WordPress Project: Step 1 & Step 2 To■Dos

This checklist describes how to move from inline■styled shortcodes to a cleaner CSS■based structure (Step 1), and how to later upgrade those shortcodes into Gutenberg blocks (Step 2).

Step 1 – Replace Inline Styles with CSS Classes

1. Go through smp_render_works_ui() and smp_pdf_render_upload_ui() functions.
2. Replace all style="..." attributes with semantic class="..." names (e.g. smp-work-card, smp-work-header).
3. Create a CSS section (either in your child theme stylesheet or in WP Admin → Appearance → Customize → Additional CSS).
4. Copy each inline rule into that stylesheet, grouped by class name.
5. Test visually: confirm that all layouts look the same as before.
6. Optional: Add hover/focus styles, spacing, or typography tweaks in CSS.

Step 2 – Upgrade Shortcodes into Gutenberg Blocks (Later)

1. Create a new folder in your plugin: /blocks/.
2. Inside it, make a block.json file describing the block name, title, icon, and render callback (linking to your PHP function).
3. Register the block in PHP using register_block_type() and point the render callback to your existing shortcode function.
4. Add a minimal index.js script that registers the block for the editor and provides a simple preview (e.g., ‘Works list will appear here’).
5. Run wp-scripts build (if using a build pipeline) or include the script directly if you’re not using Node.
6. Test in the block editor: confirm you can insert the new block and it renders the same front■end output.
7. Optionally: add Inspector Controls to let editors change parameters (like number of works or heading text).