Visual Comparison: Before and After Fix

Finanzas SD – Architecture, Flows & SOPs

Arquitectura, Flujos y Procedimientos

January 18, 2026

1 Visual Comparison: Before and After Fix

1.1 The Problem

When ungrouped Cognito users tried to access the application after PR #577, they encountered a critical error.

1.2 Before Fix

1.2.1 Application State:

- **Error Screen**: Red error banner with "Application Error"

- **Error Message**:

```
Something unexpected happened while running the application.
The error details are shown below. Contact support if this issue persists.

Error Details:
DEFAULT_ROLE is not defined
```

1.2.2 Browser Console:

```
[] Uncaught ReferenceError: DEFAULT_ROLE is not defined  at AuthProvider.tsx:210
at AuthProvider.tsx:237  at AuthProvider.tsx:307 ... (multiple stack traces)
```

1.2.3 Impact:

- □ Application completely unusable
- □ No way to proceed or recover
- □ Generic error message not helpful to users
- □ No guidance on what to do next

1.3 After Fix

1.3.1 Application State:

- **NoAccess Screen**: Clean, professional "no permissions" screen

- **Message**:

```
Sin permisos asignados
(No permissions assigned)

No tienes permisos asignados para acceder a esta aplicación.

Tu cuenta está autenticada pero no tiene grupos de Cognito asociados
que otorguen acceso a los módulos de esta aplicación.

Para obtener acceso, contacta al administrador del sistema.

[Cerrar sesión]
```

1.3.2 Browser Console:

```
[] [Router] No role assigned - user has no access { user: {…}, availableRoles: [],
groups: [] }
```

1.3.3 Impact:

- [] Application handles the situation gracefully
- [] User sees clear, actionable message in Spanish
- [] User can sign out and try different account
- [] Administrators can identify the issue from logs
- [] Security maintained (no implicit access granted)

1.4 Code Changes That Made This Possible

1.4.1 Key Changes in AuthProvider.tsx:

1. **Before (Line 210)**:

   ```
   [] const effectiveRole = currentRole || DEFAULT_ROLE; // [] DEFAULT_ROLE unde-
   fined
   ```

   After (Line 210-219):

   ```
   [] if (!currentRole) { setRouteConfigMissing(true); console.warn("[Router] No role
   assigned - user has no access", {…}); return; // [] Safe early return }
   ```

2. **Before (Line 307)**:

[] setCurrentRole(DEFAULT_ROLE); // ⬜ *Sets undefined*

After (Line 328):

[] setCurrentRole(**null**); // ⬜ *Properly clears role*

3. **Before (Line 495)**:

[] **return** canAccessRoute(route, currentRole || DEFAULT_ROLE); // ⬜ *Passes undefined*

After (Line 516-522):

[] **if** (!currentRole) { **return false**; // ⬜ *Safe null check* } **return** canAccessRoute(route, currentRole);

1.5 User Experience Comparison

Aspect	Before Fix	After Fix
Error Handling	⬜ Application crash	⬜ Graceful degradation
User Message	⬜ Technical error	⬜ Clear Spanish message
User Action	⬜ Contact support	⬜ Sign out or contact admin
Security	⚠ Undefined behavior	⬜ No access granted
Logging	⬜ Stack traces only	⬜ Clear warning logs
Recovery	⬜ Requires page refresh	⬜ Clean sign out

1.6 How to Test

1.6.1 Setup:

1. Create a Cognito user account
2. Do NOT assign any groups to the user
3. Attempt to login to the application

1.6.2 Expected Results After Fix:

1. ☐ Login succeeds (Cognito authentication works)
2. ☐ User sees NoAccess screen (not error)
3. ☐ Message explains the situation
4. ☐ Sign out button works correctly
5. ☐ Console shows warning (not error)

1.6.3 Verification:

- Check browser console for the warning message
- Verify no error stack traces appear
- Confirm "Cerrar sesión" button redirects to login
- Test with user who HAS groups (should work normally)

1.7 Security Notes

This fix maintains the security improvements from PR #577:

- **No implicit access**: Users without recognized groups get NO access
- **No default role**: No EXEC_RO or other role assigned by default
- **Clear audit trail**: Console warnings provide visibility
- **Graceful degradation**: Security maintained without breaking UX

The key difference is: - **Before PR #577**: Ungrouped users got EXEC_RO (read-only access) - **After PR #577**: Ungrouped users get no access, but crashed - **After this fix**: Ungrouped users get no access, with clear message