FREE CHECKLIST · FOR BUSINESS OWNERS WITH A WEBSITE, APP OR SOFTWARE

Your project is live. Now what?

Launch isn't the end of a project. It's the start of owning one. Seven things to take care of after launch: what a good team does, what you own on your side, what it costs, and the question to ask before you sign.



p.2	The one-page after-launch checklist: print this one	ALL 7
p.3	Get the keys · The first few weeks · Support	STEPS 1–3
p.4	Keep it running · Keep it current	STEPS 4–5
p.5	Keep improving it · Own it on your side	STEPS 6–7
p.6	Your yearly budget worksheet	WORKSHEET

How to use this: before you sign, ask the agency every question on page 2 and tick what you actually heard. After launch, use page 6 (or the Excel version) to set a yearly budget. It works for any project: a website, a web app, a mobile app or software your team uses every day.

The one-page after-launch checklist

Tick only what's in place or in writing, not what you were promised.

1 Get the keys — HANDOVER

- [ ] Code repository, domain, hosting, database and app store accounts are **in your company's name**
- [ ] Every admin login has **you as owner**, the agency as a guest
- [ ] A **written handover list**: every account, who owns it, where passwords live, how it works, how to release

2 The first few weeks — WARRANTY

- [ ] Warranty length and what it covers are in writing. It ends on: _____
- [ ] One channel for reporting problems, with a **severity scale** (critical / high / normal / low)

3 Agree how support works — SUPPORT PLAN

- [ ] Model agreed: **monthly support plan** (hours: _____) or **pay per fix**
- [ ] **Response time** and **fix time** both in writing (a service level agreement)
- [ ] You know who answers if it breaks at 2 a.m. on a Saturday

4 Keep it running — MONITORING

- [ ] **Monitoring alerts** for downtime and error spikes go to the team
- [ ] Backups exist **and a restore has been tested**
- [ ] A **renewals calendar**: domain, certificates, licences, app store accounts

5 Keep it current — MAINTENANCE

- [ ] A **scheduled maintenance** slot (monthly or quarterly): update, test, release
- [ ] Platform and security updates are planned, not left until they break something

6 Keep improving it — ROADMAP

- [ ] Analytics and a way to collect feedback are set up
- [ ] **One priority list**, small releases every few weeks, new ideas through a change process

7 Own it on your side — OWNERSHIP

- [ ] **One named person** owns the product: _____
- [ ] A yearly budget is set (page 6) and a **quarterly review** is in the calendar

STEPS 1–3

Keys. First weeks. Support.

Settle these before the final invoice is paid. That's when you have the most leverage.

1 Get the keys

RED FLAG

“We'll keep the access for now, it's easier.” It is easier. For them.

GOOD PRACTICE

A written handover list. The test: if a new team started on Monday, could they pick it up?

ASK BEFORE YOU SIGN

"Can I see the handover list before the final payment?"

2 The first few weeks

Launch is when real users find what testing missed. Good teams plan a **hypercare or warranty period**, usually a few weeks to a few months, where they watch closely and fix launch bugs at no extra cost. Your job: report problems well, through one channel, with a screenshot, what you did, what you expected, and how serious it is.

ASK BEFORE YOU SIGN

"How long is the warranty, what does it cover, and when exactly does it end?" Then put the date in your calendar.

3 Agree how support works

MONTHLY SUPPORT PLAN

A retainer with a set number of hours. Predictable, and someone is already familiar with your system.

PAY PER FIX

Looks cheaper until something urgent happens and you're one of many clients in the queue.

Either way, get two numbers in writing: **response time** (how fast someone looks) and **fix time** (how fast it's solved). The gap between them is where the surprises are.

ASK BEFORE YOU SIGN

"If checkout breaks at 2 a.m. on a Saturday, who answers, and how fast?"

STEPS 4–5

Keep it running. Keep it current.

Software doesn't run on its own, and the world around it keeps changing even if you never change a feature.

4 Keep it running

- [ ] **Monitoring:** alerts when the site is down or errors spike, before your customers tell you.
- [ ] **Backups:** not just "we have backups", but a restore that has actually been tested.
- [ ] **Renewals calendar:** domain, security certificates, software licences and plugins, app store accounts (Apple Developer Program $99 a year; Google Play $25 one-time).
- [ ] **Usage costs:** hosting, email, maps and AI services grow with every new user.

ASK BEFORE YOU SIGN

"What will this cost to run each month at today's usage, and at ten times today's usage?"

5 Keep it current

Browsers and phones update. The libraries your code is built on publish security fixes. Platforms change their rules: in 2026 Apple requires app updates to be built with its latest tools, and Google requires apps to target a recent Android version or they stop showing up for new users on newer phones. For websites, an outdated plugin or server is one of the most common ways sites get hacked.

RED FLAG

"If it isn't broken, don't touch it." With software, not touching it is how it breaks.

GOOD PRACTICE

Scheduled maintenance: a regular slot, monthly or quarterly, to apply updates, test them and release. Planned work, not emergencies.

Platform rules change every year. Check the current Apple and Google requirements with your team rather than relying on this page.

STEPS 6–7

Keep improving it. Own it.

You'll learn more in the first month of real use than in the whole build. Someone on your side needs to own what happens next.

6 Keep improving it

- [ ] **Analytics**, so you see what people actually do.
- [ ] **A simple way to collect feedback.**
- [ ] **One list of ideas and fixes**, in priority order, that everyone works from.
- [ ] **Small releases every few weeks**, each tested and shown to you before it goes live. New ideas go through the change process: you ask, they estimate, you approve, then they build.

ASK BEFORE YOU SIGN

"How will we decide what to build next, and how often will we release?"

7 Own it on your side

The agency can do everything right and the project can still drift if nobody in your business owns it. Name **one person** (not a committee) who makes decisions, approves changes and knows the budget. Set a **yearly budget** (page 6). Hold a **quarterly review**: what broke, what it cost, what's coming next, and what needs updating before it becomes urgent.

ASK YOURSELF

"If our main contact at the agency left tomorrow, would we be okay?"

WORKSHEET

Your yearly after-launch budget

A common planning rule is **15–20% of the build cost every year** for maintenance and support, plus running costs that grow with usage. It's a starting point, not a law, but it's a lot better than zero. Prefer it to add up for you? Use the Excel version.

ITEM	HOW OFTEN	YOUR ESTIMATE
Build cost (from your quote)	Once	$_____
A · Maintenance & support (15–20% of build)	**Per year**	$_____
Domain name	Per year	$_____
Security certificate (often included with hosting)	Per year	$_____
Software licences & plugins	Per year	$_____
Apple Developer Program (if you have an iPhone app)	$99 / year	$_____
Google Play developer account (if Android)	$25 one-time	$_____
B · Renewals and accounts, total	**Per year**	$_____
Hosting / servers	Monthly × 12	$_____
Database	Monthly × 12	$_____
Email / SMS services	Monthly × 12	$_____
Maps, AI or other paid services	Monthly × 12	$_____
Monitoring & backups	Monthly × 12	$_____
C · Running costs, total (and at 10× usage: $_____)	**Per year**	$_____
Yearly after-launch budget (A + B + C)		**$_____**

LAUNCHING SOON, OR ALREADY LIVE AND A BIT FRAGILE?

I've spent years in senior project management on the agency side. Book a free session: we'll go through what you have and tell you plainly what's missing.

calendar.app.google/Yy6SMMumbrQHDviP6

Book a free session →