Project Clarity Canvas

Decision-making tool for software projects

Instructions

Complete each section with precision. If a section cannot be completed with sufficient clarity, document why. That is an alert signal.

Use this canvas before committing to build. Not after.

---

1. Problem

What specific problem does this project solve?

- What real need exists?
- Why does this problem matter now?
- What happens if it's not solved?
- Who is affected by this problem?

Signal to observe: If the problem cannot be defined with precision, or if the solution is defined before the problem, the project should not be built. The solution must emerge from the problem, not precede it.

---

2. Success

How is the success of this project measured?

- What metrics indicate the problem was solved?
- What observable behavior changes?
- How is it validated that the solution works?
- What is the minimum success criterion?

Signal to observe: If success cannot be measured objectively, the project lacks validation criteria. Without metrics, validation is impossible.

---

3. Constraints

What real limits does this project have?

- Available budget
- Available time
- Technical resources
- External dependencies
- Regulations or compliance

Signal to observe: Healthy constraints improve design by forcing prioritization. Toxic constraints (constant pressure for technical shortcuts, scope changes without time or budget adjustment) systematically destroy quality.

---

4. Trade-offs

What is sacrificed and what is prioritized?

- Speed vs quality
- Features vs stability
- Cost vs time
- Scope vs depth

- Innovation vs risk

Signal to observe: If there's no honest conversation about trade-offs, or if results are promised without explicit sacrifices, expectations are misaligned. Every project involves sacrifices. Documenting them prevents later conflicts.

---

5. Non-objectives

What explicitly does this project NOT do?

- What is out of scope?
- What related problems does it not solve?
- What functionalities does it not include?
- What expectations must be adjusted?

Signal to observe: If limits are not clear, the project will grow without control and lose focus. Explicitly defining what is not built is as important as defining what is built.

---

6. Risks

What can go wrong and how is it mitigated?

- Technical risks
- Business risks
- Timeline risks
- Scope risks
- Mitigation plans

Signal to observe: If risks are not identified or minimized without justification, the project lacks realistic planning. Identifying risks is not pessimism. It's realism.

---

7. Decision

Should this project be built?

- Can all previous sections be completed with clarity?
- Are constraints healthy?
- Are trade-offs acceptable?
- Are risks manageable?
- Does the problem justify the investment?

Signal to observe: If the decision is "yes" but several previous sections are incomplete or ambiguous, reconsider. A well-founded "no" protects time, quality, and focus. An unfounded "yes" generates hidden costs.
