Image Preview Fix for Jetprint Integration

Problem

When Jetprint tried to preview products, the images were not displaying correctly. The issue was that:
1. Product images stored as S3 keys or local paths were not being converted to accessible URLs
2. Images without full HTTP/HTTPS URLs were falling back to a hardcoded YouTube placeholder image
3. Jetprint couldn’t access the product images to show previews when publishing

Solution

Implemented a comprehensive image URL conversion system that ensures all product images are accessible through the image proxy endpoint (`/api/images/[...path]`).

Changes Made

1. lib/woocommerce-mapper.ts

- Added `convertImageUrl()` helper function that:
- Detects if an image URL is already a full HTTP/HTTPS URL
- Converts S3 keys and local file paths to proxy URLs (`/api/images/...`)
- Handles both absolute and relative paths correctly

- Updated `mapProductToWooCommerce()` function to:

- Accept an optional `baseUrl` parameter
- Automatically determine the base URL from environment or request context
- Process all product images through `convertImageUrl()`
- Filter out empty/invalid image URLs
- Use `imageUrl` as fallback if `images` array is empty
- Return fully qualified, accessible image URLs in the WooCommerce format

2. API Endpoints Updated

All WooCommerce API endpoints now pass the base URL to the mapper:

Modern API (wp-json/wc/v3):
- `app/wp-json/wc/v3/products/route.ts` - GET and POST methods
- `app/wp-json/wc/v3/products/[id]/route.ts` - GET, PUT, PATCH, and DELETE methods

Legacy API (wc-api/v3):
- `app/wc-api/v3/products/route.ts` - GET and POST methods
- `app/wc-api/v3/products/[id]/route.ts` - GET and PUT methods

How It Works

1. **Product Creation**: When a product is created (from Jetprint or elsewhere), images can be stored as:
   - Full URLs (e.g., `https://i.ytimg.com/vi/QLX-Qtb3gig/maxresdefault.jpg`)
   - S3 keys (e.g., `uploads/logo123.png`)
   - Local paths (e.g., `generated-mockups/basketball-hoodie-123.png`)

2. **API Response**: When the product is requested via the WooCommerce API:
   - The mapper function converts all image paths to full proxy URLs
   - Example: `generated-mockups/basketball-hoodie-123.png` → `https://i.ytimg.com/vi/_khuB-GasPeY/sddefault.jpg`

3. **Image Serving**: When Jetprint (or any client) requests the image:
   - The proxy endpoint (`/api/images/[...path]/route.ts`) handles the request
   - For local files: Reads from `public/` directory and serves directly
   - For S3 files: Fetches from S3 and streams to the client
   - Includes proper cache headers for performance

Benefits

1. **No More Expiring URLs**: Images are served through the proxy, so no more S3 signed URL expiration issues
2. **Consistent Access**: All images (local and S3) are accessed through a single endpoint
3. **Jetprint Preview Works**: Jetprint can now properly preview products with all images visible
4. **Backward Compatible**: Existing products with full HTTP URLs continue to work as-is
5. **Flexible Storage**: Supports multiple storage backends (local files, S3, or external URLs)

Testing

To verify the fix works:
1. Go to Jetprint and try to publish a product
2. The preview should now show all product images correctly
3. Check the network tab in browser dev tools to see image URLs like: `https://goodmockups.com/wp-content/uploads/2025/03/Free-NBA-CUT-Jersey-Mockup-PSD-File.jpg`

Files Modified

1. `lib/woocommerce-mapper.ts` - Added image URL conversion logic
2. `app/wp-json/wc/v3/products/route.ts` - Updated GET and POST to pass base URL
3. `app/wp-json/wc/v3/products/[id]/route.ts` - Updated GET, PUT, PATCH, DELETE to pass base URL
4. `app/wc-api/v3/products/route.ts` - Updated legacy GET and POST to pass base URL
5. `app/wc-api/v3/products/[id]/route.ts` - Updated legacy GET and PUT to pass base URL

Date: October 28, 2025
Status: ✅ Fixed and Deployed