

TEAM TESS PLAYBOOK · THE SUBSCRIPTION EDUCATION SERIES

Raise the Rate

The Complete Lesson Plan Book

Project management & Lean Six Sigma, taught
from the industries that proved it

Fourteen modules. One concept each. One honest number, built piece by piece on your own work, until, at the capstone, your gauge page reads your rate and the distance it has travelled.

FIRST PUBLIC EDITION · OCTOBER 2026

The courses are COMING SOON. The taught editions (full lesson text and audio) open one module a week. This book is the whole map, now: every lesson plan for the entire course, with Module 1 printed complete as the series' proof.

Raise the Rate is included whole with the Team Tess Playbook subscription. This book is free and stays free.

What this book is, and is not

What it is. This book contains the complete lesson plans for the entire Raise the Rate course: all fourteen modules, in order, each with its one concept, its objectives, its source story and the evidence grade that story carries, the piece it builds in your gauge, its worked example, its field exercise in full, and the gate that opens the next module. It opens with the doctrine the course teaches and a plain account of how the course runs. **Module 1, “Your Number,” is printed in full:** complete lesson text, worked example, and field exercise, as the series' proof of quality: you can judge the whole course by its first module, because the first module is here, whole.

What it is not. It is not the taught editions of Modules 2–14. Those editions (the full lesson text and the audio for each module) **open weekly** as the courses launch: one module a week, in the order printed here. The plans in this book are complete; the lessons they plan are coming soon, on the cadence this book describes. Nothing in these pages is a summary offered in place of a lesson. The map is whole now so that you can see, before the first taught edition opens, exactly what you will build, in what order, and to what standard.

It is also not a credential. This course issues no belts and no certificates of attendance. Its one completion instrument is your own gauge page and the change in it (your delta) read at the capstone against the baseline you set in Module 1. That page is described in full in the pages that follow.

How to read this book. Read the doctrine first, it is short, and every module is an application of it. Then read “How the course runs,” so the shape of each lesson plan makes sense. Then read Module 1 complete: lesson, worked example, exercise. The thirteen plans that follow are written to the same standard and the same four beats, and each states plainly what its taught edition will contain when it opens.

Contents

Part I: The Doctrine: why nobody downstairs knows why, and the eight articles	Introduction
Part II: How the course runs: the four beats, the weekly cadence, the gauge you build, the completion instrument	Introduction
Module 1: Your Number (printed complete: lesson · worked example · field exercise)	Chapter 1
Module 2: Commitments You Can Count	Chapter 2
Module 3: Less Started, More Finished	Chapter 3
Module 4: Make the Error Impossible	Chapter 4

Module 5: Stop the Line	Chapter 5
Module 6: Report the Almost	Chapter 6
Module 7: Read It Back	Chapter 7
Module 8: Verified by Another	Chapter 8
Module 9: First Pass, Every Pass	Chapter 9
Module 10: The Ten-Minute Turn	Chapter 10
Module 11: One Plan, Four Gates	Chapter 11
Module 12: Small Batches, Fast Signal	Chapter 12
Module 13: The Weekly Review	Chapter 13
Module 14: Registers That Remember (Capstone)	Chapter 14
Your gauge, assembled: the one-page summary	After the plans
What happens next, and what this book cost to make	Closing page

A note on evidence grades, used throughout this book. Every source story is graded exactly as the study behind this course graded it: **RECORD**: a named, checkable source; **INFERENCE**: a reasoned transfer from the record, labelled as such; **FOLKLORE**: widely repeated, with no primary source, told only when it is instructive and labelled folklore on its face. Where a story carries a split grade, the plan prints the split. No plan in this book upgrades a grade.

The Doctrine

The standard the whole course teaches, stated once, in plain words, before the first lesson

The compartment disease

In a compartmentalized organization, decisions are made in separate rooms, at separate levels, and the reasons for them do not travel. Information moves in fragments, if it moves at all. So the people who carry the work (six levels down, and all the way across) end up asking whether they are in compliance: with a regulation, a standard, a directive. They ask whether, when they should be able to say why, why the requirement exists, in their own words, at their level.

Understanding sits at the top, held between a management layer and the bodies that govern, and it never reaches the floor. That is not a training gap. Training can hand a person an instruction; it cannot, by itself, make the reason for the instruction a thing the whole building holds. The gap is a doctrine gap: no standard in the building makes the why a deliverable owed to every level the work touches.

This course exists to close that gap in one operation at a time, yours. The doctrine below is the standard it teaches. Eight articles. Together they fit in your head, because a standard a practitioner cannot carry in their head will not survive contact with their building.

The eight articles

Article 1: End state first. Every effort opens from a committed end state: a named owner, a written end state, a date. Work missing any of the three is activity, not a project, and the doctrine does not pretend to manage it.

Article 2: One form, unchanged. All work reports in one uniform form (**Status · Owner · Next milestone · Gate · Metric**) and a fact written at the bottom reaches the top in the words it was written in, never summarized into somebody else's credit. The form is how truth travels; a workstream that reports in its own shape has left the standard.

Article 3: First-pass truth. Work counts when it crosses its delivery point clean the first time. Work that lands late, is never handed over, or lands with an error found afterward is an escape, written down permanently. Repairs are new work, never erasures.

Article 4: The gauge is the scoreboard. Every operation running this doctrine keeps one number: a fixed-formula composite of stage yields, counted under a permanent rule, impossible to improve by redefining it. The formula changes only in the open, by a dated amendment printed on the gauge's face, with the old series preserved.

Article 5: Gates beside the rate, held by named humans. What may be claimed (published, advertised, spent against) is governed by gates that print beside the number and never blend into it. And four decision classes never leave human hands: **spend, send, publish, change**. The system recommends, prices, and reports; a named person rules.

Article 6: Improvement arrives one small practice at a time. Change enters as one small, sourced practice at a time, restated by every process it touches, and judged on whether it moves the number it was adopted to move. An adoption that does not earn its place within its trial window is set aside, in the open, without blame. The list of practices an operation has declined (and why) is half the product: it is the record of fashions the operation will never have to un-adopt.

Article 7: The Tendrils Principle: WHY at every level. Comprehension is a deliverable of the system, owed at every level the work touches, down through the whole organization and all the way across it. A requirement is not discharged when it is transmitted; it is discharged when the person at the working level can state why it exists, not only that it exists. The doctrine's test is named **the Level Test**: pick any level, ask why, the answer must exist there, in that level's own words, or the system above it has failed, not the person. This article is the doctrine's answer to the compartment disease: decisions made in sealed rooms produce questions at the floor; reasons carried outward (by the one form of Article 2 and a standing rhythm of review) produce understanding at the floor. An operation in which, six levels down, a practitioner must ask whether they are in compliance has broken this article. The doctrine's aim is the operation where they can say why they are.

