Managing Creative Work When the Ground Keeps Moving

A field guide for people working alone on creative projects under uncertainty

***AI**: I used AI tools to refine this paper. The ideas, style and arguments are my own.*

Author: Khaled Hafez

Published by: Prism380

Version 1.0 – 2026

CONTENTS

The Promise

The final “product” you deliver at the end of a creative project is, in most cases, not identical to what you thought it would be on day 1 of working on the project. The ground keeps moving, and so does the final shape of your project. Finishing a project is more complex than doing task A, then B and then C. Priorities change, ideas flow and then they don’t. What looked great yesterday may look disappointing today. You need some method to help you navigate this uncertainty without killing your creativity, and this is what this guide will do.

Where Did I Get This Method From?

Back in the day, projects were linear. You plan, budget, execute, review, finalize, release. As Information Technology got very complicated and fast-paced, linear planning was no longer effective. You can no longer start an IT project knowing exactly how it’s going to end because there are many unknown unknowns that will affect the final product along the path from inception to delivery.

Leaders of the IT community came up with an alternative non-linear method of managing projects (Agile software development) and were able to prove that it works. This guide is based on that approach but adapted to work for creativity projects. Don’t worry, it comes without the IT jargon and management meetings.

Why it Works?

The same element of uncertainty - unknown unknowns - that affects IT projects affects creativity projects as well. No writer starts a novel knowing how it will end, even if they do, they rarely know how they’re going to reach that ending. Creativity is about uncovering unknown unknowns. This guide helps you reach that destination without having to become a management expert.

The Big Pile of Dirt

Doing creative work is like moving a big pile of dirt from point A to point B with a spoon. The spoon in this analogy is your cognitive ability, which is inherently limited. Creativity is how the pile of dirt ends up looking when it finally reaches point B. The creative process, no matter how creative it is, involves having to move that big pile of dirt from point A to point B with a spoon, which can become boring, tedious, messy and unmotivating. In other words, creativity is fun; execution is boring, most of the time. It's the boring part is where people usually give up.

If you move too fast while carrying the dirt, you will probably end up losing a lot of dirt along the way. Also, you will get exhausted quickly. If you move too slowly, it will take you way longer to finish the work and you lose motivation or get distracted by other more interesting things.

Moving the big pile of dirt successfully is about finding the right pace. Not too fast, not too slow; and not giving up when things get boring.

The Principles

1. The highest priority is to deliver “output”. Output can be defined as a piece of work that can stand on its own, or a component of a larger work, a chorus of a song, or a chapter of a story.
2. Changes are welcome, even if late into the project. Do not discourage changes to the original plan. Whenever you think that a change is necessary, do not resist that thought. Consider it thoughtfully before rejecting it.
3. Work in short cycles. Two weeks is the default, adjust to your life and style.
4. Work at a pace that you can sustain, forever. Going too fast will lead to burn out, and too slow will lead to loss of motivation.
5. Always do your best. Deliver the highest quality you can achieve without getting stuck in an endless loop of perfectionism.
6. Don’t waste effort on non-essential work. Mark things that are essential and spend the majority of your time on them.
7. On the creative level, follow your own path, not the “right” path. You define the path that works for you.
8. Reflect, regularly. From time to time, reflect on your objective, your plan, and what needs to change, if any.

The Method

1. **Define your project objectives.**
   A rough definition would do at the start, but the definition should become clearer as you progress through the project. Also, having only one objective is ok, and having several is fine too, but having too many objectives means you’re running several projects, not one, which is not a problem only if you are aware of it.
2. **Identify the deliverables.**
   Identify what is the final product(s) the project will deliver. Create a list of the deliverables, with clear definitions of each deliverable. Again, definitions are normally not very clear at the beginning.
3. **Break down to parts.**
   Create a list of the tasks you need to perform to build the deliverables. This list will go through 2 phases. The first phase is having a list of large items. The second phase is the same large items, but with the first one or two items broken down into smaller items. The smaller items are doable within a cycle of work, more on that later.
4. **Prioritize the parts.**
   Set the list in order of priority so that you always work on the highest priority parts, not the parts that are fun to work on or easy to do.

