

Taller 01 - Calificación de taller

Nombre calificador: Monitor

Nombre persona a evaluar: Jonathan Betancur Espinosa

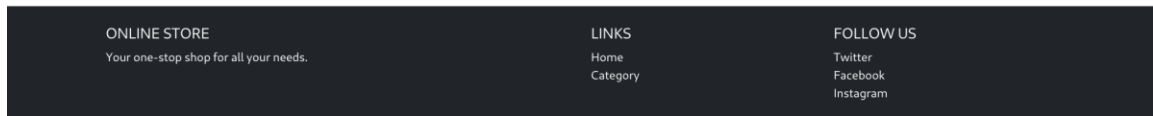
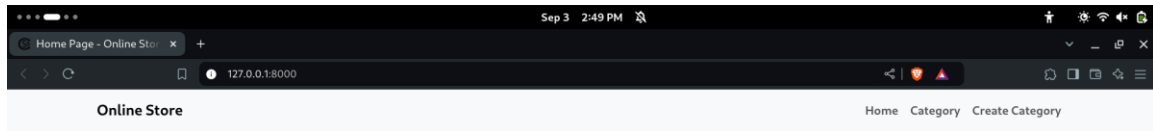
Actividades:

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

Si tiene problemas con la ejecución del proyecto lístelas acá.

Ninguno

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 utilizo un mismo estilo para todas las rutas?, ¿Hay código que le parece extraño que piensa que no debería ir ahí?, ¿etc?

Todo bien

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 utilizo 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?

- Se recomienda poner tipo de retorno en todos los métodos (falta en el método success)
- En el método show se recomienda usar getters para acceder a los atributos
- Se recomienda eliminar comentarios innecesarios. Ejm “// to be sent to the view”
- Se recomienda que la lógica de la validación se mueva al modelo
- ojo con el espaciado

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 utilizo 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?

- Se recomienda agregar getter para createdAt y updatedAt. Y ponerlos en el comentario inicial donde se listan los atributos del modelo
- Se recomienda no tener setter para el id
- Se recomienda tipado para los parametros de los métodos

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 utilizo 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?

- Se recomienda el uso de Lang
- Se recomienda el uso de getters para acceder a los atributos
- Evitar el uso de "Style" en las vistas. Mejor crear un archivo css en la carpeta public (show linea 28 se usa Style)

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

Todo funciona correctamente

7. ¿Tiene algún comentario adicional?

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 ojo con el espaciado en el controlador, recuerda correr laravel pint
- 0.1 de falta tipados de retornos en el controlador
- 0.1 de listado de comentario de atributos en el modelo y set y get incompletos (verifica con la bd)
- 0.1 de falta tipados de argumentos en el modelo