



Software health check

A twenty-question self-assessment for software already in production

Answer for one specific system, not for your software in general. Be strict: partly is usually a no that has been lucky so far. Nobody sees this but you, and at the end you count your no answers.

Updates and dependencies

	Yes	Partly	No
Dependencies and frameworks were updated in the past six months	☐	☐	☐
The server or platform it runs on still receives security patches	☐	☐	☐
You know which parts of the stack are no longer maintained	☐	☐	☐
Someone follows the security announcements for this stack	☐	☐	☐

Backups

	Yes	Partly	No
Backups run automatically, without anyone having to remember them	☐	☐	☐
A restore was tested in the past year, on a real copy	☐	☐	☐
Backups are stored somewhere other than the system itself	☐	☐	☐
You know how much data you would lose at most (an hour, a day)	☐	☐	☐

Knowledge and documentation

	Yes	Partly	No
Written documentation exists for setup and deployment	☐	☐	☐
More than one person can put a change into production	☐	☐	☐
The known workarounds (never click that twice) are written down	☐	☐	☐
If the main developer left tomorrow, work could continue within a week	☐	☐	☐

Security basics

	Yes	Partly	No
Every user has a personal account, no shared logins	☐	☐	☐

	Yes	Partly	No
Access for people who left has been removed, and you checked recently	☐	☐	☐
Admin accounts require two-factor authentication	☐	☐	☐
Passwords and keys live outside the source code	☐	☐	☐

Daily operations

	Yes	Partly	No
You hear about failures from an alert, not from a user	☐	☐	☐
The system has an owner, by name	☐	☐	☐
A test environment exists, separate from production	☐	☐	☐
You know what the system costs per month, everything included	☐	☐	☐

Count your no answers

Count every No as one point and every Partly as half a point.

0 to 2: healthy. Close the gaps you found and keep the routine that got you here.

3 to 6: normal for software that grew over the years, but each gap is now a known choice. Pick the three cheapest to close this quarter; backups and access usually top that list.

7 or more: the system runs on luck and on one or two people. Do not start with a rewrite. Start with backups, access and one page of documentation: that is a week of work and it removes the worst risks.

Repeat this check once a year, or after every big change. The score matters less than the direction it moves in.