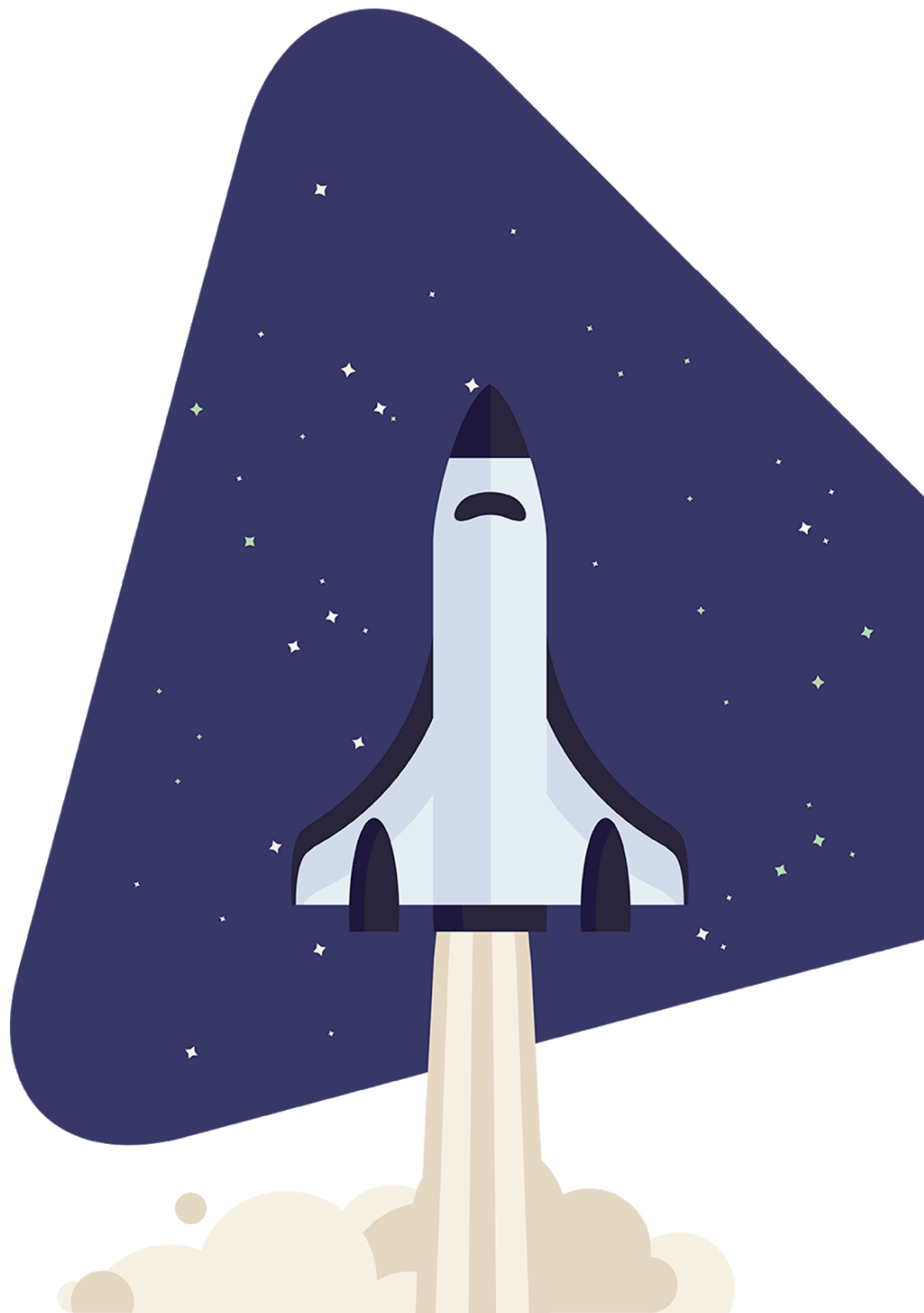


Become a Software Developer

Problemática 05



Descripción del problema a resolver

La problemática de la semana 05 consiste en agregar una página de suscripción al diario online desarrollado en la semana 03. Manteniendo el repositorio original, agregar un nuevo archivo html que mantenga la estética del diario, su cabecera y pie de página, y que contenga un formulario de suscripción para lectores. El formulario debe contener los siguientes campos: Nombre completo, email, contraseña, repetir contraseña, edad, teléfono, dirección, ciudad, código postal y DNI. Además debe contar con un botón "Enviar". El diseño debe ser responsive, de modo que si se visualiza el formulario desde un celular se deben mostrar los campos uno abajo del otro, pero si se ve desde un monitor de PC los campos se deben mostrar separados en dos columnas, con el botón enviar solo al final del formulario y centrado en medio de las dos columnas.

Se debe validar cada campo y mostrar un mensaje de error descriptivo abajo del campo que fallo. Realizar las siguientes validaciones:

- Nombre completo: Debe tener más de 6 letras y al menos un espacio entre medio.
- Email: debe tener un formato de email válido.
- Contraseña: Al menos 8 caracteres, formados por letras y números.
- Edad: Número entero mayor o igual a 18.
- Teléfono: Número de al menos 7 dígitos, no aceptar espacios, guiones ni paréntesis.
- Dirección: Al menos 5 caracteres, con letras, números y un espacio en el medio.
- Ciudad: Al menos 3 caracteres.
- Código Postal: Al menos 3 caracteres.
- DNI: Número de 7 u 8 dígitos.

La validación de cada campo se debe realizar en el evento "blur" de cada uno de los campos. Además, si algún campo tiene un error de validación, en el evento "focus" de dicho campo debe desaparecer el mensaje porque se asume que el usuario está corrigiendo el error.

Al presionar el botón "Enviar" se debe mostrar un cartel emergente con la información cargada en el formulario en caso que haya pasado todas las validaciones. Si alguna validación no pasó, además de mostrar el error debajo del campo, también se debe mostrar el error en el cartel emergente.

