**Mapa de archivos proyecto Laravel**

Estructura general de la página:

-Routes/web.php

-resources/views/layout.blade.php //Esta creo que es como una plantilla que usan las 4 siguientes

-resources/views/inicio.blade.php

-resources/views/menu.blade.php

-resources/views/reservas.blade.php

-resources/views/contacto.blade.php

**Para la sección de contacto:**

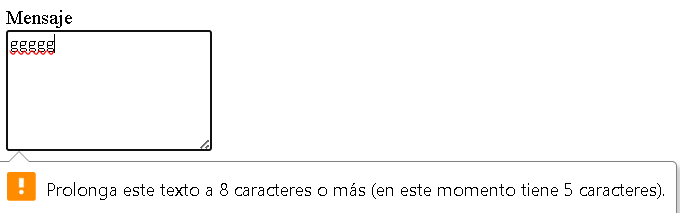
Funciona perfectamente! Ahora falta estudiarme el código

Y aplicarle posteriormente formatos para que se vea más bonito

Cambiar este texto:



Y también este texto:



Mejora importante: Configurar el envío de emails con cola (en vez de inmediato). Interviene mínimo el archivo 2.

**0).env**

Para editar la configuración del email

**1) routes/web.php (Rutas)**

Define por dónde entra la petición (GET para ver el formulario, POST para enviarlo).

Mantendremos tu ruta GET y añadiremos una POST.

Opcional: en vez de un closure, usar un Controlador nos da orden y escalabilidad.

**2)app/Http/Controllers/ContactController.php (Controlador)**

Contiene la lógica cuando llega el POST: recibe datos, delega la validación (Form Request), y decide qué responder (redirigir con éxito o con errores).

**3)app/Http/Requests/StoreContactRequest.php (Form Request)**

Validación en servidor de los datos del formulario. Así, aunque alguien intente saltarse la validación del navegador, el backend sigue protegiendo.

**4)App/Mail/ContactMessage.php**

Define cómo se construye el email: asunto, vista, datos, etc

**5)Ressources/views/emails/contacto.blade.php**

Es la vista del email. Es decir, la plantilla con el mensaje que recibiré.

**6)resources/views/contacto.blade.php (Vista)**

Es la vista del formulario en la pagina web.

Añadimos validación del lado del cliente (HTML5: required, minlength, maxlength, type="email", pattern para el teléfono) y un poquito de JS (contador de caracteres, bloqueo de doble envío).

**Para la sección de reservas:**

\*quizás esta sección la pueda mejorar encapsulando la lógica del routes/web.php en un controlador, para dejar el código más limpio



Archivos de configuración para el envío automático del email:

**app\Mail\ReservaConfirmada.php**: la clase del email

**resources\views\emails/reserva\_confirmada.blade.php**: la pantilla markdown del email

**-Routes/web.php:** Este interviene siempre en todos

**¿Qué me falta por hacer?**