5. **Pick one part to work on for the next work cycle.**
   Pick the FIRST item on the list, not any item. If the item is too big to finish in a work cycle, break it down further. If it's too small, you can add another small part as long as you don't exceed your work cycle.

   > **How to choose your cycle length**
   >
   > - **1 week if you're busy / easily derailed**
   > - **2 weeks is the default**
   > - **4 weeks if the project involves deep work that can't be broken down to smaller parts**

6. **Add/Remove parts.**
   Once you finish working on the first piece, go back to your list of parts. Answer the following questions:
   - Are there any parts that are no longer necessary? If yes, drop them!
   - Are there any parts I need to add? Add them to the list.

   Repeat the process frequently.

   > **How do you know you have finished the part?**
   >
   > **Set a clear definition of "done" before you start working on the item. For example, by the end of this task, I will have a 1-page summary of X.**

7. **Reprioritize the list.**
   Reprioritize the list considering the items you have added/deleted.
8. **Pick the next item to work on.**

2 Practical Working Methods

The Simplified Method,
Works just fine for most projects

Create 3 lists: to do, doing, done.

Ideas, parts, components all go to the “to do” list. Every 2 weeks, or whatever your “sprint” or cycle is, you pick an item from the “to do” list and place it in the “doing list”. The item you pick should be the highest priority, not the easiest, not the most fun. Here’s where you need the discipline shows and the productivity problems creep in.

> **Mandatory Rule**
>
> **You can only work on ONE item. You cannot have 2 items in the “doing” list.**
>
> **Multitasking ≠ Productivity.**
>
> Multi-tasking is a disruptive uncreative disorganized inefficient working method. Accept that and move on!

To-do	Doing	Done

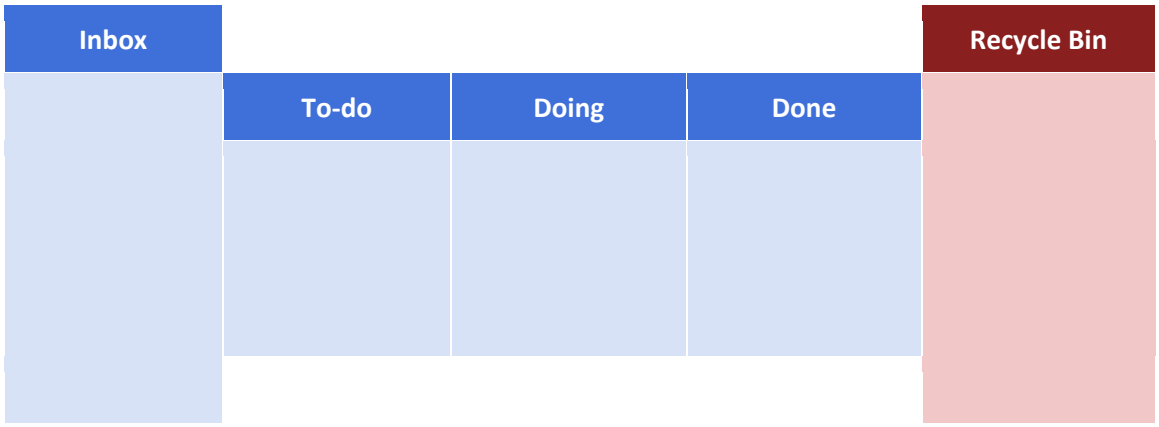
The Optimized Method,
Big, More Complex Projects

Instead of 3 lists, create five: inbox, prioritized to do list, doing, done, recycle bin.

The inbox is where you place all ideas without thinking about priorities, methods or anything related to organizing work. It’s a pure creativity box.

From time to time, you review the inbox only to add or remove items to it, no prioritizing or breaking down ideas in this list. Items that you remove from this list go to the recycle bin. You empty the bin only once you have finished the project. Or you might even keep it to use some of the items in it for other projects.

