

WIDGETS 1.0 - Errores #8559

correccion ordenado de partidos en Agenda

2026-03-03 15:52 - Anielkis Herrera

Estado:	Resuelta	Fecha de inicio:	2026-03-03
Prioridad:	Normal	Fecha fin:	
Asignado a:	Anielkis Herrera	% Realizado:	0%
Categoría:	Desarrollo	Tiempo estimado:	0.00 hora
Versión prevista:	Sprint 11	Tiempo dedicado:	0.00 hora
App related:			

Descripción

Los partidos en la agenda no se ordenan correctamente.

Ejemplo:



Histórico

#1 - 2026-03-03 15:59 - Anielkis Herrera

- Estado cambiado Nueva por Resuelta

El problema fue ocasionado porque la agenda en ese canal específico no se había refrescado y puesto la hora de cada partido.

se sube corrección en commit <https://git.datafactory.la/datafactory/df-components/commit/6b4284ffb08107113fe115c0ed93c580df9924bc>

La lógica queda así:

Descripción General

La lógica de ordenamiento clasifica los partidos de forma cronológica ascendente (los más próximos primero) utilizando dos campos principales con un mecanismo de respaldo.

Campos Utilizados

- 1. `date` (campo primario)
 - Formato: YYYY-MM-DD (ej: 2026-03-15)
 - Determina el día del partido
- 2. `scheduledStart` (campo secundario)
 - Formato: Hora dentro del día (ej: 15:00:00)
 - Determina la hora de inicio del partido

Fuentes de Datos

La lógica utiliza dos fuentes para obtener la información de ordenamiento:

- 1. Fuente Primaria: Datos del endpoint de agenda (schedule)
 - Contiene: date, scheduledStart, matchId, etc.
- 2. Fuente de Respaldo: Datos del endpoint de partido almacenados en matchStore.matches[matchId]
 - Se usa cuando scheduledStart viene vacío desde la agenda

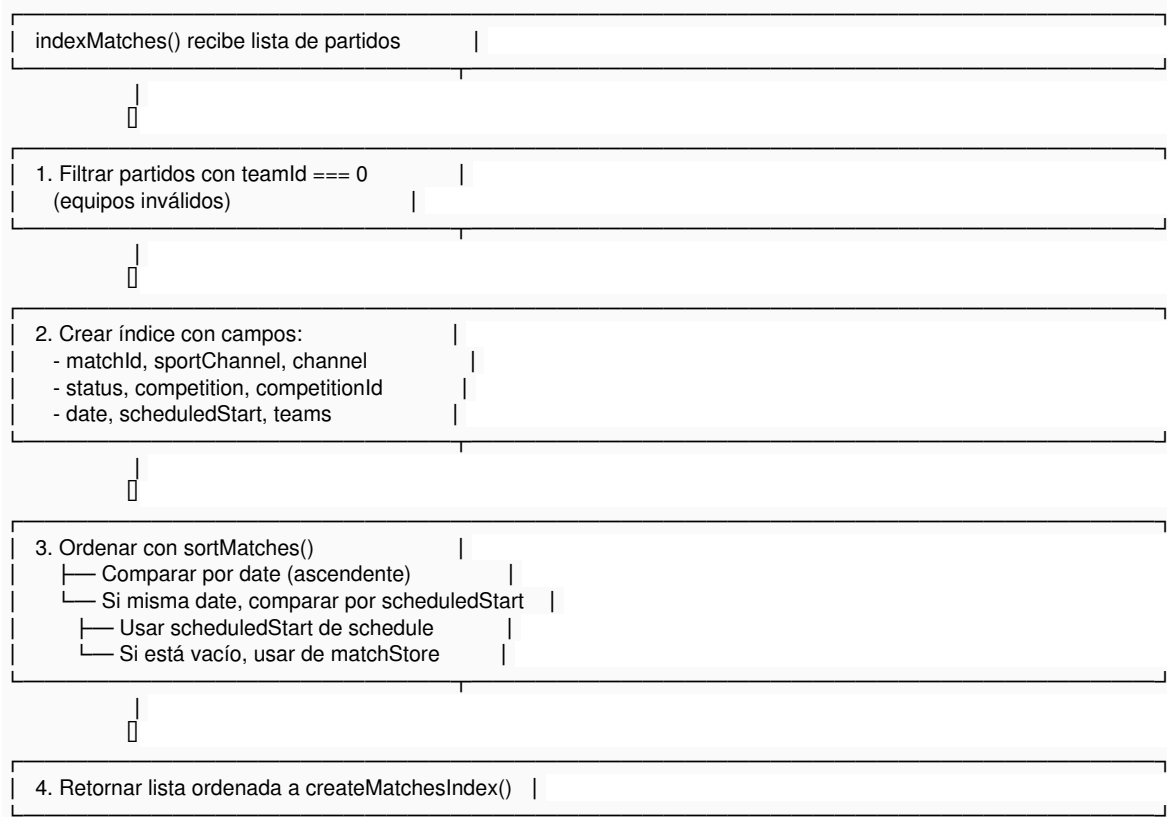
Algoritmo de Ordenamiento

```
// Paso 1: Comparar fechas primero
const dateCompare = (a.date || '').localeCompare(b.date || '')
if (dateCompare !== 0) return dateCompare
// Si las fechas son diferentes, retorna el resultado

// Paso 2: Si misma fecha, comparar horas de inicio
```

```
const aScheduledStart = a.scheduledStart || matchStore.matches[a.matchId]?.scheduledStart || "  
const bScheduledStart = b.scheduledStart || matchStore.matches[b.matchId]?.scheduledStart || "  
const timeCompare = aScheduledStart.localeCompare(bScheduledStart)  
return timeCompare
```

