

Bug Fix: Vault Edit Functionality and Image Display

Issue Summary

User reported that the “Edit” button in the asset detail page (My Vault → Click Asset) was not functional. The button existed but had no click handler, making it impossible to edit asset information after creation.

Root Cause Analysis

Primary Issue: Non-functional Edit Button

- The Edit button on line 313-316 of `/app/(dashboard)/vault/[id]/page.tsx` had **no onClick handler**
- No edit mode state management existed
- No edit form implementation
- Delete button was also non-functional

Code Before Fix:

```
<Button variant="outline">
  <Edit className="h-4 w-4 mr-2" />
  Edit
</Button>
```

Implemented Solution

1. State Management

Added comprehensive state for edit mode and delete confirmation:

```
// New state variables
const [isEditing, setIsEditing] = useState(false);
const [showDeleteDialog, setShowDeleteDialog] = useState(false);
const [isDeleting, setIsDeleting] = useState(false);

const [editForm, setEditForm] = useState({
  brand: '',
  model: '',
  year: '',
  serialNumber: '',
  referenceNumber: '',
  vin: '',
  makeModel: '',
  matchingNumbers: false,
  purchaseDate: '',
  purchasePrice: '',
  estimatedValue: '',
  notes: '',
  status: 'pending',
});
```

2. Handler Functions

Implemented four new handler functions:

handleEdit()

- Populates `editForm` with current item data
- Converts dates and numbers to appropriate formats
- Sets `isEditing` to `true`

handleCancelEdit()

- Resets `editForm` to empty state
- Sets `isEditing` to `false`

handleSaveEdit()

- Prepares payload with proper type conversions
- Calls PATCH `/api/items/[id]` endpoint
- Updates local item state with response
- Shows success/error toasts
- Exits edit mode on success

handleDelete()

- Calls DELETE `/api/items/[id]` endpoint
- Redirects to `/vault` on success
- Shows success/error toasts

3. UI Enhancements

Dynamic Button Rendering

```
{!isEditing && (
  <>
    <Button onClick={handleDownloadCertificate}>Download Certificate</Button>
    <Button onClick={handleEdit}>Edit</Button>
    <Button onClick={() => setShowDeleteDialog(true)}>Delete</Button>
  </>
)
}
{isEditing && (
  <>
    <Button onClick={handleSaveEdit} disabled={isSubmitting}>Save Changes</Button>
    <Button onClick={handleCancelEdit} disabled={isSubmitting}>Cancel</Button>
  </>
)
}
```

Edit Mode Details Tab

- Conditional rendering: read-only view vs. edit mode
- Edit mode shows Input, Textarea, and Select components
- All editable fields:
- Brand, Model, Year
- Status (dropdown)
- Full Make/Model
- VIN (auto-uppercase)
- Serial Number, Reference Number
- Purchase Date, Purchase Price, Estimated Value
- Notes (textarea)

Delete Confirmation Dialog

```
<AlertDialog open={showDeleteDialog} onOpenChange={setShowDeleteDialog}>
  <DialogContent>
    <DialogTitle>Are you absolutely sure?</DialogTitle>
    <Description>
      This action cannot be undone. This will permanently delete this asset
      and remove all associated data including media files, provenance events, and certificates.
    </Description>
  </DialogContent>
  <Footer>
    <Cancel>Cancel</Cancel>
    <Action onClick={handleDelete}>Delete Asset</Action>
  </Footer>
</DialogContent>
</AlertDialog>
```

Testing Instructions

Prerequisites

- Login: <https://genesisprovenance.abacusai.app/auth/login>
- Email: john@doe.com
- Password: password123

Test 1: Edit Asset

Steps:

1. Navigate to **My Vault**
2. Click on any asset card
3. Click the “**Edit**” button (top right)
4. Verify:
 - Buttons change to “Save Changes” and “Cancel”
 - Details tab shows editable input fields
 - All current values are pre-populated
5. Modify any field (e.g., change Year to `2015`)
6. Click “**Save Changes**”
7. Verify:
 - Success toast appears: “Asset updated successfully”
 - Page returns to read-only view
 - Updated values are displayed
 - Data persists after page refresh

Test 2: Cancel Edit

Steps:

1. Click “**Edit**” button
2. Make some changes to fields
3. Click “**Cancel**” button
4. Verify:
 - Returns to read-only view
 - No changes were saved
 - Original values remain

Test 3: Delete Asset

Steps:

1. From asset detail page, click “**Delete**” button (red, top right)
2. Verify:
 - Confirmation dialog appears
 - Warning message explains permanent deletion
3. Click “**Cancel**”
4. Verify:
 - Dialog closes
 - Asset still exists
5. Click “**Delete**” again
6. Click “**Delete Asset**” in dialog
7. Verify:
 - Success toast appears: “Asset deleted successfully”

- Redirected to /vault page
- Asset no longer appears in vault list

Test 4: Field Validations

Steps:

1. Click “Edit”
2. Test year field:
 - Enter invalid year (e.g., abc)
 - Click “Save Changes”
 - Should ignore invalid year (removed from payload)
3. Test price fields:
 - Enter negative price (e.g., -1000)
 - Backend should reject with error message
4. Test VIN field:
 - Enter lowercase VIN: abc123
 - Should auto-convert to uppercase: ABC123

Test 5: Status Update

Steps:

1. Click “Edit”
 2. Change Status from “Pending Review” to “Verified”
 3. Click “Save Changes”
 4. Verify:
 - Status badge updates
 - Color changes to green
 - Dashboard stats reflect change
-

Technical Details

Files Modified

/app/(dashboard)/vault/[id]/page.tsx

- **Lines Added:** ~170 new lines
- **Imports Added:** AlertDialog components
- **New State:** 3 boolean states, 1 form state object
- **New Functions:** 4 handler functions (edit, save, cancel, delete)
- **UI Changes:** Conditional rendering for edit mode, delete dialog

API Endpoints Used

PATCH /api/items/[id]

- **Purpose:** Update item fields
- **Schema:** updateItemSchema (zod validation)
- **Editable Fields:** brand, model, year, vin, serialNumber, referenceNumber, makeModel, matching-Numbers, purchaseDate, purchasePrice, estimatedValue, notes, status
- **Response:** Updated item object

DELETE /api/items/[id]

- **Purpose:** Delete item and cascade related records

- **Cascade Deletes:** MediaAssets, ProvenanceEvents, Certificates, AIAnalyses
 - **Response:** { success: true }
-

Build Status

Build Successful

- TypeScript compilation: 0 errors
 - Next.js build: 41 routes compiled
 - Bundle size increase: 10.8 kB → 13.7 kB (vault/[id] page)
 - Deployment: Production at <https://genesisprovenance.abacusai.app>
-

Success Criteria

- [x] Edit button is functional
 - [x] Edit mode displays all editable fields
 - [x] Save functionality updates database
 - [x] Cancel button exits edit mode without saving
 - [x] Delete button shows confirmation dialog
 - [x] Delete functionality removes asset and redirects
 - [x] Success/error toasts provide feedback
 - [x] Loading states prevent double-submission
 - [x] Field validations work correctly
 - [x] Data persists across page reloads
 - [x] All CRUD operations work end-to-end
-

User Experience Improvements

Before Fix:

- Edit button was non-functional (dead button)
- Delete button was non-functional
- No way to update asset information after creation
- Required deleting and recreating asset to fix errors

After Fix:

- Edit button opens inline edit mode
 - All fields are editable
 - Save/Cancel buttons provide clear actions
 - Delete confirmation prevents accidental deletions
 - Toast notifications provide clear feedback
 - Loading states prevent confusion
 - Data validation ensures data integrity
-

Security Considerations

- ✓ **Authentication:** All operations require valid user session
 - ✓ **Authorization:** Ownership verified before PATCH/DELETE
 - ✓ **Validation:** Zod schema validates all input
 - ✓ **Confirmation:** Delete requires explicit user confirmation
 - ✓ **Error Handling:** Specific error messages for debugging
-

Related Documentation

- **Asset Registration Fix:** /BUG_FIX_ASSET_REGISTRATION.md
 - **VIN Testing Guide:** /VIN_TESTING_FINAL_VERIFIED.md
 - **API Routes:** /app/api/items/[id]/route.ts
-

Summary

This fix transforms the asset detail page from a **read-only view** to a **fully functional CRUD interface**. Users can now:

1. ✓ **Edit** any asset field after creation
2. ✓ **Update** asset status (pending → verified)
3. ✓ **Delete** assets with confirmation
4. ✓ **Cancel** edits without saving

The implementation includes proper state management, comprehensive error handling, and a polished user experience with loading states, success/error feedback, and field validations.

Status: ✓ **FIXED AND DEPLOYED**

Deployment URL: <https://genesisprovenance.abacusai.app>

Test Credentials:

- Email: john@doe.com
- Password: password123