Article 8: Adoption by thread. No one is rebuilt at once. A practitioner, a team, an organization adopts this doctrine one thread at a time (one commitment practice, one report form, one honest number) in their own order, at their own pace, whole between steps. The person pulls the threads; the system never pulls the person.

What the doctrine is not. It is not a survey of project management, and it does not compete with the established bodies of knowledge on their ground. It teaches public methods (Toyota's system, the Last Planner, aviation's reporting culture, independent verification, queueing discipline) with their sources named, and it teaches the discipline of adopting them honestly. Its claim is narrow: an operation run on these eight articles will know its own number, will know why at every level, and will improve on evidence. That is the whole claim.

How the Course Runs

One concept a week · one artifact a week · one number the whole way

One concept per module, in four beats

One module teaches one concept. No module teaches two things. Every module runs the same four beats, because the structure is itself a lesson in standard work:

1. **The source story.** The concept taught from the industry that proved it, in that industry's real history, the plant, the construction site, the hospital floor. Evidence is graded on its face: RECORD, INFERENCE, or FOLKLORE, as defined at the front of this book.
2. **The translation.** The concept restated for your operation, whatever it is. Each translation names the gauge component the concept moves, so the course never drifts into inspiration detached from measurement.
3. **The worked example.** The house's own practice, shown at the WHAT level: the form, the numbers, the defects, anonymized, with its real figures, including the unflattering ones. Where the house's attempt failed or is incomplete, the example says so.
4. **The field exercise.** One exercise, run on your own work that week, producing one artifact. The artifact is the module. A subscriber who only reads has completed nothing, the app logs the artifact, and the next module's exercise builds on it.

The weekly cadence, and the gate

One module opens per week. The next module opens when you have logged the current module's field-exercise artifact, **logged, not graded**. The app records the artifact; nobody marks homework. Modules already opened stay open; the library is cumulative.

Why not all at once? Each exercise needs a working week to be real. A reliability score requires a week of commitments kept or missed; a census of open work means something only after a week lived under a limit. Weekly pacing with one artifact due turns the course from content to be consumed into a fourteen-week operating rhythm, the same rhythm the method itself prescribes.

Formats per module (design figures, stated from the course's format specification, the series has not yet run, and these are design figures, not measured results): the written lesson, about **20 minutes** of study reading · an audio version for the commute, about **19-24 minutes** per module, one voice, at a measured narration pace · the worked example, inside the lesson and as a one-page sheet in the app · the exercise sheet, **30-60 minutes** of work on your own

operation. Across the series: about 5 hours of audio and about 20 hours of total engagement. Module 1's figures are measured, not designed, and are printed in its chapter.

The gauge you build, piece by piece

The course is one long build. You assemble your own Lean Six Sigma gauge across the fourteen modules, in the order the house's gauge is assembled: production first, because the composite can never exceed it; delivery second; reporting throughout.

Stage	Modules	What your gauge gains
Skeleton	M1: Your Number	Component A (production yield), counted by hand · the formula, printed · the rule against redefinition, adopted in writing
Reliability	M2–M3	The commitment line (kept ÷ committed, with reason codes for misses) · the flow fields (work in process, aging, completions)
Reporting	M4	Component E (reporting conformance) + the exception queue that defends it
Candor	M5–M6	The in-process catch line · the near-miss and alert counts as leading indicators
Delivery	M7–M9	Component D (first-pass delivery yield) formalized · gate discipline · published reference bands to read it against
Speed	M10–M12	Changeover cost and batch-size lines, the levers that raise A without working harder
Governance	M13	One controllable input metric named per component · the weekly review rhythm
Capstone	M14	The full gauge page assembled · the formula placed under change control · the delta read against the M1 baseline · the gauge page issued as the completion instrument

Your gauge is complete on paper by Module 9. Modules 10–14 teach you to raise it and to govern it, and the capstone assembles the final page from parts you have already been running for weeks. That sequencing is deliberate: a gauge handed over finished would be furniture; a gauge built piece by piece is an instrument the builder trusts.

The completion instrument: your number is the credential

This course issues no belts and no certificates of attendance. A printed certificate attests that a person once sat through a course; this method exists because that attestation is not enough. So the course issues exactly one completion instrument: **your gauge delta**.

At Module 1, your first component is baselined and the page dated. At the capstone, the app assembles your full gauge page (components, formula, gates, and the delta against your Module

1 baseline, as of that date) and that page, printed from your own instruments, is the completion instrument. It says an operation measured itself honestly for fourteen weeks and its number moved by this much, under a formula that was never redefined along the way. A small delta and a large delta earn the same instrument; the page reports, it does not rank.

The subscription, the app, and this series

The series is included whole with the Team Tess Playbook subscription. There are no per-module unlocks for sale and no second paywall for the teaching: a subscriber who is paying to learn the method is never charged again for the next lesson in it.

The Playbook app holds the instruments the method needs, the uniform report form, the commitments board, and your gauge. **This series is the app's WHY layer:** the app is where the work is run; the series is where you learn why it is run that way, level by level, in the spirit of Article 7. And the app is **fully customizable:** its forms, boards, and gauge are yours to shape to your operation. The doctrine fixes the discipline (one form, one honest number, gates beside the rate) not the shape of your work.

Claims discipline: what may be said about this course. The course makes no claim to be the most efficient, the fastest, or the best of anything; no such league table exists. What may be said is: the method, the sources, and your own number. That is the whole advertisement, and it is enough.

Your Number

Build the first component of your own Lean Six Sigma gauge, and the rules that keep it honest

LESSON PLAN: MODULE 1

The one concept	Rolled yield; build your own gauge.
Objectives	By the end of this module you can: (1) state why rolled yield multiplies rather than averages, and compute it for three stage yields; (2) define a countable unit (finished and verified) in one sentence two people would apply the same way; (3) write a counting rule, including its exclusion clause, before counting anything; (4) count one production line by hand and publish Component A to one decimal, with exclusions printed.
Source story	Construction counts its promises, Glenn Ballard's reliability research RECORD · Six Sigma defect counting RECORD · Adoption-slate reference: the gauge itself.
Builds in your gauge	Component A (production yield), counted by hand · the formula, printed · the rule against redefinition, adopted in writing. (Capstone arc stage: Skeleton.)
Worked example	A real gauge, anonymized, printed in full in this chapter: A 27.2% · D 94.4% · E 100% · rate 25.7% · gates 0 of 51 and 0 of 14 · the day's three escapes, by class.
Field exercise	Build Component A: one production yield, counted by hand. Artifact: your eight-line Component A page, logged in the app. Printed in full in this chapter.
The gate	Artifact logged → Module 2 opens on the weekly cadence. Logged, not graded.
Formats & time	Measured for this module: lesson + worked example, 2,486 words, about 20 minutes of study reading · audio narration of the same lesson, 2,777-word script, about 19 minutes (the audio script is the narration of this lesson and is not reprinted in this book) · field exercise 30–60 minutes on your own work.

