

Historia de usuario #3

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 Notificaciones. |
| Descripción de la(s) funcionalidad(es) requerida(s): | Enviar notificaciones en tiempo real a los usuarios sobre eventos relacionados con los viajes (ejemplo, solicitud aceptada, cancelación, mensajes recibidos). |

Backend

| | | |
|--|--|--|
| URL https://localhost:8080/api/notifications | Método POST | Código html 200 (OK), 500 (Error del servidor) |
| Caso de uso tecnico Este endpoint envía notificaciones en tiempo real a los usuarios. Si el proceso es exitoso, retorna 200 con el mensaje enviado. Si ocurre un error, retorna 500 con un cuerpo vacío. | | |
| Datos de entrada { "user_id": 1, "message": "Tu solicitud de viaje ha sido aceptada." } | Datos de salida 201: { "status": "success", "message": "Notificación enviada." } 500: { "status": "error", "message": "Failed to send notification" } | |

Frontend

El apartado de notificaciones debe mostrar un listado de eventos recientes relacionados con los viajes del usuario, organizados cronológicamente y con la opción de expandir cada notificación para ver detalles y acceder al módulo correspondiente.

Interacción esperada:

- Notificaciones emergentes (pop-ups o banners) aparecen automáticamente en la interfaz del usuario cuando se generan eventos en el sistema.

- El usuario puede interactuar con la notificación para redirigirse al módulo correspondiente (ejemplo, detalles del viaje).

Mockups/Prototipos:



Flujo visual y eventos:

1. El sistema genera una notificación cuando ocurre un evento (ejemplo., aceptación de solicitud).
2. La notificación se muestra al usuario como un banner emergente en tiempo real.
3. El usuario puede hacer clic en la notificación para navegar al módulo correspondiente.