

24 · Teams and operations

Digital Project Launch Workbook

A fill-in workbook before you build: problem, user, alternatives, research, value, scope, validation and a first-release checklist, with a worked example.

For: Founders, small businesses and associations planning a digital product who want to test the idea before spending time and money on building it.

Level: Intermediate Version 1.0 · 2026-09-21

Why this workbook?

Today's building tools, AI tools among them, make building a website or app faster than ever. But building fast does not answer the most important question: does anyone need this product enough to use it or pay for it? This workbook helps you write down what you know and what you assume about the idea, design small tests before building, and define a small first release you can launch and learn from.

The workbook does not guarantee the idea will succeed, and it does not replace talking to real users. Its value is that it makes your assumptions written and testable, so you discover mistakes while they are still cheap to fix.

How to use it

1. **Fill in the parts in order:** each part builds on the previous one; do not write the scope before you know the user and their problem.
2. **Separate what you know from what you assume:** mark "assumption" next to every sentence without evidence yet.
3. **Turn assumptions into research questions and tests:** in parts four and seven.
4. **Revisit the workbook after each validation round:** update what changed and keep earlier versions to see how the idea evolved.
5. **Do not start building before the first-release checklist is complete:** or at least until you have consciously decided which items to postpone and why.

Part one: the problem

Write the problem from the point of view of the person living it, not from the point of view of the solution you have in mind. "There is no app for X" is not a problem; the problem is what happens to people because a solution is missing.

What is the problem? Describe it in two sentences without mentioning your solution.

Who lives with this problem, and how often does it happen to them?

What does the problem cost them today (time, money, embarrassment, missed opportunities)?

**What evidence do you have that the problem exists (conversations, complaints, personal experience)?
Write "assumption" if there is none.**

Part two: the target user

Choose a specific first user instead of "everyone". You can expand later, but the first release needs a clear person to design for and talk to.

Who exactly is the first user? (role, location, situation)

What situation or moment makes them need the solution?

Which devices and channels do they actually use? (phone, WhatsApp, computer, specific apps)

Is the user the one who pays? If not, who pays and why?

Part three: current alternatives

The user solves the problem somehow today, even badly: paper, a WhatsApp group, calling acquaintances, or ignoring it. These alternatives are your real competition.

How does the user solve the problem today? List at least three alternatives.

What do they like about these alternatives, and what annoys them?

What would make them leave their current alternative? What does switching cost them?

Part four: initial research questions

Before building, collect answers from real users. Ask about their past experiences and actual behaviour, not their opinion of your idea; people tend to be polite when asked "Would you use this?".

- Tell me about the last time you faced [the problem]. What happened?
- How did you solve it? How long did it take? Did you pay anything?
- What is the hardest part of that process?
- Have you looked for a better solution? What did you find, and why did you not use it, or why did you stop?
- Who else is involved in this decision?

What are the five most important questions you want answered before building?

Who will you talk to? How many people, how will you reach them, and when?

Part five: the value proposition

A value proposition is a sentence connecting the user and their problem to what your product delivers, and to what sets it apart from alternatives. Write it after the research, not before, and revise it whenever you learn something new.

Value proposition formula

For [first user]
who struggles with [the problem, in their words],
[product name] helps them [the outcome they want]
by [the core method],
unlike [current alternative], which [its drawback].

Write your value proposition using the formula above.

What evidence shows the user cares about this particular outcome?

Part six: scope

The first release solves the core problem for one user in the simplest possible way. Every extra feature delays launch and delays learning. Write down what is in and, explicitly, what is out, because whatever is not written comes back to the discussion every week.

What must the user be able to do in the first release? (three to five capabilities)

What will not be in the first release, even if it matters later?

What can be done manually at first instead of built? (for example, confirming orders by phone)

What are the constraints: budget, time, team, sensitive data, legal requirements?

Part seven: the validation plan

For each important assumption, design a small test with a success criterion you set **before** the test. The criterion is your own decision based on your project, not a general benchmark. Examples of tests: interviews, a simple page collecting sign-ups of interest, a prototype tried with five users, or delivering the service manually to a small number of people before building any system.

Test card

Assumption: [what we believe is true]

Test: [what we will do, with whom, how many]

Success criterion (set before starting): [the number or behaviour that means it holds]

Duration: [] Owner: []

Result: []

Decision: [continue / adjust / stop]

The riskiest assumption (if wrong, the idea falls): what is it, and how will you test it?