PART A

The Lesson

1. The promises nobody counted

Construction is an old industry, and for most of its history it ran on a sentence spoken in a trailer every Monday morning: here's what we'll get done this week. The sentence sounded like a plan. Nobody scored it. Then, in the 1990s, a researcher named Glenn Ballard started doing something almost rude in its simplicity, he wrote the promises down and counted how many were kept.

The number he found, across project after project, was this: without a system for making and scoring commitments, weekly plan reliability in construction typically ran between **30 and 55 percent**. Fewer than half the promises made on Monday were true by Friday. These were not lazy crews. They were competent people, working hard, inside a system that had never once asked whether its plans were real. The plans were weather reports nobody verified, and because nobody verified them, nobody could improve them, and week after week the same gap between promise and performance was treated as a surprise.

When Ballard's Last Planner System spread (commitments stated small, completable, and scored) reliability commonly rose to around **70 percent**, and there, honestly, it has tended to plateau. We will study that system in full in Module 2. For now, hold the founding fact of this entire series: **an industry that felt busy and professional was, by its own count, keeping fewer than half its promises, and it could not have told you that, because it had never counted**. (Evidence: RECORD: Ballard's dissertation research and two decades of replications. The figures above are the published ranges, not a vendor's brochure.)

Your operation has a version of that Monday sentence. It may be a release plan, a case queue, a content calendar, a client promise. Somewhere in it is a gap between what gets said and what gets done, a gap you currently experience as anecdotes: the rough week, the vendor who was late, the thing that almost shipped. This module replaces the anecdote with a number. Not a scorecard for someone else to grade you with. A gauge (an instrument you build, you own, and you read) whose first job is to tell you the truth about your own operation before anyone else does.

2. What a gauge is

A gauge is not a dashboard. A dashboard shows you everything and commits to nothing. A gauge answers **one question with one number**, built from components that each answer a plainer question, under rules written down before the counting starts.

The gauge you will build across this series asks three questions about your operation, in the order work actually flows:

1. **PRODUCE**. Of everything your operation is committed to produce, how much stands finished and verified? Call this **Component A: production yield**.
2. **DELIVER**. Of everything you handed over, how much landed clean the first time, on time, no error found afterward? Call this **Component D: first-pass delivery yield**. You will formalize it in Module 9.

3. **REPORT.** Do your own books foot? Do your records reconcile, and do they tell the truth about your position, including when the position is bad? Call this **Component E: reporting conformance**. You will build it in Module 4.

This week you build Component A, because A is the anchor: it is the component the whole gauge can never exceed, and it can only move when real work finishes. But the frame you build it in (the formula, the gates, the rule against redefinition) is the frame the finished gauge will live in. Build the frame carefully; you are not allowed to rebuild it quietly later. That, too, is part of the lesson.

3. Why the formula multiplies

Suppose your operation produces at 90 percent, delivers at 90 percent, and reports at 90 percent. What is your operation's rate? The comfortable answer is 90, the average. The true answer is **72.9**.

Work survives your operation the way a traveler survives a chain of connections. Of every 100 units that enter production, 90 come out finished. Of those 90, delivery keeps 90 percent: 81 arrive clean. Of those, reporting keeps 90 percent: 72.9 stand as work you can honestly claim. The stages do not average; they **compound**. Lean Six Sigma calls this rolled yield, and it is the first piece of arithmetic this series will ask you to internalize, because every soft dashboard you have ever seen uses the average, and the average is how a chain of three mediocre stages gets reported as excellence.

The formula (yours, from this week forward)

$$\text{RATE} = A \times D \times E = \text{production yield} \times \text{first-pass delivery yield} \times \text{reporting conformance}.$$

Until you build D and E, your rate is a production rate, A alone, labelled as such. The label is part of the honesty.

Multiplication has a second virtue: it forbids trading. Under an average, a great month in production can paper over a delivery collapse. Under a product, it cannot, a defect in any stage taxes the whole number, in the open, and a zero anywhere is a zero everywhere. When you see the worked example in Part B, watch for the shadow figure: a second gate, factored in, takes the example operation's rate to exactly 0.0%. The example prints that figure itself, beside the headline, on purpose. Your gauge will print its uncomfortable figures too. That is not pessimism. That is what makes the headline worth believing on the day it finally rises.

4. Component A, defined tightly enough to count

A production yield needs three things, and most home-made metrics die for lack of one of them:

- **An inventory.** A bounded, enumerable set of things your operation is committed to produce, cases, claims, documents, tickets, builds, files, units. If you cannot list it, you cannot count it, and “we're about 80% done” is the sound of an unlisted inventory.

- **A unit, defined so two people count it the same.** Not “worked on”, finished and verified. The definition of done must include the check: a unit counts when it has passed whatever verification your operation owes it, not when effort stopped.
- **A counting rule, written before the count.** What counts, what does not, when the count is taken, and who takes it. Written before, because a rule written after the count is not a rule; it is an alibi.

Component A is then one division: **verified units ÷ countable inventory units**: the inventory minus the items excluded under the rule, with the exclusions printed, never silently dropped. Counted by hand, the first time. Not sampled, not estimated, not pulled from a tool's cheerful default report. By hand, because the act of counting your own inventory, item by item, is where you will discover what your inventory actually contains: the duplicates, the zombies, the items that were never real. Every operation that has done this exercise has found them. Yours will too, and that discovery is worth more than the percentage.

5. Gates ride beside the rate

A rate tells you how you are doing. A **gate** tells you what you may claim. They are different instruments, and confusing them is how operations end up advertising things that are not true yet.

A gate is a count of units that have crossed a standard a rate can only approach gradually, for instance, units that have reached 100 percent, complete and cleared, out of all the units there are. Gates usually sit at zero for a long time, and that is the gate working, not failing: it exists to mark the distance between “well underway” and “done,” and to forbid the claim of done until it is done. Your gauge will carry gates beside the rate, always, never blended into it. Progress toward a gate is measured by a component (A climbs as units finish); the gate itself moves only when a unit crosses the whole standard. In the worked example, one gate stands at 0 of 51 and the other at 0 of 14, next to a healthy production yield of 27.2%. All three numbers are true at once. Learning to hold them at once (momentum and distance, without letting either one lie about the other) is most of what this series is for.

6. The formula is a promise

Here is the rule that separates a gauge from a mood. **Your formula is printed, dated, and fixed. It may be changed only in the open: the new formula, the reason, and the date written on the gauge's face, with the old series kept and the break marked.** A change may never be applied backward to raise a number already reported. If a future formula would read higher on the same facts, you report both numbers for a full cycle and let the lower one stand beside the higher until the new series has earned its own history.

