







Print Area Configuration Backend Testing Guide

This guide provides step-by-step instructions for testing the complete backend persistence implementation.

Prerequisites

Before testing, ensure:

1.  Supabase project is created
2.  `.env` file is configured with Supabase credentials
3.  Database schema has been run in Supabase SQL Editor
4.  `product-templates` storage bucket is created and set to public
5.  At least one user account has been marked as admin
6.  Development server is running: `npm run dev`

Test Scenarios

Scenario 1: Admin Access Control ✓

Objective: Verify that only admins can access the Print Area Admin panel.

Steps:

1. **Test Non-Admin User:**
 - Sign in with a regular (non-admin) user account
 - Navigate to the Enhanced Designer page
 - Click on "Print Area Admin" (if available)
 - **Expected:** See "Access Denied" message with red alert icon
 - **Expected:** Message says "You must be an administrator to access..."
 - **Expected:** Only option is to close the dialog
2. **Test Admin User:**
 - Sign out and sign in with an admin user account
 - Navigate to the Enhanced Designer page
 - Click on "Print Area Admin"
 - **Expected:** Loading spinner shows "Checking Access"
 - **Expected:** Then shows "Loading Product" spinner
 - **Expected:** Finally opens the Print Area Configuration panel
3. **Test Unauthenticated Access:**
 - Sign out completely
 - Try to access Enhanced Designer
 - **Expected:** Redirected to sign-in page or access denied

Pass Criteria:

- ✓ Non-admins cannot access Print Area Admin

- ✓ Admins can access Print Area Admin
 - ✓ Unauthenticated users cannot access
-

Scenario 2: Load Existing Product Configuration ✓

Objective: Verify that product configurations load correctly from Supabase or fallback to products.json.

Steps:

1. Load Default Product (First Time):

- Open Print Area Admin
- Select "Tote Bag" from the product dropdown
- **Expected:** Template image loads from `/templates/bag/template.png`
- **Expected:** Print areas (Front, Back, Small Front Logo) are visible as blue rectangles
- **Expected:** Console shows: "Using configuration from products.json"

2. Save and Reload:

- Make a small change (move a print area)
- Click "Save" button
- **Expected:** Success message: "Configuration saved successfully to database!"
- **Expected:** Message appears in green with checkmark
- Close the Print Area Admin
- Reopen Print Area Admin
- **Expected:** Console shows: "Loaded configuration from Supabase"
- **Expected:** Your changes are preserved

Pass Criteria:

- ✓ Products load from products.json initially
 - ✓ Saved configurations load from Supabase
 - ✓ Template images display correctly
 - ✓ Print areas render in correct positions
-

Scenario 3: Upload Product Template Image ✓

Objective: Verify template image upload to Supabase Storage.

Steps:

1. Prepare Test Image:

- Have a PNG or JPG image ready (e.g., a photo of a tote bag)
- Recommended size: 800x800 pixels

2. Upload Template:

- Open Print Area Admin for any product
- Click the purple "Template" button
- Select your image file
- **Expected:** Button shows spinning loader with text "Template"
- **Expected:** After upload completes, success message appears
- **Expected:** Message: "Template image uploaded successfully!"
- **Expected:** Canvas refreshes with new template image

3. Verify in Supabase:

- Go to Supabase Dashboard > Storage > product-templates bucket
- **Expected:** See uploaded image file
- **Expected:** File name format: `{productKey}_{timestamp}.{extension}`

4. Verify Persistence:

- Click "Save" to save configuration with new template
- Close and reopen Print Area Admin
- **Expected:** New template image loads from Supabase Storage
- **Expected:** Image persists across sessions

Pass Criteria:

- ✓ Image uploads successfully to Supabase Storage
- ✓ Canvas updates with new template immediately
- ✓ Template URL is saved in database
- ✓ Template persists after save/reload

Scenario 4: Create New Print Area ✓

Objective: Verify creating a new print area and saving it to database.

