

# Mainframe-Aware AI Systems Checklist

## Core positioning

Marry mainframe and AI.

The serious opportunity is not pretending legacy systems do not exist. It is modernizing around them properly.

This checklist is for teams carrying IBM Z-era or other long-lived system-of-record realities while trying to add AI, automation, or new execution layers.

## 1. System-of-record boundaries

- ? What system still owns the truth for this workflow?
- ? What must never be bypassed by an AI layer?
- ? Where would a mistaken automation decision create operational or regulatory risk?

## 2. Data contracts

- ? What data is clean enough for automation and what still needs human judgment?
- ? What fields, files, or interfaces are stable enough to integrate safely?
- ? What breaks if one upstream legacy dependency changes shape?

## 3. Failure handling

- ? What is the safe failure mode if the AI layer misfires?
- ? Can operators inspect what happened quickly?
- ? Is there a rollback or manual override path?

## 4. Audit and trust

- ? Can the business explain why the system made a decision?
- ? Are logs and handoffs legible enough for operators and leaders?
- ? Will the modernization layer increase trust or quietly reduce it?

## 5. Execution sequencing

- ? What should be diagnosed first before anyone buys a larger build?
- ? Which part of the workflow is the highest-friction bottleneck right now?
- ? What can be modernized safely this quarter versus what needs deeper review?

## Fast rule

If the AI layer cannot explain how it touches a system of record, survives a failure, and hands control back to an operator safely, it is not ready for production.

## Next step

After reading this, next step is a 48h review.

If one handoff, interface, or workflow already feels risky, do not guess.

Use the Mainframe-Aware AI Architecture Review:

? View the review page ? <https://slickofficials.com/mini-offers/mainframe-aware-ai-architecture-review>

? Buy Standard Review ? <https://paystack.shop/pay/backend-architecture-review-standard>

? Buy Premium Review ? <https://paystack.shop/pay/backend-architecture-review-premium>