Why so strict? Because the first time any measured operation feels pain, the temptation arrives, not to do better work, but to redefine the denominator. Exclude the hard cases. Count the drafts as done. Average instead of multiply. Each redefinition is small, reasonable, and fatal: the moment the formula bends, the gauge stops measuring the operation and starts measuring the operator's comfort, and it will never tell the truth again. Six Sigma's real inheritance is not the Greek letter; it is this refusal. Your number can only ever go up one way (by the work improving)

and that constraint is precisely what will make the number valuable to you, to a client, or to anyone you ever show it to.

7. Reading your first number

Your baseline will probably be lower than you expect. Construction's professionals found 30 to 55 percent. The worked example in Part B (a real operation's real gauge, anonymized) reads **25.7%** on the day it was first assembled, a day that operation's own register counts among its most productive ever: fifty-four deliverables closed, its books reconciled to the cent. Both things are true. The number was low because the standard was absolute and the inventory was honestly counted, not because the operation was idle.

So read your first number the way an instrument-maker reads a first calibration: not as a verdict, as a zero point. Everything this series does from here (the commitments, the work-in-process limits, the error-proofing, the candor practices) exists to move this number by moving the work, and your gauge, honestly kept, is how you will know whether any of it actually worked. A baseline you flinched from is a baseline you cannot improve on paper or in fact. Write it down exactly as it counts. Date it. That page is the first honest document your operation has produced about itself, and it is the instrument this whole series will eventually be judged by, yours, in Module 14, when we read the delta.

8. Where the series goes from here

Next week, Module 2 puts your commitments under the same discipline: one week of promises stated so that done is binary, scored as kept ÷ committed, Ballard's instrument, on your work. Module 3 caps the work you start, because your flow time is set by your inventory of open efforts, not by your effort. Module 4 builds Component E in your books. By Module 9 your gauge has all three components and reads as a true rolled yield. The rest of the series raises it, and Module 14 places the formula under change control and reads your delta. One concept a week, one artifact a week, one number the whole way. Open the exercise sheet (Part D) now, while the definitions are fresh, your inventory is waiting to be counted.

PART B

The Worked Example: a Real Gauge, Anonymized

The house's own operation, shown at the WHAT level: components, formula, gates, and the day's defects by class. Names and unpublished detail are withheld; every figure is real, from the gauge's inaugural census of October 24, 2026. The operation is a documentation-and-reconciliation program: a fixed inventory of items across multiple jurisdictions, each item counted only when reconciled and verified against its source of record.

25.7%

THE RATE (ROLLED YIELD)

0 of 51

GATE 1 · UNITS AT 100%,
COMPLETE & CLEARED

Caps what may be
claimed or published. At

0 of 14

GATE 2 · BUILD LEGS AT
FULL GRADE

Shadow figure, printed
beside the headline: with

= 27.2% production × 94.4% first-pass × 100% reporting

zero, the gate is the rule working.

this gate factored at 0/14, the fully-loaded rate is **0.0%**.

Comp.	What it asks	Count	Value
A	Production yield: of the committed inventory, how much stands reconciled and verified	217 of 799 items	27.2%
D	First-pass delivery: of the day's closed deliverables, how many landed clean the first time	51 of 54 clean · 3 escapes (below)	94.4%
E	Reporting conformance: the books foot and reconcile to the cent	Chains footed; zero unexplained entries	100%
E'	Position: reported separately, never blended into E	Operating buffer, current period	−\$40.16

Read E and E' together, the way the example prints them: the reporting is perfect and the position is tight, in the same week, on the same page. A true book in a tight week conforms; a padded book in a flush week would not.

The day's three escapes: printed unsoftened, by class

1 · Finished, never handed over. A deliverable was completed and sat in its folder for hours while the desk waited on it, never delivered, never reported. Under the counting rule, a file that was never handed over is unlanded: built is not delivered.

2 · Prepared, not surfaced at its hour. A report owed at a named hour was finished on time (text and audio verified) and then never presented; the recipient had to ask for it. A deliverable the recipient must request is a late land.

3 · A wrong fact, caught in re-verification. A document delivered as final carried a factual error on its face, found when the desk re-verified it before it travelled further. The corrected edition is owed and scheduled. Caught in-process work is counted separately (the same day logged **3 in-process catches, 0 escapes among them**) because the catching machinery deserves its own line, and because hiding catches would teach the desk to stop catching.

How to read this gauge: the headline (25.7%) is what survives the whole chain. Component A is the road, it moves only when items reconcile, roughly a tenth of a composite point per item, so the climb is honest arithmetic, not a mood swing. The gates (0 of 51, 0 of 14) say nothing is

finished to the full standard yet, momentum and distance, both true, neither lying about the other. When this operation's rate rises, it will rise the slow way: item by item, week by week, under a formula that has not moved since the day it was printed. That is the instrument you are building this week.

About the audio for this module. The taught edition's audio is the narration of this same lesson (one voice, signposted, with the arithmetic spoken slowly and the worked example's numbers walked through aloud) about 19 minutes. It teaches the same lesson by ear and is not reprinted here; this printed lesson is the document of record.

PART D

The Field Exercise: Count One Thing True

Your artifact: one Component A page, logged in the app. **Time:** 30–60 minutes, on your own work, this week. Module 2 opens on the weekly cadence once this page is logged, logged, not graded. Nobody marks this. The count is the teacher.

The page you are building

Your Component A page carries exactly eight lines. Nothing else goes on it:

1. The operation and the one production line you are measuring (one line only this week).
2. The **unit**: your definition of one finished, verified item, in one sentence a stranger could apply.
3. The **counting rule**: what counts as inventory, what is excluded and why, when the count is taken, who counts.
4. **Inventory**: the full count of units the line is committed to produce.
5. **Excluded under the rule**: items removed by the rule's own terms, named and counted, never silently dropped.
6. **Verified**: units finished and passed the check your definition requires.
7. **Component A**: $\text{verified} \div \text{countable inventory}$ (inventory minus the items excluded under the rule), as a percentage, one decimal.
8. The **date** and your name as counter. A count without a counter is a rumour.

The six steps

1. **Pick one production line (≈ 5 min).** The line where an outsider could ask “how many, and how do you know?” and you would currently have to wave your hands. Cases, claims, documents, tickets, builds, files, if your work feels uncountable, count deliverables or commitments; every operation finishes something. One line only: a broad count teaches nothing a narrow count doesn't teach faster.

