Mistake-Proofing Design Checklist

A practical Poka-Yoke review sheet for preventing errors before they become defects

How to use this checklist: Use it during design review, process validation, line setup, CAPA, Kaizen, or risk-reduction projects. For each item, mark Yes / No / N/A and capture evidence or action needed. A strong Poka-Yoke should be simple, reliable, easy to use, and preferably preventive rather than only detective.

Process / Product		**Review Date**	
Team / Owner		**Location / Line**	
Mistake / Error Risk		**Customer / Safety Impact**	
Reviewer		**Revision**	

Quick Design Rule

Best target: make the wrong action impossible. If that is not practical, make the error immediately visible and stop it from moving forward.

1. Understand the Error Risk

#	Checklist question	Yes	No	N/A	Evidence / action needed
1	Is the specific mistake clearly defined in simple language?	☐	☐	☐	
2	Is the point where the mistake occurs known and visible in the process?	☐	☐	☐	
3	Is the potential defect, rework, delay, safety risk, or customer impact identified?	☐	☐	☐	
4	Has the team checked actual process data, complaints, scrap, audit findings, or operator feedback?	☐	☐	☐	
5	Is the root cause understood instead of assuming “operator error”?	☐	☐	☐	

2. Design for Prevention First

#	Checklist question	Yes	No	N/A	Evidence / action needed
1	Does the design prevent the wrong part, orientation, quantity, or sequence from being used?	☐	☐	☐	
2	Can the process be changed so the correct action is the easiest action?	☐	☐	☐	
3	Can a physical guide, fixture, locator, shape match, color cue, or interlock remove the error opportunity?	☐	☐	☐	
4	Does the solution reduce reliance on memory, attention, training, or manual inspection?	☐	☐	☐	
5	Has the team considered prevention before adding another inspection step?	☐	☐	☐	

3. Detection and Immediate Feedback

#	Checklist question	Yes	No	N/A	Evidence / action needed
1	If prevention is not possible, does the mistake become visible immediately?	☐	☐	☐	
2	Does the system alert the user at the point of error through light, sound, message, or status indication?	☐	☐	☐	
3	Can the process stop, lock, reject, or hold the item before the defect moves forward?	☐	☐	☐	
4	Is the feedback clear enough to tell the user what went wrong and what to correct?	☐	☐	☐	
5	Is the detection method fast enough to avoid batching defects?	☐	☐	☐	

4. Usability and Human Factors

#	Checklist question	Yes	No	N/A	Evidence / action needed
1	Can a new or trained user understand the Poka-Yoke without lengthy explanation?	☐	☐	☐	
2	Is the design easy to see, reach, clean, operate, and reset?	☐	☐	☐	
3	Does the design avoid creating extra awkward movement, confusion, or workaround behavior?	☐	☐	☐	
4	Are labels, colors, icons, or indicators simple and consistent with the work environment?	☐	☐	☐	
5	Has operator or user feedback been included before final release?	☐	☐	☐	

5. Reliability and Maintainability

#	Checklist question	Yes	No	N/A	Evidence / action needed
1	Will the mistake-proofing device continue to work under normal production conditions?	☐	☐	☐	
2	Can the device tolerate dust, oil, vibration, wear, cleaning, temperature, or handling variation?	☐	☐	☐	
3	Is there a simple way to verify that the Poka-Yoke is still functioning correctly?	☐	☐	☐	
4	Are maintenance, calibration, replacement, or inspection needs defined?	☐	☐	☐	
5	Does the solution avoid frequent nuisance alarms or false rejects?	☐	☐	☐	

6. Verification, Control, and Launch

#	Checklist question	Yes	No	N/A	Evidence / action needed
1	Has the solution been tested with correct, incorrect, missing, reversed, and borderline conditions?	☐	☐	☐	
2	Are acceptance criteria defined for pass/fail performance?	☐	☐	☐	
3	Is the control documented in the work instruction, control plan, PFMEA, checklist, or training material?	☐	☐	☐	
4	Is ownership assigned for daily checks and long-term control?	☐	☐	☐	
5	Is there a reaction plan if the Poka-Yoke fails or is bypassed?	☐	☐	☐	

Final Decision

Overall status	☐ Approved ☐ Approved with actions ☐ Not approved	**Target date**	
Open actions		**Owner**	
Verification evidence		**Reviewer sign-off**	
Next review date		**Comments**	

Practical Notes

- A Poka-Yoke is strongest when it prevents the mistake at the source, not when it only finds the defect later.
- Keep the solution simple enough that people use it naturally during normal work.
- Review the design after launch. Real-world use often reveals improvement opportunities that were not visible during design review.