The prioritized to-do list includes your upcoming 3 to 4 tasks, broken down to small doable-in-2-weeks size. You can revise and reprioritize this list from time to time. No shame in taking items from this list to the recycle bin, or back to the inbox.



Software and Tools

You can use this method without relying on any technology, or using utility software like Google Sheets, or even Microsoft Word. If you find yourself comfortable with using some technology, keep it simple. Stick to one tool only! My favorites are Trello and OneNote. Trello is better suited for creating lists and moving items between them, OneNote is more suited for dumping creative ideas quickly and organizing them later. Tools are meant to make our life easier, if you find yourself spending more time on organizing your tools instead of doing actual work, it means you're using too many tools. Again, keep it minimal.

Running Multiple Projects

- **Keep them separate**: Different setting, different working environments, different time of the day, different device or tool.
- **Apply the 80/20 rule**: One project gets 80% of your attention, the other gets 20%.

Know When to Quit

Not all projects are meant to be completed. Knowing and deciding when to stop is emotionally difficult. Here are some rules from the finance world that are helpful.

You don't always have to give up the project entirely. You always have the following options:

1. **Put on hold**, do not scrap the work you've done, just put it aside for a few weeks or months, even years. If you see yourself going back to it several times and reconsidering it, that's a sign to go back to work.
2. **Slow down**: do not entirely give up, just slow down and see how things go.

3. **Reduce scope:** instead of a 12-song album, reduce your scope to a 4-tracks LP.
4. **Terminate:** Sometimes you need to give up, that's bravery.

Example Project

Project: Writing a 10-chapter non-fiction book

The exact content is not fully clear yet. The goal is to finish a *coherent, publishable manuscript*, not to perfectly predict everything from day one.

Planning Sheet (High-Level Only)

Working title	TBD
Target length	45,000 – 55,000 words
Initial structure	10 chapters
Audience	People interested in thoughtful, practical perspectives on the topic
Tone	Clear, calm, non-academic
Deadline (soft)	9–12 months
Constraints	▪ Working alone ▪ Limited daily energy ▪ 12 months budget ▪ Working on project competes with other responsibilities

Definition of "Done"

A complete manuscript that:

- Has a clear structure
- Can be read end-to-end without major confusion
- Is good enough to be edited, not rewritten

1. **Initial Decomposition (Parts, Not Tasks)**

   At the start, the project is broken into parts, not detailed tasks.

   At this stage, **chapter titles are allowed to be vague**. Precision comes later.

Initial parts list
▪ Overall book outline and scope ▪ Chapter 1–10 (placeholders, not final) ▪ Conceptual spine (what holds the book together) ▪ Introduction ▪ Conclusion

- Revision pass

2. **First Prioritization**

   Not all parts are equally important early on.

   Everything else stays deliberately fuzzy.

Top priorities
▪ Clarify the core idea of the book ▪ Draft one representative chapter ▪ Discover whether the book is viable as imagined

3. **First Working Cycle (2 Weeks)**

Sprint objective	Test whether the book idea can survive contact with reality
Selected part (ONLY ONE)	Draft Chapter 1 (rough, incomplete draft is fine)
Definition of done for this cycle	▪ A rough chapter draft (3,000–4,000 words) ▪ No concern for style or polish ▪ The chapter can be read without the need for further explanation
What is explicitly *not* done in this cycle	▪ No outline for all chapters ▪ No research rabbit holes ▪ No editing or beyond basic clarity ▪ Simplified formatting (headers and paragraphs, minimal graphics)

4. **End-of-Cycle Review**

   After 2 weeks, the answer the following:

   - Does this chapter *feel* like it belongs in a book?
   - Did the writing reveal a better structure than the original plan?
   - Is the original premise still valid?

   Outcome:

   - Two chapter ideas are dropped
   - One new chapter idea emerges
   - The book is reduced from 10 chapters to 8

   The plan changes. This is expected, not a failure.