2. **Define the unit (≈10 min).** Write the sentence: “One unit = _____, finished and verified by _____.” If two honest people applying your sentence to the same item could disagree, the sentence isn't done. Rewrite until they can't.
3. **Write the counting rule (≈5 min).** Before you count a single item. Include the exclusion clause now (duplicates, items that belong to another line, items opened in error) so that when you find them (you will), the rule, not your mood, decides what happens to them.
4. **Count the inventory by hand (≈15-30 min).** The whole denominator. No sampling, no tool reports, no “about.” List them or tally them, but touch every one.
5. **Count the verified (≈10 min).** Apply the unit definition item by item. “Mostly done” does not count. “Done but unchecked” does not count. The stringency is the point.
6. **Divide, date, sign, log (≈5 min).** One division. Write the percentage exactly as it comes out: 41.0% is a fine first number; a rounded-up 45% is a small lie that will compound. Enter the page in the app; your baseline is now of record.

Worksheet

#	Item (or tally block)	In inventory? (rule)	Verified? (definition)	Notes / exclusion reason
Totals		Inventory: _____	Verified: _____	Excluded under rule: _____ A = _____ %

A small worked example (so you can see the size of the thing)

A four-person billing team takes its application queue. Unit sentence: “One unit = one application processed end-to-end and passed by the second-person check.” Counting rule: everything in the quarter's queue log counts; duplicates and another team's misroutes are excluded under the rule and printed. Two afternoons of counting by hand: **inventory 1,240 · excluded under the rule 111 (61 duplicates, 38 misroutes, 12 opened in error) · verified 986**. Component A, on the countable inventory (the full inventory of 1,240 minus the 111 excluded under the rule, which is 1,129: **986 ÷ 1,129 = 87.3%**) with the exclusions printed

on the same page, so any reader can recompute it either way. The team lead's comment, recorded under the count: "We thought we were behind on processing. We're actually behind on queue hygiene. Different problem." That sentence is what this exercise is for.

Pitfalls, each one has ended a real gauge:

- **Denominator shopping:** choosing the inventory that makes the number kindest. The rule is written first precisely so the inventory can't be chosen after its contents are known.
- **Counting drafts:** "done except for the check" is not done. The definition includes verification or it includes nothing.
- **Mid-count redefinition:** if the definition proves broken, stop, rewrite the page, recount from zero under the new rule, and keep the broken page. A recounted baseline is honest; a patched one is fiction.
- **Sampling:** an estimate of your inventory is an opinion about your inventory. First count: everything, by hand. Later counts may use your systems, once your systems have survived comparison with your hands.
- **Rounding toward comfort:** report one decimal, as computed. The gauge's credibility is built from a hundred small refusals like this one.

Done looks like this: eight lines on a page, a real denominator you touched item by item, a percentage you did not round toward comfort, your name and the date at the bottom. Log it. Next week, Module 2 (Commitments You Can Count) puts your weekly promises under the discipline Ballard brought to the trailer: stated so done is binary, scored as kept ÷ committed, with a reason code on every miss. Your gauge is about to get its second line, and your operation its first reliability number.

What this module teaches, shows, and never shows, stated once for Module 1 and in force for every module:

Teaches: rolled yield and gauge construction (public Lean Six Sigma method) and Ballard's published reliability research. **Shows:** the house's gauge at the WHAT level (components, formula, gates, real figures, defects by class) anonymized. **Never:** the house's unpublished detail or any principal's private information. Teaching the method is the product.

Commitments You Can Count

Promises stated so that “done” is binary, and scored

LESSON PLAN: MODULE 2

The one concept	PPC, percent plan complete: promises stated so done is binary, scored as kept ÷ committed.
Objectives	By the end of this module you can: (1) restate a week's work as binary commitments, each either done or not done, with no third state; (2) score the week's PPC as kept ÷ committed, to one decimal; (3) attach a reason code to every miss and name the week's most frequent code; (4) state two ways PPC gets gamed and the check that catches each.
Source story	The Last Planner System, Glenn Ballard RECORD · the ~70% plateau in plan reliability, and how PPC gets gamed RECORD . Adoption-slate reference: A8.
Builds in your gauge	The commitment line: kept ÷ committed, with reason codes for misses. (Capstone arc stage: Reliability.)
Worked example	Shows the house's own weekly commitment set at the WHAT level: commitments as written, the week's PPC as counted, and the misses with their reason codes, including any week the score fell.
Formats & time	Design figures: lesson ≈ 20 minutes · audio ≈ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 3 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: one week of binary commitments, with the first PPC scored, logged in the app as your commitment page for the week.

Spec. (1) Write one week of commitments before the week begins; each commitment is small, completable within the week, and stated so that done is binary, a reader could score it without asking you. (2) Keep the list as written; do not edit a commitment mid-week to make it easier, a changed promise is a miss with a reason code, not a rewrite. (3) At week's end, score each commitment kept or not kept and compute $PPC = \text{kept} \div \text{committed}$, one decimal. (4) Attach a reason code to every miss, in your own short code set, and count the codes. (5) Date and sign the page; log it. **Done looks like:** a week's promises a stranger could score, one PPC figure, and a reason code on every miss.

Less Started, More Finished

Limit the work in process, and let flow time fall

LESSON PLAN: MODULE 3

The one concept	Work-in-process limits; Little's Law, flow time is set by how much is open, not by how hard people work.
Objectives	By the end of this module you can: (1) census every open effort in your operation and state the count; (2) state Little's Law in your own words and apply it to your own open work; (3) set a first work-in-process cap in writing, with the date it takes effect; (4) name what happens to new work when the cap is full, before the situation arises.
Source story	Queueing theory's identity, Little's Law RECORD : a record by construction: it holds as arithmetic · Toyota's kanban system RECORD . Adoption-slate reference: A10.
Builds in your gauge	The flow fields: work in process, aging, and completions. (Capstone arc stage: Reliability.)
Worked example	Shows the house's own open-work census at the WHAT level: the count as found, the cap as set, and what the census revealed about aging work.
Formats & time	Design figures: lesson $\approx$ 20 minutes · audio $\approx$ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 4 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: your work-in-process census, with your first cap set in writing, logged in the app.

Spec. (1) Census your own work in process: list every open effort, with its age, started when, untouched since when. Count them; the count is the first line of the artifact. (2) Write the flow fields beside the census: open count, oldest age, completions in the last full week. (3) Set your first cap in writing: the maximum number of efforts open at once, the date it takes effect, and the rule for new work when the cap is full. (4) Date and sign; log it. **Done looks like:** a complete census you can defend item by item, three flow fields, and a cap with a date on it.

