

## Taller 01 - Calificación de taller

Nombre calificador: Valentina Morales Villada

Nombre persona a evaluar: Samuel Rendón Trujillo

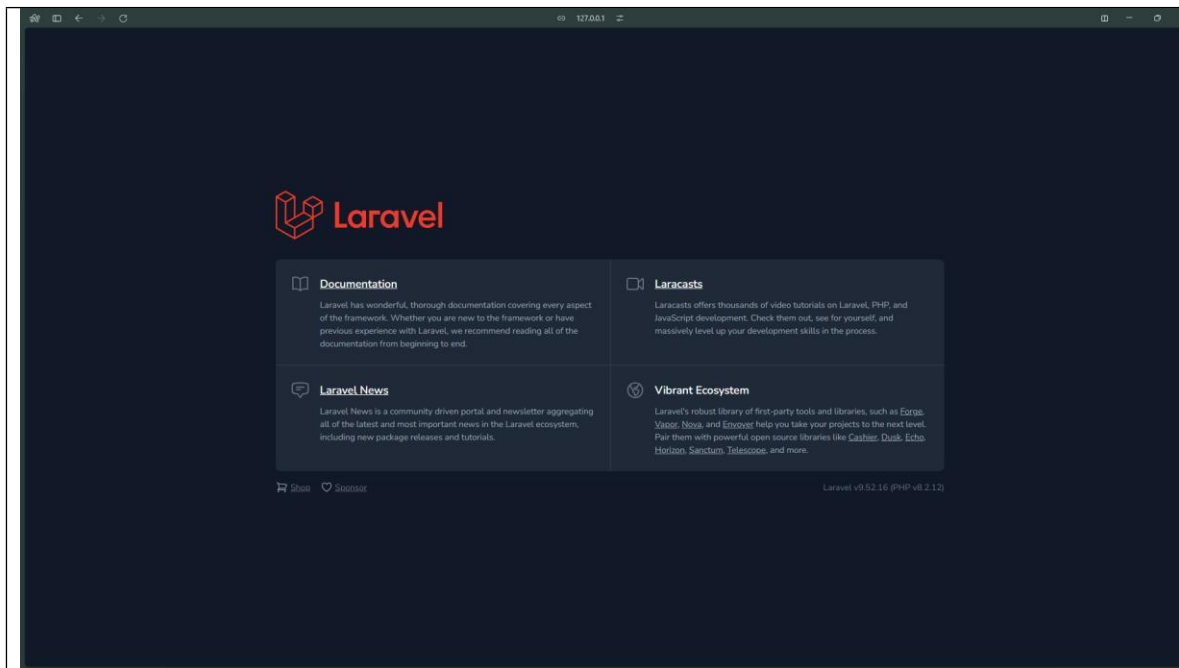
### Actividades:

1. Descargue y ponga a correr el proyecto del compañero que le fue asignado.

Si tiene problemas con la ejecución del proyecto lístelas acá.

No hubo problemas.

Pegue un pantallazo de la aplicación corriendo en la ruta inicial acá:



2. Revise el **routes/web.php**, si tiene comentarios, sugerencias déjelas a continuación, en caso contrario, si le parece que el código está bien desarrollado, deje este bloque en blanco.
  - **Nota:** pregúntese cosas tales como: ¿El programador utilizó un mismo estilo para todas las rutas?, ¿Hay código que le parece extraño que piensa que no debería ir ahí?, ¿etc?

- Se recomienda eliminar comentarios innecesarios
- Se recomienda eliminar código que no se va a usar. Como el default de "welcome" que trae laravel
- Se recomienda usar la convención que recomienda el profe para definir las rutas, ya que de esa manera no sería necesario importar los controladores

3. Revise el **controlador** creado y cada uno de sus **métodos**, si tiene comentarios, sugerencias déjelas a continuación, en caso contrario, si le parece que el código está bien desarrollado, deje este bloque en blanco.
- **Nota:** pregúntese cosas tales como: ¿El programador utilizó un mismo estilo para todos los métodos?, ¿Qué hay de la manera de nombrar las variables y los métodos?, ¿Hay código que le parece extraño que piensa que no debería ir ahí?, ¿etc?

```
public function store(Request $request)
{
    $request->validate([
        'username' => 'required|string|max:255',
        'email' => 'required|string|email|max:255|unique:custom_users',
        'password' => 'required|string|min:8',
    ]);

    $user = CustomUser::create([
        'username' => $request->username,
        'email' => $request->email,
        'password' => bcrypt($request->password),
    ]);

    return redirect()->route('custom-users.index')->with('success', 'Elemento creado satisfactoriamente.');
```

