



Ecode

Software Development

Programación Web Full Stack

Trabajo integrador - S7 - M08C20

Trabajo Integrador - Sprint 7

> Introducción

Que levante la mano 🙋🙋 quien nunca haya cargado un formulario de mala gana 📋✍️😡 y con datos que ni siquiera tenían sentido. Tal vez para acceder al increíble servicio que ofrecía ese sitio que nos contaron, o por participar de esa promoción increíble que solo nos pedía llenar un formulario de 352 campos. ¿Todas y todos culpables? Ya me parecía... 😏

Si el formulario se llena con datos inválidos no hay problema, el problema es si llegan a la base de datos, ahí es donde se desata el caos y se rompe todo. **¡Para eso están las validaciones!**

En este sprint vamos a estar validando tanto desde el front-end como desde el back-end.



> Requisitos

1. **Base de datos bien configurada:** la protección final y más importante es la de la base de datos. Si todo falla, la base de datos evitará que se carguen datos que puedan dañarla.

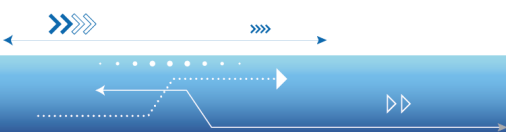
> Objetivo

Vamos a estar validando los formularios del sitio trabajando de atrás para adelante.

En la **primera parte** vamos a estar validando los datos desde el **back-end**. Esta es una protección muy importante ya que como sabemos el usuario no puede acceder al mismo.

En la **segunda parte** vamos a estar validando en el **front-end**. Con esto evitaremos estar enviando información inválida al servidor y, además, mejoraremos la experiencia del visitante.

Importante tener en cuenta que la validación del front-end puede ser deshabilitada desde el navegador. Por eso es indispensable que también validemos en el back-end.



> Metodologías ágiles

¡Vamos con la séptima vuelta de retro y planificación! 📝🧐💡✨

Como siempre les recomendamos que no se salteen este paso, es muy importante. 😊👍

La retrospectiva

A esta altura es posible que el equipo haya alcanzado una buena velocidad de trabajo. Si es así, pueden enfocarse en mantenerla. De lo contrario, deberán enfocarse en los puntos que se puedan mejorar.

Implementen nuevamente la dinámica de la **estrella de mar**, resaltando aquello que hay que:

1. Comenzar a hacer.
2. Hacer más.
3. Continuar haciendo.
4. Hacer menos.
5. Dejar de hacer.

Pueden leer más sobre [esta ceremonia aquí](#).

El tablero de trabajo

Momento de **reiniciar el tablero** para acomodar este sprint de bases de datos. Como siempre, lo que haya quedado del anterior sprint se suma al actual.

Recuerden que durante la planificación es importante:

- Debatir cada tarea en conjunto para asegurarse de que no haya dudas sobre su alcance (hasta dónde van a hacer) y sobre cómo van a resolverla.
- Estimar la dificultad de la tarea y si ésta requiere de que alguna otra tarea esté terminada antes de poder iniciarla (para determinar el orden).
- Asignar tentativamente los responsables de cada una de ellas.



(Opcional) La reunión daily o weekly

La **daily standup** es una reunión, que en los equipos de **Scrum** se realiza todos los días, donde cada integrante habla como máximo 3 minutos de 3 temas puntuales:

- Qué hizo ayer.
- Si se encontró con algún impedimento.
- Qué va a hacer hoy.

El formato está pensado para ser rápido y liviano, solo queremos la información más importante de las tareas y los impedimentos.

Importante: no es necesario que esto lo hagan todos los días, al menos una vez por semana sería ideal.

> Consignas

Planificación y trabajo en equipo

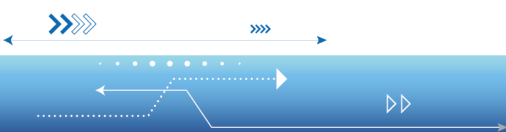
1. Realizar un breve retrospectiva

Piensen qué hicieron bien en el sprint anterior, qué hicieron mal, qué deberían empezar a hacer, qué deberían dejar de hacer, sigan [esta dinámica](#).