Make the Error Impossible

Design the check into the work, and keep the books continuously true

LESSON PLAN: MODULE 4

The one concept	Poka-yoke: error-proofing; and continuous close: records reconciled as work happens, not reconstructed afterward.
Objectives	By the end of this module you can: (1) name one recurring error in your operation and state how often it has occurred in a stated window; (2) build one validation that makes that error impossible (not discouraged, impossible) and show it working; (3) state your operation's reporting-conformance rule: what "the books foot" means, line by line; (4) open an exception queue that holds every item your conformance rule cannot yet clear.
Source story	Payer first-pass yield discipline RECORD as a discipline; its exact bands INFERENCE : the plan prints the split grade · ThedaCare medication reconciliation RECORD . Adoption-slate reference: A16.
Builds in your gauge	Component E (reporting conformance) + the exception queue that defends it. (Capstone arc stage: Reporting.)
Worked example	Shows the house's own books at the WHAT level: the conformance rule, the reconciliation as counted, and the exception queue with its open items named by class.
Formats & time	Design figures: lesson ≈ 20 minutes · audio ≈ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 5 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: one validation, built and working, that makes one recurring error impossible, with the error, the validation, and the proof logged in the app.

Spec. (1) Find one recurring error: an error your operation has made more than once, stated with the window you counted it in. (2) Build one validation at the point the error occurs, a check the work cannot pass while the error is present. A reminder, a poster, or a sterner instruction does not meet this spec; the error must be unable to proceed. (3) Demonstrate it: run the old error at the validation once and record the result. (4) In the same week, write your Component E rule and open your exception queue with its first items. (5) Date and sign; log it. **Done looks**

like: an error that used to recur, a validation that stops it cold, the stopped attempt on record, and an exception queue holding what remains.

Stop the Line

Authority to halt the work, held by the people doing it

LESSON PLAN: MODULE 5

The one concept	Andon authority: any person who sees a defect may stop the work, and the stop is honoured, not punished.
Objectives	By the end of this module you can: (1) state, from the source story, what changed when stop authority reached the floor; (2) write your team's stop rule (who may stop what, for what, and what happens next) in one paragraph; (3) open a save log that records each stop, what it caught, and what it cost; (4) state the rule that protects the person who stopped the line in good faith.
Source story	NUMMI: General Motors' worst-performing plant, reopened under Toyota's system, transformed within one year RECORD . Adoption-slate reference: A1.
Builds in your gauge	The in-process catch line: defects caught while the work can still be saved, counted separately from escapes. (Capstone arc stage: Candor.)
Worked example	Shows the house's own stop practice at the WHAT level: the rule as written, the stops as logged, and what each stop caught before it could escape.
Formats & time	Design figures: lesson ≈ 20 minutes · audio ≈ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 6 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: your team's stop rule, written (with the save log opened) logged in the app.

Spec. (1) Write the stop rule in one paragraph: who may stop the work, on seeing what, by what act; what the team does when a stop is called; and how work restarts. (2) State, in the same artifact, the protection: a good-faith stop is recorded as a save, never as a fault. (3) Open the save log with its columns: date · who stopped · what was caught · what it would have cost if it had escaped · how work restarted. (4) Publish the rule to the team it governs, a rule nobody has seen stops nothing. (5) Date and sign; log it. **Done looks like:** a stop rule your team has actually received, and a save log standing open for its first entry.

Report the Almost

Near misses are free lessons, if the system thanks the reporter

LESSON PLAN: MODULE 6

The one concept	Near-miss learning; safety alerts: report what almost happened, count it, and learn before the escape arrives.
Objectives	By the end of this module you can: (1) distinguish an escape, an in-process catch, and a near miss, and classify three examples correctly; (2) open a near-miss register with a filing rule a busy person can actually follow; (3) file the first three near misses from your own operation, each with what almost happened and what would have followed; (4) state why a rising near-miss count is read as a leading indicator, not as deteriorating work.
Source story	NASA's Aviation Safety Reporting System, operating since 1976 RECORD · Virginia Mason's patient safety alert counts RECORD . Adoption-slate references: A3 · A2.
Builds in your gauge	The near-miss and alert counts as leading indicators, beside the catch line. (Capstone arc stage: Candor.)
Worked example	Shows the house's own near-miss practice at the WHAT level: the register's shape, the alerts as counted, and one near miss that changed a rule before it could become an escape.
Formats & time	Design figures: lesson ≈ 20 minutes · audio ≈ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 7 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: a near-miss register, opened (with the first three near misses filed) logged in the app.

Spec. (1) Open the register with its fields: date · reporter · what almost happened · what would have followed · what, if anything, changed as a result. (2) Write the filing rule on the register's face: who may file, how long a filing takes, and the standing assurance that a filing is thanked, never blamed. (3) File the first three near misses from your own work, real ones, however small; an “almost” you nearly forgot counts. (4) Date and sign; log it. **Done looks like:** an open register, a filing rule that protects the reporter, and three honest almos standing in it.

Read It Back

Checklists with read-back, on the steps that cannot be undone

LESSON PLAN: MODULE 7

The one concept	Checklists with read-back on irreversible steps: the doer reads the step aloud; another voice confirms it before the step is taken.
Objectives	By the end of this module you can: (1) name the steps in your operation that are irreversible, and say what makes each one so; (2) write a checklist for one irreversible step that fits the moment it is used in; (3) run one read-back on live work and record what it caught or confirmed; (4) state the difference between a checklist that is performed and one that is merely possessed.
Source story	Aviation checklist culture and crew resource management RECORD . Adoption-slate reference: A7.
Builds in your gauge	Gate discipline: the first of the delivery-stage practices that keep Component D honest. (Capstone arc stage: Delivery.)
Worked example	Shows the house's own checklist practice at the WHAT level: the irreversible step chosen, its checklist as written, and a read-back as it actually ran.
Formats & time	Design figures: lesson $\approx$ 20 minutes · audio $\approx$ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 8 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: one checklist for one irreversible step (with one read-back run and recorded) logged in the app.

Spec. (1) Pick one irreversible step: a step whose error cannot be cheaply undone, sent, published, deleted, administered, signed. (2) Write its checklist: the few checks that must pass before the step, in the order they must pass, short enough to be used at full speed. (3) Run one read-back on live work: one person works the list aloud, a second voice confirms each item before the step proceeds. (4) Record the run: date, step, who read, who confirmed, anything the read-back caught. (5) Date and sign; log it. **Done looks like:** a checklist short enough to survive contact with a busy day, and one real read-back standing behind it.

