



Xero Integration Connection Fix



Issues Fixed

1. Database Foreign Key Constraint Issue

Problem: The Xero integration was failing to save tokens due to a foreign key constraint violation in the `createdById` field.

Root Cause: The `XeroService.saveTokensToDatabase()` method was hardcoding `'system'` as the `createdById`, but no user with that ID existed in the database.

Fix Applied:

- Updated `XeroService` constructor to accept `userId` parameter
- Modified `saveTokensToDatabase()` to use the actual user ID instead of `'system'`
- Updated all API endpoints to pass the current user's ID to `XeroService`

2. Missing Error Handling & Logging

Problem: Limited visibility into connection failures and authentication issues.

Fix Applied:

- Added comprehensive logging to track the OAuth flow
- Enhanced error handling in callback processing
- Added detailed status information in API responses

3. Redirect URI Configuration

Problem: Potential issues with redirect URI configuration for different environments.

Fix Applied:

- Updated Xero config to use `NEXTAUTH_URL` as fallback for redirect URI
- Better handling of environment-specific URLs



How to Test the Fix

Step 1: Check Xero Status

1. Log into the application
2. Go to **Settings** page
3. Find the **Xero Integration** section
4. Click **Check Status** or view the integration status

Step 2: Connect to Xero

1. Click **"Connect to Xero"** button
2. You should be redirected to Xero's authorization page
3. Log into your Xero account and authorize the connection
4. You'll be redirected back to the settings page with a success/error message

Step 3: Verify Connection

1. The status should show “Connected” with your organization name
2. Try the “**Test Connection**” button to verify API access
3. Check if sync features are working

Troubleshooting Guide

Issue: “Failed to save integration settings”

Cause: Database constraint or permission issue

Solution:

- Check that you’re logged in with a valid user account
- Verify database connection is working
- Check server logs for specific database errors

Issue: “Xero credentials not configured”

Cause: Missing environment variables

Solution: Verify these environment variables are set in `.env` :

```
XERO_CLIENT_ID="71C201203893448AA87867F089F938B2"  
XERO_CLIENT_SECRET="C3g9DQQfddSRmG4wKkcabTC-MeDf710IOM0Y9P171XRD_RvQ"  
XERO_REDIRECT_URI="http://localhost:3000/api/xero/callback"  
XERO_SCOPES="accounting.transactions accounting.contacts accounting.settings"
```

Issue: “Failed to connect to Xero: Invalid or expired tokens”

Cause: Token refresh failed or tokens are corrupted

Solution:

- Try disconnecting and reconnecting to Xero
- Clear any existing Xero integration records from database
- Check Xero app permissions in your Xero developer console

Issue: “No Xero organisation found”

Cause: User doesn’t have access to any Xero organizations

Solution:

- Ensure you’re logging into the correct Xero account
- Verify you have appropriate permissions in the Xero organization
- Check that the organization is active

Technical Details

Files Modified:

- `lib/xero-service.ts` - Added user ID parameter and better error handling
- `lib/xero-config.ts` - Improved redirect URI handling
- `app/api/xero/auth/route.ts` - Added configuration validation
- `app/api/xero/callback/route.ts` - Enhanced logging and error handling

- `app/api/xero/status/route.ts` - Updated to pass user ID
- `app/api/xero/test/route.ts` - Updated to pass user ID

Database Schema:

The `XeroIntegration` table structure:

```
model XeroIntegration {
  id          String    @id
  tenantId    String    @unique
  tenantName  String?
  accessToken  String
  refreshToken String
  expiresAt   DateTime
  scopes      String[]
  isActive    Boolean   @default(true)
  connectedAt DateTime   @default(now())
  lastSyncAt  DateTime?
  createdById String     -- Now properly references User.id
  User        User       @relation(fields: [createdById], references: [id])
}
```

OAuth Flow:

1. User clicks "Connect to Xero"
2. `/api/xero/auth` generates authorization URL
3. User authorizes on Xero's website
4. Xero redirects to `/api/xero/callback` with authorization code
5. Service exchanges code for access/refresh tokens
6. Tokens are saved to database with proper user association
7. Connection is tested immediately
8. User is redirected back with success/error status

Server Logs

When testing, check the server console for detailed logs:

- "Generated Xero auth URL: ..." - Authorization URL created
- "Processing Xero callback with code: ..." - OAuth callback received
- "Token response received: ..." - Token exchange status
- "Tenants found: X" - Number of Xero organizations found
- "Saving tokens to database..." - Database save attempt
- "Xero connection test result: ..." - Final connection test

Expected Behavior After Fix

1. **Successful Connection:** Status shows "Connected" with organization name
2. **Token Storage:** Tokens are properly saved in `XeroIntegration` table
3. **Test Connection:** API calls to Xero work successfully
4. **Auto-Refresh:** Tokens are automatically refreshed when needed

5. **Error Handling:** Clear error messages for any connection issues

The Xero integration should now work reliably for connecting, storing credentials, and maintaining the connection with proper token refresh handling.