Steps:

1. Open New Area Dialog:

- Open Print Area Admin
- In the "Print Areas" section (left sidebar), click green "+ Add" button
- **Expected:** Dialog appears: "Add New Print Area"

2. Create Print Area:

- Enter name: "Test Logo"
- Click "Add Area" button
- **Expected:** Dialog closes
- **Expected:** New blue rectangle appears on canvas at position (200, 200)
- **Expected:** "Test Logo" appears in Print Areas list
- **Expected:** Label "Test Logo" shows above the rectangle

3. Save Configuration:

- Click "Save" button
- **Expected:** Saving spinner shows
- **Expected:** Success message appears
- **Expected:** "Configuration saved successfully to database!"

4. Verify in Supabase:

- Go to Supabase Dashboard > Table Editor > print_areas
- **Expected:** See new row for "Test Logo"
- **Expected:** Check area_key, name, x, y, width, height values

5. Verify Persistence:

- Close Print Area Admin
- Reopen Print Area Admin
- **Expected:** "Test Logo" print area is still there
- **Expected:** In correct position with correct size

Pass Criteria:

- ✓ New print areas can be created
 - ✓ Print areas save to database
 - ✓ Print areas persist after page refresh
 - ✓ Data in database matches visual configuration
-

Scenario 5: Edit Print Area Properties ✓

Objective: Verify editing print area position, size, and properties saves correctly.

Steps:**1. Select and Drag Print Area:**

- Click on a print area rectangle on canvas
- **Expected:** Rectangle shows corner handles and borders
- **Expected:** Print area appears highlighted in left sidebar
- Drag the rectangle to a new position
- **Expected:** Label moves with rectangle
- **Expected:** Position values update in sidebar (X, Y)

2. Resize Print Area:

- Grab a corner handle of the selected rectangle
- Drag to resize
- **Expected:** Rectangle resizes smoothly
- **Expected:** Size values update in sidebar (Width, Height)

3. Edit Properties Manually:

- In the "Area Details" section (left sidebar)
- Change the name to "Updated Name"
- Change X to 300
- Change Y to 250
- **Expected:** Rectangle moves to new position
- **Expected:** Label updates with new name

4. Save Changes:

- Click "Save" button
- **Expected:** Success message
- **Expected:** Console log: "Configuration saved successfully"

5. Verify Persistence:

- Refresh the page
- Reopen Print Area Admin
- **Expected:** All changes are preserved
- **Expected:** Rectangle in new position with new name

Pass Criteria:

- ✓ Print areas can be moved by dragging
- ✓ Print areas can be resized with handles
- ✓ Manual property edits work correctly
- ✓ All changes persist to database
- ✓ Changes load correctly after refresh

Scenario 6: Delete Print Area ✓

Objective: Verify deleting a print area removes it from database.

Steps:

1. Delete Print Area:

- In the Print Areas list (left sidebar), find a print area
- Click the red trash icon next to it
- **Expected:** Confirmation dialog: "Are you sure you want to delete..."
- Click "OK"
- **Expected:** Print area disappears from canvas
- **Expected:** Print area removed from sidebar list
- **Expected:** Label removed from canvas

2. Save Configuration:

- Click "Save" button
- **Expected:** Success message

3. Verify in Supabase:

- Go to Supabase Dashboard > Table Editor > print_areas
- **Expected:** Deleted print area row is gone

4. Verify Persistence:

- Close and reopen Print Area Admin
- **Expected:** Deleted print area does not reappear

Pass Criteria:

- ✓ Print areas can be deleted
 - ✓ Confirmation dialog appears
 - ✓ Visual elements removed from canvas
 - ✓ Database row deleted
 - ✓ Deletion persists after reload
-

Scenario 7: Grid and Snap-to-Grid ✓

Objective: Verify grid display and snap-to-grid functionality works correctly.