Verified by Another

The verifier is never the builder

LESSON PLAN: MODULE 8

The one concept	Independent verification: work is verified by someone who did not build it, verifier $\neq$ builder, as a rule, not a courtesy.
Objectives	By the end of this module you can: (1) state why self-verification misses what independent verification catches, from the source record; (2) name a verifier who is not the builder for one live deliverable, with the verification standard stated in advance; (3) record the verification's outcome (passed, or returned with findings) without editing the record; (4) state what happens to a finding: who owns the repair, and why the repair is new work, not an erasure.
Source story	Defense and NASA independent verification and validation, learned in writing and at cost RECORD . Adoption-slate reference: A11.
Builds in your gauge	Gate discipline, completed: verification standing between delivery and claim. (Capstone arc stage: Delivery.)
Worked example	Shows the house's own verification practice at the WHAT level: a deliverable, its independent verifier, and the findings as recorded, including a return.
Formats & time	Design figures: lesson $\approx$ 20 minutes · audio $\approx$ 19-24 minutes · field exercise 30-60 minutes.
The gate	Artifact logged → Module 9 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: one live deliverable under independent verification (verifier named, standard stated, outcome recorded) logged in the app.

Spec. (1) Choose one live deliverable now in production. (2) Name its verifier: a person who did not build any part of it. If everyone touched it, the verifier is the person furthest from its making, and the artifact says so. (3) State the verification standard before the verifier begins: what “passes” means, item by item. (4) Record the outcome exactly as returned, passed, or returned with findings listed. (5) Date and sign; log it. **Done looks like:** a named verifier who built nothing on the item, a standard written before the check, and an outcome recorded without softening.

First Pass, Every Pass

Count delivery the way it landed, the first time

LESSON PLAN: MODULE 9

The one concept	First-pass yield: of everything handed over, how much landed clean the first time, with the four delivery keys, translated for any operation.
Objectives	By the end of this module you can: (1) define “landed clean” for your operation (on time, handed over, no error found afterward) in one sentence; (2) formalize Component D under a counting rule written before the count; (3) count one full week of first-pass delivery and state D to one decimal; (4) read your D against published reference bands and state, in writing, what the comparison does and does not entitle you to claim.
Source story	DORA's decade of published delivery research and its performance bands RECORD . Adoption-slate reference: A9.
Builds in your gauge	Component D (first-pass delivery yield), formalized, the gauge now reads as a true rolled yield: $\text{RATE} = A \times D \times E$. (Capstone arc stage: Delivery, completing the gauge on paper.)
Worked example	Shows the house's own delivery counting at the WHAT level: the week's handovers, the clean count, the escapes by class, and D as computed, the same counting the Module 1 example showed at 94.4%.
Formats & time	Design figures: lesson $\approx$ 20 minutes · audio $\approx$ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 10 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: Component D, formalized (one week of first-pass delivery, counted) logged in the app as your Component D page.

Spec. (1) Write the counting rule first: what counts as a handover, what “clean” means, when the count closes, who counts. (2) Count one full week: every handover, marked clean or escape, with escapes classed, late, unlanded, or error found afterward. (3) Compute $D = \text{clean} \div \text{handed over}$, one decimal, and print it beside your A and E so the gauge reads whole for the first time. (4) Write one sentence reading D against the published bands, and one sentence on what the

comparison does not prove. (5) Date and sign; log it. **Done looks like:** a rule written before its count, a full week honestly classed, and a three-component gauge on one page.

The Ten-Minute Turn

Make the changeover cheap, and small runs become possible

LESSON PLAN: MODULE 10

The one concept	SMED, single-minute exchange of die: separate the changeover steps that must happen while work is stopped from those that can happen while it runs, and move the second kind out.
Objectives	By the end of this module you can: (1) map one changeover you repeat, step by step, with the stopped time stated; (2) classify each step as must-be-stopped or can-run-alongside, and defend the classification; (3) move one step out of the stopped window and re-measure the changeover; (4) state your changeover cost as a line on your gauge's supporting page.
Source story	Shigeo Shingo's press changeovers RECORD · Southwest Airlines' 1972 ten-minute turn RECORD . Adoption-slate reference: A4.
Builds in your gauge	The changeover-cost line: a lever that raises A without working harder. (Capstone arc stage: Speed.)
Worked example	Shows the house's own changeover practice at the WHAT level: a repeated switch of work, its steps as mapped, and the step moved out of the stopped window.
Formats & time	Design figures: lesson $\approx$ 20 minutes · audio $\approx$ 19-24 minutes · field exercise 30-60 minutes.
The gate	Artifact logged → Module 11 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: one changeover, mapped (with one step moved out of the stopped window and the new changeover time measured) logged in the app.

Spec. (1) Map one changeover you repeat: switching tasks, tools, cases, or contexts. List every step in order and time the stopped portion honestly. (2) Classify each step: must happen while stopped, or could happen before or after while other work runs. (3) Move one step of the second kind out of the stopped window (prepare it in advance, or finish it afterward) and run the changeover once under the new arrangement. (4) Record both times: before, after. (5) Date and sign; log it. **Done looks like:** a step map a colleague could follow, one step genuinely relocated, and two honest times.

One Plan, Four Gates

One plan of record for each effort, reviewed at gates, worked to a standard

LESSON PLAN: MODULE 11

The one concept	Case conferences, tollgates, and standard work: one plan of record per effort, the whole team conferring on it, progress ruled at four gates.
Objectives	By the end of this module you can: (1) open one effort under a single plan of record that names owner, end state, and date (Article 1); (2) define four gates for that effort, each with the evidence that passes it; (3) state the standard work for the effort's recurring tasks, the current best way, written down; (4) hold the first gate conference on the plan and record its ruling.
Source story	ThedaCare Collaborative Care RECORD · Toyota Production System standard work RECORD . Adoption-slate references: A6 · A5.
Builds in your gauge	The batch-size and flow discipline that supports the Speed stage: efforts sized and gated so signal arrives early. (Capstone arc stage: Speed.)
Worked example	Shows the house's own plan-of-record practice at the WHAT level: one effort, its single plan, its four gates, and a gate ruling as recorded.
Formats & time	Design figures: lesson ≈ 20 minutes · audio ≈ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 12 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: one effort opened under a single plan of record with four gates, the plan and its first gate ruling logged in the app.

Spec. (1) Choose one live effort and write its single plan of record: owner, end state, date, and the uniform form's fields: Status · Owner · Next milestone · Gate · Metric. Where older plans for the same effort exist, the artifact names this one as the record and says what happens to the others. (2) Define four gates across the effort's life, each passable only on stated evidence. (3) Write the standard work for one recurring task inside the effort. (4) Confer on the plan with everyone the effort touches, rule at the first gate, and record the ruling. (5) Date and sign; log it.

Done looks like: one plan everyone can point to, four gates with evidence attached, and a first ruling on record.