Entregable: Actualizar el archivo retro.md con las principales conclusiones de la retro del sexto sprint.

2. Actualizar el tablero de trabajo

Discutan las tareas que se desprenden de este documento, determinen en qué orden deberán realizarlas, asignen integrantes a cada tarea.



Entregable: link al documento o plataforma que utilicen para organizar el trabajo.

(Opcional) Implementar daily / weekly standups

Cada equipo habla como máximo 3 minutos de 3 temas puntuales:

- Qué hizo ayer.
- Si se encontró con algún impedimento.
- Qué va a hacer hoy.

Entregable: archivo daily.md con sus principales impresiones (positivas, neutras o negativas) sobre la utilidad de esta ceremonia.

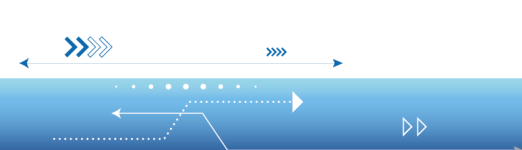
Validaciones de back-end y front-end

3. Validaciones del back-end

Si queremos hacer que nuestro sitio sea una fortaleza a prueba de corrupción de datos, necesitamos validar todos aquellos puntos donde nos llegue información desde el lado del usuario.

Como ya se habrán imaginado, estamos hablando de los formularios. 📄🔍🔥 Los siguientes puntos son una guía, su sitio puede requerir otras validaciones así que ajústense según lo necesiten.

Vamos con esas consignas:



- **Registro de usuarios**

- **Nombre y apellido**
 - Obligatorio.
 - Deberá tener al menos 2 caracteres.
- **Email**
 - Obligatorio.
 - Deberá ser un formato de e-mail válido.
 - No puede repetirse con los e-mails ya registrados.
- **Contraseña**
 - Obligatoria.
 - Deberá tener al menos 8 caracteres.
 - (Opcional) → Deberá tener letras mayúsculas, minúsculas, un número y un carácter especial.
- **Imagen**
 - Deberá ser un archivo válido (JPG, JPEG, PNG, GIF).

- **Login de usuarios** (este ya lo deberían tener de sprints anteriores 😊👍)

- **Email**
 - Obligatorio.
 - Deberá ser válido.
 - Deberá existir en base.
- **Contraseña**
 - Obligatoria.
 - Deberá coincidir con la existente en base.

- **Creación y modificación de productos**

- **Nombre**

- **Obligatorio.**

- **Deberá tener al menos 5 caracteres.**

- **Descripción**

- **Deberá tener al menos 20 caracteres.**

- **Imagen**

- **Deberá ser un archivo válido (JPG, JPEG, PNG, GIF).**

- **(Opcional) Tablas secundarias**

- **Verificar que los valores existan en base. Es decir, que los valores de talles, colores, etc. que lleguen sean válidos en la base.**

- **(Opcional) Resto de los formularios del sitio**

Ver para mejorar:
 Que al devolver los errores de validacion, se muestren los campos ingresados por el req.body de los desplegables de marcas y tipo de producto, etc.
 Mejorar estilos de errores, de forma local y con color rojo y verde.

Entregable: implementación de Express Validator en las rutas que reciban formularios.

4. **Validaciones del front-end**

Ya lo dijimos muchas veces y vale repetir: si bien no podemos confiar en las validaciones del front-end porque nuestro visitante puede deshabilitarlas, en el 99% de los casos funcionarán como esperamos.

Nuestro objetivo acá es doble:

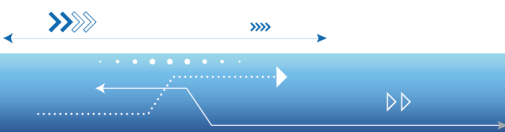
1. Por un lado, vamos a validar los datos antes de enviarlos para evitarle procesos innecesarios al servidor.
2. Por el otro, vamos a darle feedback inmediato y de calidad a nuestro visitante. Esto se traduce en una mucho mejor experiencia (UX) dentro de nuestro sitio.