Flujo de Ejecución



Casos Especiales

Caso	Comportamiento
Ambos campos vacíos	El partido se ordena al final de la lista
Solo `date` presente	Se ordena por fecha; dentro del mismo día, va al final
Solo `scheduledStart` presente	Compara como cadena vacía en date, luego usa la hora
Misma fecha y hora	Mantiene orden estable (orden original preservado)
`matchStore` no tiene el partido	Usa cadena vacía " como fallback

Ejemplo Práctico

Entrada (sin ordenar):

```
[  
  { matchId: 1, date: '2026-03-15', scheduledStart: '20:00:00' },  
  { matchId: 2, date: '2026-03-14', scheduledStart: "" }, // vacío  
  { matchId: 3, date: '2026-03-15', scheduledStart: '15:00:00' },  
  { matchId: 4, date: '2026-03-14', scheduledStart: '18:00:00' }  
]
```

matchStore (respaldo):

```
{
  matches: {
    2: { scheduledStart: '19:00:00' } // Tiene el valor que falta en schedule
  }
}
```

Salida (ordenada):

```
[
  { matchId: 4, date: '2026-03-14', scheduledStart: '18:00:00' }, // 14 de marzo, 18:00
  { matchId: 2, date: '2026-03-14', scheduledStart: " " }, // 14 de marzo, 19:00 (usó respaldo)
  { matchId: 3, date: '2026-03-15', scheduledStart: '15:00:00' }, // 15 de marzo, 15:00
  { matchId: 1, date: '2026-03-15', scheduledStart: '20:00:00' } // 15 de marzo, 20:00
]
```

Ubicación en el Código

Archivo: src/utls/sortMatches.js

Función exportada:

```
1 export function sortMatches(matches, matchStore)
```

Parámetros:

- matches: Array de partidos a ordenar
- matchStore: Store de Pinia con datos de partidos (para fallback)

Retorna: El mismo array ordenado (sort in-place)

Integración en TickerWidget

```
// En TickerWidget.vue
const indexMatches = (matches, store = matchStore) => {
  const filteredMatches = matches.filter(/* filtrar teamId === 0 */)

  const indexedMatches = filteredMatches.map(/* crear índice */)

  // Aplicar ordenamiento
  return sortMatches(indexedMatches, store)
}
```

Esta lógica asegura que los partidos siempre se muestren en orden cronológico, incluso cuando la API de agenda no proporciona completa la información de scheduledStart.

Ficheros

Screenshot_20260303_132939.jpeg	24,913 KB	2026-03-03	Anielkis Herrera
---------------------------------	-----------	------------	------------------