

Historia de usuario #2

Anexo de Documentos Relacionados:

- APIDog. (s.f.). *REST API URL best practices & examples*. APIDog. <https://apidog.com/blog/rest-api-url-best-practices-examples/>

Descripción conceptual

Módulo	Gestión de Viajes.
Descripción de la(s) funcionalidad(es) requerida(s):	Permitir a los conductores publicar viajes con detalles como origen, destino, horario, precio y número de asientos.

Backend

URL	Método	Código html
https://localhost:8080/api/trips	POST	201 (Creado), 400 (Solicitud Incorrecta)
Caso de uso tecnico Este endpoint permite a los conductores publicar viajes. Si los datos son válidos, retorna 201 con el ID del viaje creado. Si hay errores, retorna 404 con un cuerpo vacío.		
Datos de entrada { "origin": "Universidad Nacional", "destination": "Calle 45", "schedule": "2025-02-01T08:00:00", "seats": 3, "price": 10000 }	Datos de salida 201: { "status": "success", "data": { "trip_id": 1, "origin": "Universidad Nacional", "destination": "Calle 45", "seats": 3 } } 404: { "status": "error", "message": "Missing origin or destination" }	

Frontend

El formulario para publicar un viaje debe permitir al conductor ingresar los detalles necesarios de manera clara y organizada. Debe incluir campos para especificar el origen y destino del viaje, un selector de fecha y hora para el horario, un campo numérico para el precio, y un control deslizante o selector para el número de asientos, limitado entre 1 y 5. Además, el formulario debe mostrar validaciones dinámicas en tiempo real, resaltando en rojo cualquier campo con errores o vacío.

El botón "Publicar" debe permanecer deshabilitado hasta que todos los campos estén correctamente llenados, cambiando su estado a viaje publicado con éxito cuando la información sea válida. De lo contrario se mostraran en rojo los campos que faltan por completar

Interacción esperada:

- Formulario para que el conductor ingrese los detalles del viaje.
- Botón "Publicar" para enviar los datos al servidor.
- Visualización del mensaje de estado para indicar el estado final de la publicación.

Mockups/Prototipos:

The image displays two side-by-side mobile application mockups for a travel booking interface. Both mockups are set within a light gray frame with black bars on the left and right sides, representing the phone's bezels.

Left Mockup (Form State):

- Origen:** A text input field with the placeholder text "Value".
- Destino:** A text input field with the placeholder text "Value".
- Horario:** A text input field with the placeholder text "Value".
- Precio:** A text input field with the placeholder text "Value".
- # Asientos:** A text input field with the placeholder text "Value".
- Publicar:** A green button with the text "Publicar".

Right Mockup (Success State):

- Origen:** A green button with the text "UNAL".
- Destino:** A green button with the text "CALLE 45".
- Horario:** A green button with the text "28/01/2025 - 6:00 PM".
- Precio:** A green button with the text "10.000".
- # Asientos:** A green button with the text "3".
- Message:** A green button with a white exclamation mark icon and the text "VIAJE PUBLICADO CON EXITO".

A mobile application interface for booking a trip. The form consists of five input fields, each with a label above it. The 'Origen' field contains 'UNAL' in a green box. The 'Destino' field contains 'VALUE' in a red box. The 'Horario' field contains '28/01/2025 - 6:00 PM' in a green box. The 'Precio' field contains '10.000' in a green box. The '# Asientos' field contains 'VALUE' in a red box. At the bottom, there is a red rounded rectangle containing a white exclamation mark icon and the text 'TODOS LOS CAMPOS DEBEN SER LLENADOS'.

Origen

UNAL

Destino

VALUE

Horario

28/01/2025 - 6:00 PM

Precio

10.000

Asientos

VALUE

! TODOS LOS CAMPOS DEBEN SER LLENADOS

Flujo visual y eventos:

1. El conductor llena los campos del formulario y presiona "Publicar".
2. Si la publicación es exitosa, aparece un mensaje de éxito.
3. Si hay errores, se resaltan los campos incorrectos.