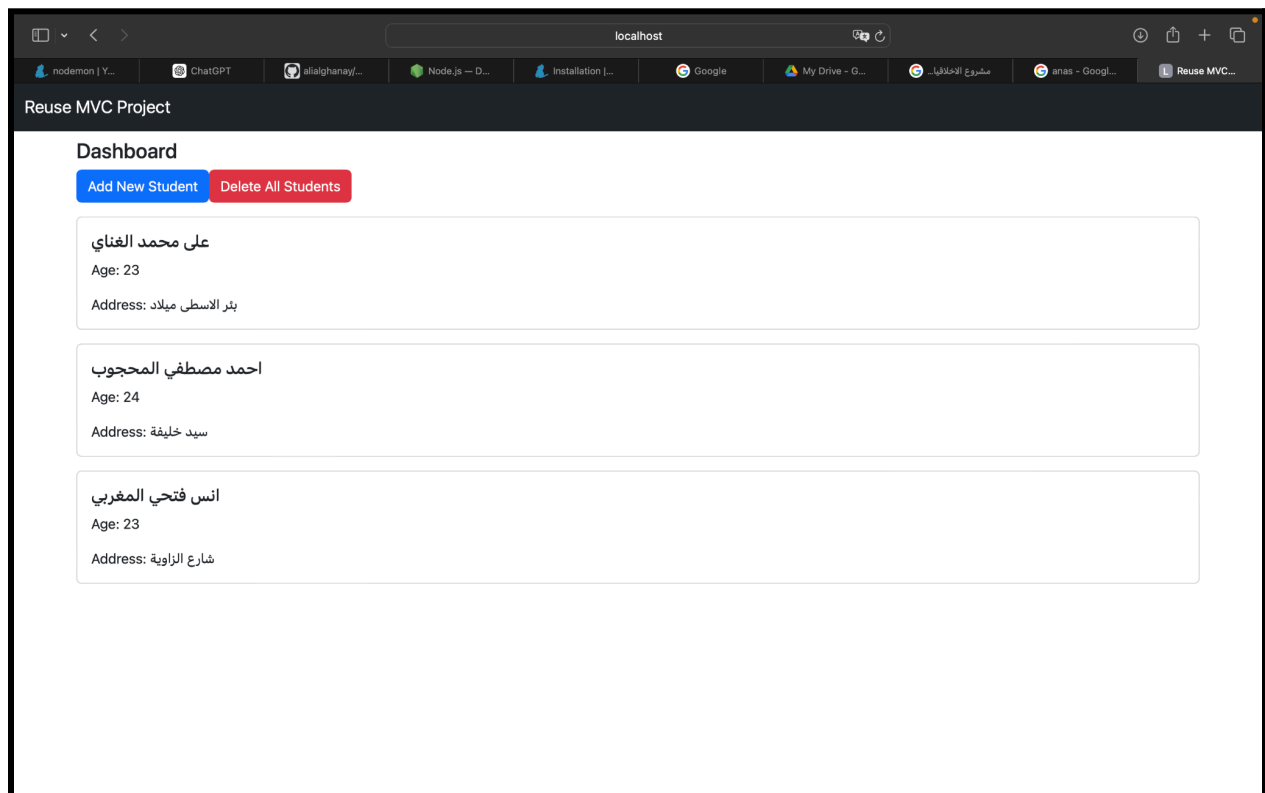
المسارات تعرف نقاط النهاية وتعينها لدوال المتحكمات المناسبة.

web.php:

التاريخ: 2024-06-19.

```
routes > web.php
1  <?php
2
3  use App\Http\Controllers\StudentController;
4
5  Route::get('/', [StudentController::class, 'index']);
6  Route::get('/add-student', [StudentController::class, 'create']);
7  Route::post('/api/add-student', [StudentController::class, 'store']);
8  Route::post('/api/deleteAll', [StudentController::class, 'deleteAll']);
9
```

النتيجة النهائية:



الخاتمة

يبرز هذا المشروع استخدام نمط MVC في بناء تطبيق ويب بسيط لإدارة معلومات الطلاب. من خلال فصل الاهتمامات إلى نماذج وعروض ومتحكمات، يصبح قاعدة الكود أكثر تنظيمًا ونقلًا، وأكثر صيانة. كما يعزز هذا النهج إعادة استخدام البرمجيات، مما يسهل توسيع وصيانة التطبيق في المستقبل.