

TDZ Fix: Finanzas SD Cost Catalog Route

Finanzas SD – Architecture, Flows & SOPs

Arquitectura, Flujos y Procedimientos

January 15, 2026

1 TDZ Fix: Finanzas SD Cost Catalog Route

1.1 Issue Summary

Route: /finanzas/sdmt/cost/catalog

Error: ReferenceError: Cannot access 'B' before initialization

Component: src/features/sdmt/cost/Catalog/SDMTCatalog.tsx

Date Fixed: 2025-11-17

1.2 Root Cause

The component had a **Temporal Dead Zone (TDZ) violation** where a variable was referenced before its declaration:

```
[] // BEFORE (lines 83-87): Using lineItemsError BEFORE declaration const lineItemsErrorMessage = lineItemsError ? lineItemsError instanceof Error ? lineItemsError.message : String(lineItemsError) : "";
// Line 114: lineItemsError declared here (TOO LATE!) const { error: lineItemsError, ... } = useProjectLineItems();
```

1.3 Solution

Moved the useProjectLineItems() hook call to execute **before** any usage of its returned values:

```
[] // AFTER: Hook called first (lines 105-111) const { lineItems: queryLineItems, isLoading: isLineItemsLoading, isFetching: isLineItemsFetching, error: lineItemsError, invalidate: invalidateLineItems, } = useProjectLineItems();
// Then compute dependent values (lines 113-118) const lineItemsErrorMessage = lineItemsError ? lineItemsError instanceof Error ? lineItemsError.message : String(lineItemsError) : "";
```

1.4 Why This Happened

JavaScript/TypeScript's const and let declarations have **block scope** with a TDZ, meaning: - The variable exists in the scope from the start - But it cannot be accessed until the line where it's declared and initialized - Accessing it before declaration throws ReferenceError: Cannot access 'X' before initialization

In minified code, variable names are shortened (e.g., lineItemsError → B), which is why the error message referenced variable B.

1.5 Prevention Guidelines

To avoid similar TDZ issues in React components:

1. **Declare all hooks at the top** of the component function, following React's Rules of Hooks
2. **Order declarations properly:**
 - State hooks (useState, etc.)
 - Context hooks (useContext, custom hooks like useProject())
 - Computed values that depend on hook results
3. **Never reference a variable before its declaration line**
4. **Be careful with destructuring** - ensure the source is declared first
5. **Watch for circular dependencies** between modules (though not the issue here)

1.6 Files Modified

- src/features/sdmt/cost/Catalog/SDMTCatalog.tsx: Reordered hook calls and variable declarations

1.7 Verification

- Build succeeds without errors
 - TypeScript type-checking passes (no new errors)
 - ESLint passes with no warnings
 - Route renders without runtime errors
-

2 TDZ Fix: Finanzas SD Cost Forecast Route - “Cuadrícula de Pronóstico 12 Meses”

2.1 Issue Summary

Route: /finanzas/sdmt/cost/forecast

Error: ReferenceError: Cannot access 'X' before initialization

Component: src/features/sdmt/cost/Forecast/components/ForecastRubrosTable.tsx

Date Fixed: 2026-01-11

Trigger: Clicking “Cuadrícula de Pronóstico 12 Meses” (12-Month Forecast Grid) to expand the collapsible section

2.2 Root Cause

The ForecastRubrosTable component had a **Temporal Dead Zone (TDZ) violation** where a function was called before its declaration:

```
[] // BEFORE: Function called inside useMemo (line 192) const visibleCategories
= useMemo(() => { // ... filtering logic ... const recalculatedTotals = recalculate-
CategoryTotals(filteredRubros); // Called here // ... }, [searchFilter, categoryTotals,
categoryRubros, filterMode]);
// Line 201: Function declared here (TOO LATE!) const recalculateCategoryTotals
= (rubros: CategoryRubro[]): CategoryTotals => { // ... implementation ... };
```

When the `visibleCategories` `useMemo` executes, it tries to call `recalculateCategoryTotals`, but that function hasn't been initialized yet because it's declared later in the code.

In minified production builds, `recalculateCategoryTotals` gets shortened to a single letter (e.g., X), causing the error:

```
ReferenceError: Cannot access 'X' before initialization
```

2.3 Solution

Moved the `recalculateCategoryTotals` function declaration **before** the `visibleCategories` `useMemo`:

```
[] // AFTER: Function declared first (line 159) const recalculatCategoryTotals =
(rubros: CategoryRubro[]): CategoryTotals => { // ... implementation ... };
// Then used in useMemo (line 203+) const visibleCategories = useMemo(() => { // ...
... filtering logic ... const recalculatedTotals = recalculatCategoryTotals(filteredRubros);
// Now safe // ... }, [searchFilter, categoryTotals, categoryRubros, filterMode]);
```

2.4 Why This Happened

JavaScript/TypeScript `const` and `let` declarations are hoisted but remain in the **Temporal Dead Zone (TDZ)** until the execution reaches the declaration line:

1. The variable name is reserved in scope from the start
2. But it cannot be accessed until the line where it's declared and initialized
3. Accessing it before declaration throws `ReferenceError: Cannot access 'X' before initialization`
4. In production builds, variable names are minified (e.g., `recalculateCategoryTotals` → X)

The issue only appeared when expanding the “Cuadrícula de Pronóstico 12 Meses” section because: - That's when ForecastRubrosTable component is first rendered - The visibleCategories useMemo executes immediately on mount - It tries to call recalculateCategoryTotals which is still in TDZ

2.5 Prevention Guidelines

To avoid similar TDZ issues in React components:

1. **Declare helper functions BEFORE they're used** in useMemo/useEffect/useCallback
2. **Order declarations properly:**
 - Imports
 - Type definitions
 - Helper functions (that don't depend on state/props)
 - State hooks (useState, etc.)
 - Context hooks (useContext, custom hooks)
 - Computed values (useMemo, useCallback) that use the helpers
3. **Never reference a const/let before its declaration line**
4. **Consider moving complex helper functions outside the component** if they don't need closure over state/props
5. **Use ESLint rules** to catch potential TDZ issues during development

2.6 Files Modified

- src/features/sdmt/cost/Forecast/components/ForecastRubrosTable.tsx: Moved recalculateCategoryTotals function declaration before visibleCategories useMemo
- docs/notes-finanzas-tdz.md: Updated documentation with this case

2.7 Verification

- Build succeeds without errors
- TypeScript type-checking passes
- ESLint passes with no warnings
- Route loads without errors
- Clicking “Cuadrícula de Pronóstico 12 Meses” renders without TDZ error
- Component can be dynamically imported without throwing