Updated Parts List (Excerpt)
▪ Core premise (refined) ▪ Chapter 1 (exists) ▪ Chapter 2 (new focus) ▪ Chapter 3 (merged from two old ideas) ▪ Chapter 4–8 (placeholders) ▪ Introduction ▪ Conclusion ▪ Revision pass

The list is **simpler** than before, not more complex.

5. **Second Working Cycle (2 Weeks)**

Sprint objective	
Selected part (ONLY ONE)	Draft Chapter 2
Definition of done for this cycle	
What is explicitly *not* done in this cycle	

Rule still applies:

- Only one item in "Doing."
- During this cycle, if new ideas appear, they go to the **inbox**, not the current sprint.

6. **Mid-Project Adjustment (After Several Cycles)**

At some point:

- Energy drops when chapters exceed a certain length
- The book reads better with shorter chapters

Decision:
Reduce scope again.

- 8 chapters → 6 chapters
- Same total word count
- Clearer pacing

This is acceptable **optimization**, not quitting.

7. **When to Stop Writing New Chapters**

New chapter drafting stops when:

- All core ideas are expressed
- Additional chapters would repeat existing points
- Progress shifts from "adding value" to "adding volume"
- Emerging ideas do not belong to the core of the book. Do not waste these ideas, but do not add them unless they are essential.

At this point, the project naturally transitions into **revision cycles**, which are treated as separate parts.

8. **Final Note on the Example**

At no point did we:

- Fully plan all 10 chapters in advance
- Know exactly how the book would end
- Work on more than one major part at the same time

The book reached completion **because the plan was allowed to change**, not despite it.

Failure Modes

A project may succeed from a process perspective but still fail on the creative or perception level. On the other hand, rarely does a project fail on the process level and still succeed creatively. The failure modes below, which are not conclusive, are related to process failures, not talent or "creative worth" issues.

Mode 1 - Too much ambiguity	
Root cause	Objective not clearly defined
What goes wrong	Project expands indefinitely or drifts from main objective.
Early signs	Constantly adding items to the "to-do" column.
Corrective actions	▪ Clearly define objectives. ▪ Set strict rule to what can be part of the project and what gets rejected. ▪ If project is too large, divide to smaller projects.

Mode 2 - No clear definition of "done"	
Root cause	There is no clear criteria on when to consider a component, part or section of the work to be complete.
What goes wrong	Small tasks or parts never get fully completed. They keep expanding or changing indefinitely.
Early signs	▪ Constantly getting stuck in tasks that never seem to end.
Corrective actions	▪ Set a clear definition of what the item or part should be like when completed.

Mode 3 – Plan too detailed	
Root cause	You try to plan every small detail from the inception of the project.
What goes wrong	Instead of focusing on detailing a few tasks, you try to detail every task on the "to-do" column, or the items on the to-do column are too small.
Early signs	▪ You spend more time planning than doing actual work.

Corrective actions	▪ Set strict rule: only detail the upcoming 2 or 3 tasks.

Mode 4 – Jumping to next step too soon	
Root cause	Not giving yourself time between tasks and projects.
What goes wrong	Fatigue causes output to decrease in quality and/or quantity; creativity suffers.
Early signs	▪ You're exhausted more often. ▪ Project is going too fast.
Corrective actions	▪ Pace yourself. ▪ Take breaks.

Mode 5 – Lack of tangible output	
Root cause	You consider research as output and tend to overthink.
What goes wrong	Tasks never end or never start. You scrap too many parts.
Early signs	▪ Too many changes in plan. ▪ Items stay on the "doing" column for too long.
Corrective actions	▪ Limit research to a narrow focus. ▪ Do not list "research" as an item on your to-do list. ▪ If you find yourself overthinking, take a break.

Mode 6 - Perfectionism Disguised as Quality	
Root cause	Too much focus on small details.
What goes wrong	Tasks never end. Very low output rate. High error rate.
Early signs	▪ Stuck in planning phase. ▪ Stuck in the same task for too long.
Corrective actions	▪ Set priorities per output. Decide on what is essential and what is "nice-to-have" ▪ Set a clear test (checklist) on whether a part is complete. ▪ Stick to priorities. Finish the highest priority work first.