PSScript Application Fixes

This document outlines the fixes implemented to address two critical issues in the PSScript application:

1. **Script Deletion Error**: When clicking the trash can icon to delete a script, an error message "Failed to delete script(s). Please try again." was displayed.
2. **Network Error During Upload**: When attempting to upload a script file, a "Network error. Please check your connection." message was displayed.

Implemented Fixes

1. Script Deletion Error Fix

The script deletion error was caused by improper error handling in the backend controller and insufficient transaction management. The following changes were made:

Backend Changes:

- Enhanced the `deleteScript` method in `ScriptController.ts` to properly handle transaction rollbacks
- Added proper error responses with success flags
- Improved error handling for transaction rollbacks
- Added more detailed error messages for different failure scenarios

Frontend Changes:

- Updated the `deleteScript` method in `api.ts` to provide more specific error messages
- Enhanced the delete script mutation in `ScriptManagement.tsx` to properly handle success and error cases
- Added user-friendly error messages when deletion fails

2. Network Error During Upload Fix

The network error during upload was caused by CORS issues and insufficient error handling. The following changes were made:

Backend Changes:

- Enhanced the CORS middleware in `corsMiddleware.ts` to properly handle file uploads
- Added support for credentials and proper origin handling
- Implemented a network error handling middleware in `uploadMiddleware.ts`
- Added timeout handling for upload requests

- Improved error responses with more detailed information

Frontend Changes:

- Enhanced the upload error handling in the API service

Testing the Fixes

To test these fixes, restart the application using the provided scripts:

```
# Restart just the backend
./restart-backend.sh

# Restart just the frontend
./restart-frontend.sh

# Restart both services
./restart-all.sh
```

After restarting, you should be able to: 1. Delete scripts by clicking the trash can icon without errors 2. Upload new script files without network errors

Technical Details

Script Deletion Fix

The key improvement in the script deletion process is proper transaction management. Now, if any part of the deletion process fails (such as deleting related records), the entire transaction is rolled back, and a clear error message is returned to the user.

```
// Rollback transaction if there was an error
if (transaction) {
  try {
    await transaction.rollback();
  } catch (rollbackError) {
    console.error('Error rolling back transaction:', rollbackErro
  }
}

// Return a structured error response
res.status(500).json({
  message: 'Failed to delete script',
  error: error.message,
  success: false
});
```

Upload Error Fix

The upload process now includes better CORS handling and network error management:

```
 // Middleware to handle network errors during upload
const handleNetworkErrors = (req: Request, res: Response, next: Ne
  // Set a longer timeout for upload requests
  req.setTimeout(120000); // 2 minutes

  // Handle connection close events
  req.on('close', () => {
    if (!res.headersSent) {
      logger.warn('[UPLOAD] Client closed connection before respor
    }
  });

  // Handle timeout events
  req.on('timeout', () => {
    logger.error('[UPLOAD] Request timeout');
    if (!res.headersSent) {
      res.status(408).json({
        error: 'request_timeout',
        message: 'The request timed out. Please try again with a
        success: false
      });
    }
  });

  next();
};
```

The CORS middleware was also enhanced to properly handle credentials and origins:

```javascript
  // Get the origin from the request
const origin = req.headers.origin || '*';

// Add permissive CORS headers specifically for file uploads
res.header('Access-Control-Allow-Origin', origin);
res.header('Access-Control-Allow-Methods', 'POST, OPTIONS');
res.header('Access-Control-Allow-Headers', 'Origin, X-Requested-Wi
res.header('Access-Control-Allow-Credentials', 'true');
res.header('Access-Control-Max-Age', '86400'); // 24 hours
```

Conclusion

These fixes address the core issues with script deletion and file uploads in the PSScript application. The improved error handling and transaction management ensure a more robust user experience.

Generated 2026-01-16 21:23 UTC