Bonus: Si pudieron resolver todo lo anterior y les queda tiempo, agregar un título al formulario que diga "HOLA" y que a medida que se edita el campo nombre completo, se vaya modificando en tiempo real el título del formulario. Por ejemplo, si el nombre es JUAN PEREZ, el título debe decir "HOLA JUAN PÉREZ" e ir modificándose en tiempo real a medida que se presionan teclas (utilizar evento keydown y focus)

Subir todos los cambios a Github y responder la problemática con la URL de archivo en Github.



Pasos a seguir:

- 1) Continuando con el proyecto del diario realizado en la Semana 03 crear un nuevo archivo subscription.html.
- 2) Agregar el código HTML para que tenga la estructura solicitada en la problemática semanal.
- 3) Agregar el código CSS necesario para que el formulario siga la estética del diario y que su diseño sea responsive.
- 4) Guardar los cambios en git, subirlos a Github y verificar que la página se visualiza correctamente en Github Pages.
- 5) Agregar el código JavaScript necesario para capturar el evento "blur" de cada campo, validar el contenido y mostrar el mensaje de error correspondiente.
- 6) Agregar el código JavaScript necesario para capturar el evento "focus" de cada campo y ocultar el mensaje de error si es que había uno.
- 7) Agregar el código JavaScript necesario para mostrar un cartel al presionar el botón "Enviar".
- 8) Agregar el código JavaScript necesario para editar en tiempo real el título del formulario.
- 9) Hacer commits con el progreso y subir todos los cambios siempre verificando que se vea correctamente en Github Pages.
- 10) Responder a la Problemática Semanal 05 de Classroom con el link de Github de archivo html creado para el formulario de suscripción.

Se evaluará:

1. Solución de la problemática planteada; al Github Pages debe visualizarse correctamente el formulario de suscripción con las validaciones y el cartel emergente.
2. Correcto uso de los eventos y listener para cada situación.
3. Comprensión de la problemática a resolver.
4. Prolijidad y consistencia del código escrito. Todo el código y los comentarios deben ser escritos en inglés.
5. Correcta uso de Git, Github, Documento Readme.



6. Entrega en tiempo y forma.
7. Participación activa con el grupo de trabajo, preguntando a tutores y ayudando a compañeros.

