Business Automation Audit Checklist

A seven-section audit for finding, prioritising and safely rolling out business automation. Score each line, total each section, then use the summary page to decide where to start.

How to score: 0 = not in place, 1 = partly in place, 2 = fully in place. Each section has 7 items, so the maximum is 14 per section and 98 overall.

Section score	What it means
11-14	Strong. Ready to automate in this area.
6-10	Gaps to close before or during your first build.
0-5	Fix the basics here first; automating now is likely to create rework.

This checklist is a free planning tool from SkillStack Suite. It is general guidance, not legal, security or compliance advice.

1. Where the manual hours actually go

Before choosing anything to automate, measure where repetitive effort is really spent. Estimates from memory are usually wrong by a wide margin, so log the work for one or two normal weeks.

#	Check	Score (0-2)
1	Each team has listed its recurring manual tasks (copying, re-keying, chasing, reconciling, formatting, sending).	
2	Each task has a measured or logged time per occurrence, not a guess.	
3	Each task has a frequency (per day, week or month) and a named owner.	
4	Weekly hours per task have been calculated: minutes x occurrences / 60.	
5	Time spent fixing errors from manual work is logged separately.	
6	Tasks done outside normal hours or at month-end peaks are captured.	
7	The top ten tasks by weekly hours are ranked on one page.	
	Section total (max 14)	

Tip: a shared sheet with columns Task, Owner, Minutes, Times per week, Hours per week is enough. Sort by the last column.

Notes and actions

Action	Owner	By when

2. Processes worth automating first

The best first candidates are high-volume, rules-based and stable. Leave judgement-heavy or constantly changing work until later.

#	Check	Score (0-2)
1	Candidate processes follow written or clearly repeatable rules.	
2	Inputs arrive in a consistent, structured format (forms, exports, system records).	
3	The process runs often enough that savings are worth building for (rough guide: 2+ hours per week).	
4	The process has not changed significantly in the last six months.	
5	A failure in the automated version would be detectable and recoverable.	
6	Each candidate has an expected hours-saved estimate and a build-effort estimate.	
7	One or two quick wins have been chosen to build first, not ten at once.	
	Section total (max 14)	

Tip: score each candidate 1-5 on volume, rule clarity, input quality and stability. Start with the highest total.

Notes and actions

Action	Owner	By when

3. Data and systems inventory

Automation connects systems. You need to know what each system holds, who owns it and how data can get in and out.

#	Check	Score (0-2)
1	Every system the chosen processes touch is listed (CRM, accounting, email, forms, spreadsheets, files).	
2	Each system has a named business owner and an admin.	
3	For each system, the integration route is known: native connector, API, webhook, export or none.	
4	The system of record for each key field (customer, order, invoice, status) is agreed.	
5	Field names and formats that differ between systems are documented (dates, IDs, currency).	
6	Access will use a dedicated service account or approved connection, not a personal login.	
7	Data protection obligations for personal data in each flow have been checked.	
	Section total (max 14)	

Tip: if two systems disagree about a customer's status, decide which one wins before you automate the sync.

Notes and actions

Action	Owner	By when

4. Exception paths you must design for

Most automation failures come from cases nobody designed for. List them before you build, not after the first incident.

#	Check	Score (0-2)
1	Missing or blank required fields have a defined handling rule.	
2	Duplicate records (same customer, same order) have a defined handling rule.	
3	Values outside expected ranges (negative amounts, future dates) are caught.	
4	Records the automation cannot classify go to a named human queue, not nowhere.	
5	Each exception queue has an owner and a target response time.	
6	Approval steps that must stay human are explicitly marked.	
7	At least ten real past examples, including awkward ones, have been walked through the design.	
	Section total (max 14)	

Tip: ask the person who does the job today, "What are the weird ones?" Their answers are your exception list.

Notes and actions

Action	Owner	By when

5. Reliability and failure handling

An automation that fails silently is worse than a manual process. Plan how you will know it broke and how you will recover.

#	Check	Score (0-2)
1	Every automation sends an alert to a named person or channel when a run fails.	
2	Retries are configured for temporary errors (timeouts, rate limits) with a sensible limit.	
3	Re-running a step will not create duplicates (each record has a unique ID that is checked first).	
4	A run log records what was processed, when and with what result.	
5	A daily or weekly check confirms expected volumes actually ran.	
6	Credentials, API keys and connections have an owner and a renewal reminder.	
7	A written fallback explains how to do the work manually if the automation is down.	
	Section total (max 14)	

Tip: test failure on purpose. Disconnect a system in a test run and confirm the alert arrives.

Notes and actions

Action	Owner	By when

6. Rollout and training

People need to know what changed, what they still own and what to do when something looks wrong.

#	Check	Score (0-2)
1	The automation has been tested with real (or realistic) data before going live.	
2	A parallel run compared automated output with the manual process for at least a few cycles.	
3	The people affected know what the automation now does and what they still do.	
4	A one-page guide explains how to spot problems and who to contact.	
5	The go-live date avoids month-end, peak season or other high-risk periods.	
6	Someone owns the automation after launch, with a backup owner.	
7	Changes to the automation follow a simple request and review step.	
	Section total (max 14)	

Tip: retire the old manual steps deliberately. If nobody is told to stop, both versions run and errors double.

Notes and actions

Action	Owner	By when

7. Measuring hours saved

Measure results the same way you measured the baseline in section 1, so the comparison is honest.

#	Check	Score (0-2)
1	Baseline weekly hours for each automated task were recorded before launch.	
2	Post-launch hours are measured after a settling-in period (four to eight weeks).	
3	Time spent handling exceptions and maintaining the automation is counted against the savings.	
4	Error rates before and after are compared, not only time.	
5	Freed-up time is assigned to named work, so savings become real capacity.	
6	Results are reviewed quarterly and automations that no longer pay off are revised.	
7	A short summary of hours saved and lessons learned is shared with the team.	
	Section total (max 14)	

Tip: net hours saved = baseline hours - (remaining manual hours + exception handling + maintenance).

Notes and actions

Action	Owner	By when

Summary and next steps

Section	Score / 14	Priority (H/M/L)
1. Where the manual hours actually go		
2. Processes worth automating first		
3. Data and systems inventory		
4. Exception paths you must design for		
5. Reliability and failure handling		
6. Rollout and training		
7. Measuring hours saved		
Total / 98		

Reading your result

- Start with the lowest-scoring section among sections 1-4. These determine whether you are automating the right thing.
- Do not launch anything until section 5 scores at least 8. Silent failures cost more than manual work.
- Use section 7 to report results in hours, not in number of automations built.

Want to go further? The Workflow Automation for Business Teams course walks through process mapping, exception design, failure handling and measurement in 18 practical lessons, with a Certificate of Completion. Details at skillstacksuite.com.