SAMPLE DOCUMENT. This shows what every script/automation engagement delivers: the documented script itself, a parameters reference, a safe dry-run mode, and click-by-click setup steps. The script excerpted here ("mailbox folder cleanup") is a representative example with fictional details.

Automation Delivery Pack

Example deliverable: Exchange Online mailbox maintenance script · PowerShell 7 + Microsoft Graph

What You Receive (every engagement)

ITEM	DESCRIPTION
The script	Fully commented — every block explains what it does and why, so any future admin can maintain it
Dry-run mode	Default-on preview: see exactly what *would* happen before anything actually moves or changes
Run log	Every item touched is logged to a timestamped CSV — your audit trail
Setup guide	Click-by-click app registration / permissions walkthrough with screenshots (page 2 shows the format)
Usage reference	Each parameter documented with copy-paste examples for your exact scenarios
Handover note	Plain-English summary: what was built, where it lives, what to check monthly

Code Style Sample (excerpt)

```
<#
.SYNOPSIS
  Moves mailbox items matching an age condition into a target archive
  subfolder via Microsoft Graph (certificate-authenticated, no stored
  passwords).
.NOTES
  - Native Exchange Online cmdlets cannot move individual items into
    an archive subfolder; this is why Graph is used here.
  - Default mode is -DryRun: nothing moves until you remove the switch.
  - Graph throttling (HTTP 429) is handled with exponential backoff so
    large mailboxes complete safely.
#>
param(
  # Mailbox to process, the user's primary mailbox address
  [Parameter(Mandatory)] [string]$Mailbox,

  # Move items older than this many days (e.g. 1095 = 3 years).
  # Omit together with -OlderThanDate to move ALL items.
  [int]$OlderThanDays,

  # ...or an explicit cutoff date, e.g. "2023-01-01"
```

```
    [datetime]$OlderThanDate,

    # Archive subfolder to move items into (created if missing)
    [string]$TargetFolder = "Archive/Aged Mail",

    # Preview only — log what WOULD move, move nothing (default ON)
    [switch]$DryRun = $true
)
```

Setup Guide (format sample — real guide includes screenshots)

1. **Create the app registration** — Entra admin center → App registrations → New. Name it for the task so it's identifiable in audits later.
2. **Grant least-privilege permissions** — only `Mail.ReadWrite` (Application), admin-consented. Nothing broader: the app can touch mail and nothing else.
3. **Certificate authentication** — a self-signed certificate is generated on your admin machine; the public key uploads to the app. No passwords or client secrets stored anywhere.
4. **Restrict scope (recommended)** — an application access policy limits the app to only the mailboxes you name, instead of tenant-wide access.
5. **First run is always a dry run** — you review the CSV of what would move, and only then run live.

Usage Reference (format sample)

SCENARIO	COMMAND
Preview: items older than 3 years	`.\Move-AgedMail.ps1 -Mailbox <mailbox> -OlderThanDays 1095`
Live run, same condition	`.\Move-AgedMail.ps1 -Mailbox <mailbox> -OlderThanDays 1095 -DryRun:$false`
Everything before a fixed date	`.\Move-AgedMail.ps1 -Mailbox <mailbox> -OlderThanDate "2023-01-01" -DryRun:$false`
All items, custom folder	`.\Move-AgedMail.ps1 -Mailbox <mailbox> -TargetFolder "Archive/Pre-Migration" -DryRun:$false`

Safety Principles (every script, every time)

- **Dry-run by default** — destructive or bulk operations never run on the first pass.
- **Least privilege** — the narrowest permission that can do the job, scoped to named mailboxes where supported.
- **No stored secrets** — certificate auth, never passwords in files or scheduled tasks.
- **Logs as deliverable** — if it isn't logged, it didn't happen; every run produces an audit CSV.
- **Throttle-aware** — API rate limits are handled in code, not discovered in production.

The standard: everything delivered is something your next IT person can read, understand, and maintain without me. Documentation isn't an add-on — it's the deliverable.