En este método se declara la variable \$user pero no se está usando en ningún lado, por lo que me parece un poco innecesario, ya que solo con el modelo es suficiente. (CustomUser::create)

- Se recomienda pasar la lógica de validate al modelo
- Se recomienda usar tipado para el retorno y los parámetros de los métodos

No usar el compact, usar el viewData que vimos en clase

4. Revise el **modelo** creado y cada uno de sus **métodos**, si tiene comentarios, sugerencias déjelas a continuación, en caso contrario, si le parece que el código está bien desarrollado, deje este bloque en blanco.
- **Nota:** pregúntese cosas tales como: ¿El programador utilizó un mismo estilo para todos los métodos?, ¿Qué hay de la manera de nombrar los atributos, variables y los métodos?, ¿Se definieron métodos get y set?, ¿Hay código que le parece extraño que piensa que no debería ir ahí?, ¿etc?

No se definieron los setters y getters, y se deja el use HasFactory, lo cual es innecesario ya que no se crearon objetos de CustomUser con ayuda del factory.

```
2024_08_28_011013_create_custom_users_table.php CustomUser.php X
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class CustomUser extends Model
9  {
10     use HasFactory;
11
12     protected $fillable = ['username', 'password', 'email'];
13 }
14
```

5. Revise las **vistas**, si tiene comentarios, sugerencias déjelas a continuación, en caso contrario, si le parece que el código está bien desarrollado, deje este bloque en blanco.
- **Nota:** pregúntese cosas tales como: ¿El programador utilizó un mismo estilo para todas las vistas?, ¿Se utilizan bloques php dentro de las vistas, o solo blade?, ¿Qué hay de la manera de nombrar las vistas?, ¿Se utilizó Lang, o se copiaron directamente los textos en las vistas?, ¿Hay algún comentario sobre la apariencia de las vistas?, ¿Hay código que le parece extraño que piensa que no debería ir ahí?, ¿etc?

Olvidó cambiar el welcome default de laravel por el que contenía los dos botones de las acciones que se pueden realizar, crear usuarios y listar usuarios. Por lo que hay 2 archivos welcome.blade.php y uno no se está utilizando.

```
2024_08_28_011013_create_custom_users_table.php welcome.blade.php X
1  <a href="{{ route('custom-users.create') }}">Crear Custom User</a>
2  <a href="{{ route('custom-users.index') }}">Listar Custom Users</a>
```

También la forma de llamar las variables que debe hacerse con los getters, y como no se implementaron no se pudieron llamar de esta forma.

```

v class="row justify-content-center">
  <div class="col-md-8">
    <div class="list-group">
      @foreach ($users as $user)
        <a href="{{ route('custom-users.show', $user->id) }}" class="list-group-item list-group-item-action">
          <div class="d-flex justify-content-between align-items-center">
            <h5 class="mb-1">User ID: {{ $user->id }}</h5>
            <small class="text-muted">{{ $user->username }}</small>
          </div>
        </a>
      @endforeach
    </div>
  </div>
</div>
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

```

- Se recomienda no abrir etiqueta php en las vistas
- Se recomienda el uso de Lang

6. ¿Todas las 7 actividades del taller 1 funcionaron correctamente, o encontró problemas en algunas de ellas?

Todas funcionan correctamente, aunque no saca mensaje de creado exitosamente y mensaje de eliminado. Tampoco sale la vista inicial con los dos botones, pero la vista si se implementó, solo que no se llamó correctamente.

7. ¿Tiene algún comentario adicional?

En general está muy bien, y cumple con las actividades propuestas. Solo faltaron algunos detallitos.

## Entrega Taller 01 - Calificador

- Suba este documento a Teams.
- Se deberá subir este documento a más tardar el 31 de agosto a las 10:00 am.
- El docente y/o monitor revisará cada taller y cada calificación de cada “compañero calificador” el fin de semana.

- 0.0 validaciones en controlador (Ojo que en otra entrega reduce nota)
- 0.0 acceder sin get y set a los datos del modelo (Ojo que en otra entrega reduce nota)
- 0.1 no hay estándar para definir las rutas, se usan dos tipos
- 0.1 de falta tipados de argumentos en el controlador
- 0.1 de falta tipados de retornos en el controlador
- 0.4 de no definir comentario de atributos, y no definir gets y sets, mas el has factory que no se usa
- 0.3 de usar php en vistas
- 0.2 de no usar un layout