Power Automate Approval Workflow: Deployment Checklist

Use this checklist to ensure every critical step is completed, guaranteeing a robust, compliant, and efficient Power Automate approval workflow integrated with SharePoint.

Phase 1: SharePoint Foundation & Data Model

✅	Task	Practical Advice
☐	**Metadata Definition:** Define and create all necessary **custom columns** in the SharePoint Library (e.g., Value, Status, Document Type).	These columns will govern the routing logic and conditional branching in Power Automate.
☐	**Versioning Enabled:** Enable **Versioning** in the Document Library to track changes during the review process.	Ensure major/minor versions are set up correctly, focusing on major versions for compliance.
☐	**Content Approval Setup:** Implement SharePoint **Content Approval** settings to force documents into a "Pending" state upon submission.	This restricts editing and prevents accidental approval of draft content.
☐	**User Profile Check:** Ensure Microsoft 365/Azure AD user profiles have the **Manager field** populated for dynamic approver assignment.	If this field is blank, the `Get Manager (V2)` action will fail the flow.
☐	**Critical Tip (Status):** For **workflow automation**, always verify that the **Status column** uses a **Choice field** to prevent typos and ensure Power Automate conditions work reliably.	

Phase 2: Power Automate Core Design & Logic

✅	Task	Practical Advice
☐	**Select Correct Trigger:** Use the **"When a file is created (properties only)"** trigger for efficiency, instead of the generic "When a file is created or modified."	The "properties only" trigger is faster and consumes fewer resources, key for **automated approvals**.
☐	**Prevent Infinite Loops:** Implement a **Trigger Condition** to prevent flow re-runs (e.g., only run the flow if the 'Approval Status' column is NOT equal to 'Flow Running').	This is critical if the flow updates the same SharePoint item that started it.
☐	**Dynamic Assignment:** Use the **Get Manager (V2)** action to assign **automated approvals** dynamically based on the submitter's organizational hierarchy.	This makes your flow resilient to staff changes.
☐	**Conditional Routing:** Use **Conditional Logic (Condition/Switch)** to route documents based on metadata (e.g., Value > $5000) for multi-stage processes.	This ensures the correct hierarchical sign-off for your **document approval process**.
☐	**Custom Responses:** For complex decisions, select **Custom Responses** in the approval action to allow options like "Approve," "Reject," and "Request More Info."	This improves the quality of feedback from approvers.
☐	**Pro Tip (Grouping):** Use **Scopes** to logically group related actions (e.g., "If Approved," "If Rejected"). This drastically speeds up troubleshooting your **Power Automate approval workflow**.	

Phase 3: Escalation and Audit Trail (Compliance & UX)

✅	Task	Practical Advice
☐	**Manage SLA:** Configure the **Timeout** setting (e.g., 48 hours) within the approval action to enforce Service Level Agreements.	Ensure the timeout is configured to skip weekends or holidays if necessary.
☐	**Escalation Path:** Design a clear **Escalation Path** using a Condition following the Timeout (e.g., **Reassign Task** to a backup or Notify the approver's supervisor).	Prevents bottlenecks and keeps the **workflow automation** moving.
☐	**Professional Formatting:** Use **Markdown** formatting in the approval email's Details field for clean, professional, and easy-to-read requests (improves UX).	Makes the decision-making process faster for the approver.
☐	**Log Audit Trail:** Use the **Update file properties** action to write the final **Outcome, Approver, and Comments** back to the SharePoint item.	Creates a permanent, searchable audit trail directly alongside the document.
☐	**Notify Requestor:** Include a final **Send an email** action to notify the *original requestor* of the final decision (Approved/Rejected).	Closes the loop and ensures the user is immediately informed of the outcome.
☐	**Final Check:** Test the flow with both **Approve** and **Reject** outcomes, and ensure a **non-managerial** test user can successfully trigger and view the flow.	