Illustrative work sample

Deployment automation runbook

A colleague opens the stock catalogue after a release. The version is correct, but can they still check whether an item is available? That lookup decides whether the new release stays in service.

This local rehearsal uses a fictional, read-only catalogue. Two folders hold the releases. A pointer selects which one answers a lookup.

How to run the release

1. Establish a working baseline

The service owner approves the candidate's SHA-256 checksum. The operator records the active slot and checks that version 1.4.1 returns Available for item DEMO-42. Stop if the baseline fails.

2. Check the candidate and switch

In the inactive slot, verify the file's checksum, version 1.4.2 and the same stock lookup. Switch only if all three pass. Repeat the checks through the active slot so you see what a user would get.

3. Roll back and prove the previous release works

If any check fails after the switch, point the catalogue back to the previous slot. Verify its checksum, version 1.4.1 and the stock answer again. If any check still fails, stop and alert the service owner.

What happened in three local runs

Clean release	Version 1.4.2 returned Available through the active slot and stayed in service.
Altered release file	One extra newline changed the checksum. The workflow stopped before switching. Version 1.4.1 kept answering lookups.
Lookup failed after the switch	Version 1.4.2 reported the correct ID, but the injected fault blocked the lookup. The workflow switched back and verified that 1.4.1 returned Available.

What the on-call engineer takes away

Hand over the workflow, this runbook and the run record. In staging, have the on-call engineer repeat a clean release and a rollback. For sign-off, give the service owner the run reference, both checksums, the check results and the final active version.

Bruno Ramos de Matos