Vamos con esas consignas:



- **Registro de usuarios**
 - Nombre y apellido
 - Obligatorio.
 - Deberá tener al menos 2 caracteres.
 - Email
 - Obligatorio.
 - Deberá ser válido.
 - (Opcional) → No puede repetirse con los e-mails ya registrados.
 - Contraseña
 - Obligatoria.
 - Deberá tener al menos 8 caracteres.
 - (Opcional) → Deberá tener letras mayúsculas, minúsculas, un número y un carácter especial.
 - Imagen
 - Deberá ser un archivo válido (JPG, JPEG, PNG, GIF).
- **Login de usuarios** (este ya lo deberían tener de sprints anteriores 😊👍)
 - Email
 - Obligatorio.
 - Deberá ser válido.
 - (Opcional) → Debe existir en la base.
 - Contraseña
 - Obligatoria.
- **Creación y modificación de productos**
 - Nombre
 - Obligatorio.
 - Deberá tener al menos 5 caracteres.
 - Descripción
 - Deberá tener al menos 20 caracteres.
 - Imagen
 - Deberá ser un archivo válido (JPG, JPEG, PNG, GIF).
- **(Opcional) Resto de los formularios del sitio**

Entregable: implementación de **validator.js** ([documentación](#)/[descarga](#)) o validaciones "custom" en los formularios mencionados.



> Resumen de entregables

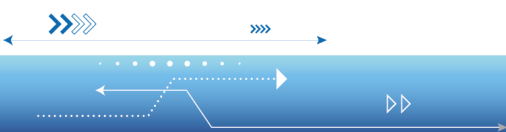
- Archivo **retro.md** con el resultado de la retrospectiva.
- (Opcional) Archivo **daily.md** con sus opiniones sobre las dailys/weeklys
- Tablero de trabajo actualizado.
- Validaciones del back-end con **Express Validator**:
 - Registro de usuarios.
 - Login de usuarios.
 - Creación y modificación de productos.
 - (Opcional) Resto de los formularios del sitio.
- Validaciones del front-end con **JavaScript**:
 - Registro de usuarios.
 - Login de usuarios.
 - Creación y modificación de productos.
 - (Opcional) Resto de los formularios del sitio.

> Cierre

Las validaciones son vitales si queremos que nuestro sitio funcione bien a lo largo del tiempo. Ya sea por error, o por mala intención, siempre llegarán datos que son inválidos o que de alguna manera son incorrectos. 🤖 🤖 🤖 Si nos descuidamos y dejamos que pasen, va a ser muy difícil que podamos corregir el problema sin terminar con un gran dolor de cabeza. 🔥 🚒 🔥

Poquito a poquito y sin darse cuenta están terminando su sitio...

- 🔍 Investigación → Lista ✓
- 🎨 Diseño → Listo ✓
- 🌐 HTML + CSS → Listo ✓
- ⚙️ Vistas dinámicas → Listas ✓
- 📁 Base de datos → Lista ✓
- 👤 Usuarios con su login y registro → Listo ✓
- 📦 Productos con su CRUD completo → Listo ✓
- 🛡️ Validaciones → En camino ⌚



¡Uf! ¡Cuánto camino recorrido!

Mirando atrás hay que reconocer sin duda que no ha sido un camino fácil 🧑🏫, piensen en todos los conceptos que hemos visto en tan solo algunos meses. Seguro a esta altura tengan la cabeza dando vueltas 🤖 y sueñen con objetos literales en lugar de ovejitas. 🐑 🐑 🐑 🐑

Como dice el dicho: “Nada que valga la pena es fácil de conseguir”, así que felicitaciones por llegar hasta aquí. **¡Vamos por este penúltimo sprint!**

admin

- permitir crear /editar prod. -ok
- no muestra carrito. -ok
- permite eliminar producto -ok

usuario reg

- No permitir crear /editar prod. -ok
- Muestra carrito. -ok
- NO permite eliminar producto - FALTA

usuario no reg.

- no puede añadir al carrito -ok
- No permitir crear/edita/ prod -ok
- No pueda eliminar producto -- FALTA
- No puede comprar - ok.