Second assumption and its test:

Third assumption and its test:

Part eight: first-release checklist

- ☐ The first user and their problem are written down and backed by real conversations.
- ☐ The riskiest assumption has been tested and the result recorded.
- ☐ The core capabilities work end to end on a phone.
- ☐ What is out of the first release is written down and agreed.
- ☐ Users have a way to get help when something goes wrong, and someone is responsible for replying.
- ☐ Personal data collected is the minimum needed, and a privacy policy is published.
- ☐ Passwords and keys are stored outside the code, and backups work.
- ☐ Core usage is measured: how many people started, and how many completed the key step.
- ☐ There is a list of ten to twenty early users you will contact directly at launch.
- ☐ The date of the first post-launch review is set, along with the questions it will answer.

Completed example

Illustrative example

The idea: "Sanaa", an invented platform in Tripoli connecting households with trusted tradespeople for home repairs (plumbing, electrical, painting). All names and numbers are invented for illustration.

1. Problem: when something breaks at home, the family looks for a tradesperson through acquaintances and WhatsApp groups, waits for replies that do not come, and does not know the price until the visit. Cost: days of waiting, and sometimes a higher price than expected or work that has to

be redone. Evidence: conversations with eight families the team knows, who described the same situation. How widespread the problem is: "assumption".

2. First user: households in specific Tripoli neighbourhoods, specifically the person who handles repairs at home and uses a phone and WhatsApp daily. Moment: an urgent or moderately urgent fault. Payer: the household. Open question: will tradespeople also pay for jobs? An "assumption" to test.

3. Alternatives: a relative's recommendation (trusted but slow), neighbourhood WhatsApp groups (fast but hit-and-miss), the usual tradesperson (excellent when available). What annoys users: not knowing the price in advance, and missed appointments. Switching cost: low if the first experience is good.

4. Research questions: When did you last need a tradesperson, and how long did it take to get one? How did you find out the price? What happened if the work was poor? Would you pay a little extra for a guaranteed appointment? Plan: ten interviews with households and five with tradespeople over two weeks.

5. Value proposition: "For Tripoli families who wait days to find a trusted tradesperson, Sanaa helps them book a known tradesperson at a set time with an estimated price before the visit, unlike WhatsApp groups, which guarantee neither a reply nor a price." Evidence: six of ten households interviewed named surprise pricing as their biggest frustration.

6. Scope: in the first release: requesting a service in three categories only, showing two or three tradespeople per category with ratings from previous jobs on the platform, and appointment confirmation. Out: online payment, a tradesperson app, a complex rating system. Done manually: the platform team confirms appointments by phone, and collects ratings by message after the visit. Constraints: a two-person team, a limited budget, and customer phone numbers stored securely and shared only with the chosen tradesperson.

7. Validation plan:

Assumption	Test	Success criterion (set by the team in advance)	Result
Families will book through an intermediary instead of their contacts	Run the service manually through a WhatsApp number for three weeks in two neighbourhoods	At least 15 real requests	22 requests
Tradespeople will keep appointments	Track punctuality on the manual jobs	Most visits on time	17 of 22 on time
Tradespeople will pay a commission	Offer a small commission to the five tradespeople after week two	Three of five agree	Two agreed; decision: adjust the model and test a monthly subscription instead

8. First-release checklist: complete except usage measurement, which the team decided to add a week before launch. First review two weeks after launch, with one core question: how many families booked a second time?

Common mistakes

- **Mistake:** starting with the solution and a feature list. **Fix:** write the problem and the user first, and let scope come last.
- **Mistake:** asking people "Would you use this app?". **Fix:** ask about their past behaviour and what they actually paid.
- **Mistake:** setting the success criterion after seeing the result. **Fix:** write it on the test card before starting.
- **Mistake:** automating everything from day one. **Fix:** do manually whatever can be done manually until people prove they want it.
- **Mistake:** ignoring alternatives because "nobody offers what we offer". **Fix:** the alternative may be paper or a WhatsApp group, and that is what you must beat.
- **Mistake:** collecting more personal data than needed. **Fix:** collect the minimum, and explain to users why you need it and how you protect it.

Workbook completion checklist

- ☐ The problem is written from the user's point of view, without mentioning the solution.
- ☐ The first user is specific, and it is clear who pays.
- ☐ At least three current alternatives, with their strengths and weaknesses.
- ☐ Research questions are written, and you know whom you will talk to and when.
- ☐ Every assumption is marked "assumption".
- ☐ The value proposition follows the formula and is backed by research evidence.
- ☐ What is in and out of the first release is written down.
- ☐ Each important assumption has a test card with a success criterion set in advance.
- ☐ The first-release checklist is reviewed, and postponed items are deliberate and justified.

About the author — Mohamad Charafeddine is an entrepreneur from Tripoli, Lebanon: founder of the Passion digital marketing agency, president of Takaful Youth Association, and founder of digital platforms. He trains individuals and teams in applied AI and marketing.

Examples in this file are fictional. Free to use in your work and organisation; credit the source if you republish.
mohamadcharafeddine.com/en/resources/digital-project-launch/