YOUR READING MAP

Small software, useful work

Build a service around a narrow, repeated customer task.

Your mission

Find a recurring problem, test a simple solution and build the smallest dependable version. Invite a few pilot customers, watch them use it and learn whether the value is worth paying for.

03–04 — **Find a problem worth serving**
A fictional maker tool and customer interview questions.

05–06 — **Define and test the first version**
A focused product brief and an observed user test.

07–09 — **Run a real service**
Data protection, pricing, billing and support.

10–11 — **Explore 10 business ideas**
Small software concepts with practical first tests.

12–14 — **Prepare your pilot**
Action plan, worksheet and source notes.

01 / UNDERSTAND IT

Maya builds a small tool

Fictional example: an order-deadline tracker for makers.



One task keeps returning

Maya sees makers checking several notebooks to find orders due soon. She sketches a tool that stores an order label, due date and status, then shows a daily due-soon list. She tests the list manually before building the app.

Micro-SaaS, in simple English

SaaS means software as a service: customers use software you operate, usually online. “Micro” describes a narrow focus and often a small team. It is not a promise of easy income. Someone still owns maintenance, reliability and customer support.

☺ Maya’s first rule

“Solve one repeated job well.” Her first version does not include a store, accounting system or AI assistant. Those are separate problems.

02 / FIND THE PROBLEM

Ask about the last time

Real behaviour is more useful than compliments about an idea.

1

SHOW ME

“Walk me through the last time you did this task. What did you use?”

2

FIND THE FRICTION

“Where did it slow down or go wrong? How often does that happen?”

3

CHECK THE ALTERNATIVE

“What have you already tried? What do you pay or spend time on now?”

4

TEST THE COMMITMENT

Show a small demo and a real pilot offer. Ask who will use it and who decides to pay.

Write down evidence, not guesses

Record the task, frequency, current workaround and exact pain point. Look for a repeated pattern among similar users. A free signup or friendly “yes” is not proof of willingness to pay.