Steps:

1. Toggle Grid:

- Open Print Area Admin
- In "Grid & Guides" section, uncheck "Show Grid"
- **Expected:** Grid lines disappear from canvas
- Check "Show Grid" again
- **Expected:** Grid lines reappear

2. Adjust Grid Size:

- Move the "Grid Size" slider
- **Expected:** Grid spacing changes in real-time
- Try values: 10px, 20px, 50px

3. Test Snap to Grid:

- Ensure “Snap to Grid” is checked
- Drag a print area
- **Expected:** Rectangle snaps to grid intersections
- **Expected:** Movement feels “magnetic” to grid points
- Uncheck “Snap to Grid”
- Drag again
- **Expected:** Smooth, continuous movement

Pass Criteria:

- ✓ Grid visibility toggles correctly
 - ✓ Grid size adjusts dynamically
 - ✓ Snap-to-grid constrains movement
 - ✓ Disabling snap allows free movement
-

Scenario 8: Export and Import Configuration ✓

Objective: Verify configuration can be exported to JSON and imported back.

Steps:

1. Export Configuration:

- Open Print Area Admin
- Configure some print areas
- Click blue “Export” button
- **Expected:** JSON file downloads: `{productKey}-print-areas.json`
- Open the file
- **Expected:** Valid JSON with product key, print areas, timestamp

2. Modify Print Areas:

- Delete some print areas or change positions
- Don’t save changes

3. Import Configuration:

- Click green “Import” button
- Select the previously exported JSON file
- **Expected:** Print areas revert to exported configuration
- **Expected:** Canvas updates immediately

4. Save Imported Configuration:

- Click “Save”
- **Expected:** Imported configuration saves to database

Pass Criteria:

- ✓ Export creates valid JSON file
 - ✓ JSON contains all configuration data
 - ✓ Import restores configuration
 - ✓ Canvas updates after import
 - ✓ Imported config can be saved
-

Scenario 9: Multiple Products Workflow ✓

Objective: Verify switching between products and managing multiple configurations.

Steps:

1. **Configure Product A** (e.g., Tote Bag):
 - Open Print Area Admin
 - Select "Tote Bag"
 - Add/modify print areas
 - Click "Save"
 - **Expected:** Success message
2. **Switch to Product B** (e.g., T-Shirt):
 - Close Print Area Admin
 - Reopen and select "T-Shirt"
 - **Expected:** T-Shirt template and print areas load
 - **Expected:** Different from Tote Bag configuration
3. **Configure Product B:**
 - Modify T-Shirt print areas
 - Click "Save"
 - **Expected:** Success message
4. **Switch Back to Product A:**
 - Close and reopen Print Area Admin
 - Select "Tote Bag" again
 - **Expected:** Original Tote Bag configuration intact
 - **Expected:** No cross-contamination from T-Shirt
5. **Verify in Database:**
 - Check Supabase product_templates table
 - **Expected:** Separate rows for each product
 - Check print_areas table
 - **Expected:** Correct product_template_id for each print area

Pass Criteria:

- ✓ Each product has independent configuration
 - ✓ Switching products loads correct configuration
 - ✓ Saving one product doesn't affect others
 - ✓ Database maintains separate records per product
-

Scenario 10: Error Handling ✓

Objective: Verify proper error handling for various failure scenarios.

Steps:

1. **Test Network Failure:**
 - Open Print Area Admin
 - In browser DevTools, go to Network tab
 - Set throttling to "Offline"
 - Try to save configuration

- **Expected:** Error message appears in red
- **Expected:** Message indicates network/save failure
- Restore network
- Try save again
- **Expected:** Success message

2. **Test Invalid File Upload:**

- Click "Template" upload button
- Select a non-image file (e.g., .txt or .pdf)
- **Expected:** Error message: "Please select an image file."
- Select a valid image
- **Expected:** Upload proceeds normally

3. **Test Non-Admin Save Attempt** (if mock mode):

- In mock mode, simulate non-admin
- Try to save
- **Expected:** Error message: "Only administrators can save configurations."

4. **Test Invalid Import:**

- Click "Import" button
- Select a non-JSON file or invalid JSON
- **Expected:** Error alert: "Invalid configuration file"

Pass Criteria:

- ✓ Network errors show user-friendly messages
- ✓ File type validation works
- ✓ Permission errors are clear
- ✓ Invalid data is rejected gracefully
- ✓ No crashes or unhandled exceptions

Performance Testing

Load Time Tests

1. **Initial Load:**

- Measure time from clicking Print Area Admin to panel fully loaded
- **Target:** < 2 seconds

2. **Save Operation:**

- Measure time from clicking Save to success message
- **Target:** < 1 second

3. **Template Upload:**

- Upload a 2MB image
- **Target:** < 5 seconds

Canvas Performance

1. **Drag Performance:**





- Drag print areas around canvas
- **Expected:** Smooth, 60fps movement
- No stuttering or lag

2. Multiple Print Areas:

- Create 10+ print areas
 - Drag and resize
 - **Expected:** Still performs smoothly
-

Browser Compatibility

Test in the following browsers:

-  Chrome (latest)
 -  Firefox (latest)
 -  Safari (latest)
 -  Edge (latest)
-

Database Verification Checklist

After completing all tests, verify in Supabase:

product_templates Table:

- ✓ Rows exist for configured products
- ✓ product_key is unique
- ✓ template_url points to correct storage location
- ✓ colors is valid JSON array
- ✓ base_price is set
- ✓ created_by references valid user ID
- ✓ timestamps are correct

print_areas Table:

- ✓ Rows exist for all configured print areas
- ✓ product_template_id links to correct product
- ✓ area_key is unique per product
- ✓ Position and size values are realistic (not negative, not extreme)
- ✓ max_width and max_height are set
- ✓ timestamps are correct

Storage Bucket (product-templates):

- ✓ Bucket exists and is public
 - ✓ Uploaded images are accessible via URL
 - ✓ File names follow pattern: `{productKey}_{timestamp}.{ext}`
 - ✓ Old images are replaced (not duplicated) when re-uploading
-

Common Issues and Solutions

Issue: “Access Denied” for Admin User

Solution:

1. Check Supabase Dashboard > Authentication > Users
2. Select the user
3. Verify `raw_user_meta_data` contains `{"is_admin": true}`
4. If not, run SQL:

```
sql
UPDATE auth.users
SET raw_user_meta_data = jsonb_set(
  COALESCE(raw_user_meta_data, '{}'::jsonb),
  '{is_admin}',
  'true'::jsonb
)
WHERE id = 'user-id-here';
```

Issue: Template Images Not Loading

Solution:

1. Check Supabase Storage > product-templates bucket exists
2. Verify bucket is set to public
3. Check storage policies allow public read access
4. Verify URL in database is correct

Issue: Changes Not Persisting

Solution:

1. Check browser console for errors
2. Verify Supabase credentials in `.env`
3. Check RLS policies allow authenticated users to read/write
4. Verify `is_admin()` function exists and returns true for admin users

Issue: “Configuration saved” but Data Not in Database

Solution:

1. Check browser Network tab for 401/403 errors
2. Verify user is authenticated (check auth token)
3. Check Supabase logs for RLS policy violations
4. Verify database functions exist

Sign-Off Checklist

Before pushing to production:

- ✓ All 10 test scenarios pass
- ✓ No console errors during normal operation
- ✓ Database contains correct data after testing
- ✓ Storage bucket contains uploaded images
- ✓ Admin access control works correctly

- ✓ Non-admins see appropriate denial message
 - ✓ All CRUD operations work (Create, Read, Update, Delete)
 - ✓ Configuration persists across page refreshes
 - ✓ Multiple products work independently
 - ✓ Error messages are user-friendly
 - ✓ Performance is acceptable
 - ✓ Code is committed with clear commit message
 - ✓ Documentation is up to date
 - ✓ .env.example contains all required variables
 - ✓ .env is in .gitignore
-

Next Steps After Testing

1. Document any bugs found during testing
 2. Fix critical issues before deployment
 3. Create admin user accounts for production
 4. Set up Supabase project for production
 5. Update environment variables for production
 6. Deploy and test in production environment
 7. Train administrators on how to use the system
-

Support

For issues or questions:

- Check Supabase logs: Dashboard > Logs
- Check browser console for JavaScript errors
- Review SUPABASE_SETUP.md for configuration help
- Check database schema in supabase/schema.sql