Small Batches, Fast Signal

Ship less at once, and learn sooner whether it works

LESSON PLAN: MODULE 12

The one concept	The small-batch rule: work in batches small enough that the signal from the first piece arrives before the rest is built.
Objectives	By the end of this module you can: (1) name one batch you currently ship whole and state its size; (2) split it so a first portion can ship and be judged on its own; (3) define, in advance, the signal you will read from the first portion and what each possible reading will make you do; (4) record the signal as read, including when it says stop.
Source story	Zara's design-to-store cycle RECORD . Adoption-slate reference: A15.
Builds in your gauge	The batch-size line: the second Speed lever: smaller batches, faster correction, higher A. (Capstone arc stage: Speed.)
Worked example	Shows the house's own batch practice at the WHAT level: a batch as formerly shipped, the split as made, and the signal read from the first portion.
Formats & time	Design figures: lesson $\approx$ 20 minutes · audio $\approx$ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → Module 13 opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: one batch, split (with the signal from its first portion read and recorded before the rest is built) logged in the app.

Spec. (1) Split one batch you ship: choose the smallest first portion that can stand judgement on its own. (2) Before shipping it, write the signal: what you will look at, when, and what each reading decides, proceed, adjust, or stop. (3) Ship the first portion and read the signal at the stated time. (4) Record the reading and the decision it produced, exactly as they happened, a “stop” faithfully recorded earns this artifact in full. (5) Date and sign; log it. **Done looks like:** a batch that no longer travels whole, a signal defined before it could be wished for, and a decision that followed the reading.

The Weekly Review

Review by exception, and steer by the inputs you control

LESSON PLAN: MODULE 13

The one concept	Exception-driven review; input metrics: review what departed from standard, and steer by one controllable input per component, not by the outcome alone.
Objectives	By the end of this module you can: (1) name one controllable input metric for each gauge component (A, D, and E) that your operation can move this week; (2) distinguish each input from the outcome it steers, in writing; (3) run one weekly review by exception: the departures from standard, in order, with rulings recorded; (4) set the review's standing rhythm, day, hour, attendees, and the page it reads from.
Source story	Amazon's Weekly Business Review RECORD : graded as what it is: a participant account, carried at that strength. Adoption-slate references: A13 · A14.
Builds in your gauge	One controllable input metric named per component · the weekly review rhythm. (Capstone arc stage: Governance.)
Worked example	Shows the house's own weekly review at the WHAT level: the page it reads, the exceptions as listed, and the rulings as recorded.
Formats & time	Design figures: lesson ≈ 20 minutes · audio ≈ 19-24 minutes · field exercise 30-60 minutes.
The gate	Artifact logged → Module 14, the capstone, opens on the weekly cadence. Logged, not graded.

The field exercise, in full

Artifact: your input-metric page (one controllable input per component) with one weekly review run and its rulings recorded, logged in the app.

Spec. (1) Name one controllable input per gauge component: for A, D, and E, an input your operation directly controls this week (not the component itself, and not a hope). Write beside each input the outcome it is meant to steer. (2) Convene one review on the exception rule: only departures from standard are discussed, in the order the gauge page lists them. (3) Record the rulings: what was decided, by whom, by when. (4) Fix the rhythm in writing: the standing day, hour, and page for every future review. (5) Date and sign; log it. **Done looks like:** three inputs

you can actually pull, one review that ruled on exceptions only, and a rhythm that will still be there next week.

Registers That Remember

Capstone, place the gauge under change control, assemble the page, read the delta

LESSON PLAN: MODULE 14 · CAPSTONE

The one concept	Change control on records of record: the register remembers; changes are amendments, printed and dated, never edits that rewrite the past.
Objectives	By the end of this module you can: (1) state the change-control rule for your gauge: who may amend the formula or a counting rule, how, and with what printed on the page; (2) assemble your full gauge page from the components and lines built across Modules 1–13, with nothing recomputed under a new rule; (3) read your delta against your Module 1 baseline and state it exactly as counted; (4) issue your gauge page as your completion instrument, dated, with its gates standing beside its rate.
Source story	Defense configuration management RECORD . Adoption-slate reference: A12.
Builds in your gauge	The capstone itself: the full gauge page assembled · the formula placed under change control · the delta read against the M1 baseline · the gauge page issued. (Capstone arc stage: Capstone.)
Worked example	Shows the house's own gauge page at the WHAT level, assembled: components, formula, gates, and position line, the page Module 1 showed on its first day, carried forward under an unmoved formula.
Formats & time	Design figures: lesson $\approx$ 20 minutes · audio $\approx$ 19–24 minutes · field exercise 30–60 minutes.
The gate	Artifact logged → the course is complete. The logged capstone artifact is the assembled gauge page itself.

The field exercise, in full: the capstone

Artifact: your sealed gauge page (the assembled page, formula printed, delta read) logged in the app. This page is the course's completion instrument.

Spec. (1) Assemble the page from parts already built: Component A (M1), the commitment line (M2), the flow fields (M3), Component E and its exception queue (M4), the catch, near-miss, and alert lines (M5–M6), the delivery practices and Component D (M7–M9), the changeover and batch-size lines (M10–M12), and the input metrics and review rhythm (M13). Nothing on the

page is recomputed under a new rule for the occasion. (2) Print the formula on the page's face ($\text{RATE} = A \times D \times E$) with the date it was adopted in Module 1. (3) Place the formula and every counting rule under change control in writing: amendment only in the open, new rule and reason and date printed, old series kept, no amendment ever applied backward. (4) Read the delta: today's rate against your Module 1 baseline, stated exactly, with the gates standing beside the rate. (5) Date and sign; log it. **Done looks like:** one page that says what your operation's number is, how it is computed, what may not yet be claimed, and how far the number has travelled since the week you first counted it, under a formula that never moved.

Your Gauge, Assembled

The whole instrument on one page, what fourteen weeks build

The three components

Comp.	Question it answers	Built in	Definition
A	PRODUCE: of everything committed, how much stands finished and verified?	M1; raised by M10-M12	Verified units ÷ countable inventory units (inventory minus items excluded under the rule, exclusions printed), counted under a rule written before the count
D	DELIVER: of everything handed over, how much landed clean the first time?	Formalized in M9; defended by M7-M8	Clean first-pass handovers ÷ all handovers in the window
E	REPORT: do the books foot, and do they tell the truth about the position?	M4; defended by its exception queue	Reporting conformance under the operation's stated rule; position reported separately, never blended in

Supporting lines stand on the same page: the commitment line (kept ÷ committed, with reason codes), the flow fields (work in process, aging, completions), the in-process catch line, the near-miss and alert counts, the changeover-cost and batch-size lines, and one controllable input metric per component, reviewed weekly by exception.

$$\text{RATE} = \text{A} \times \text{D} \times \text{E}$$

Production yield × first-pass delivery yield × reporting conformance. The stages multiply; they never average. A defect in any stage taxes the whole number, in the open, and a zero anywhere is a zero everywhere.

Gates beside the rate, always

The rate says how you are doing; the gates say what you may claim. Gates count units that have crossed a whole standard (complete and cleared) out of all the units there are. They print beside the rate, never blended into it. A gate at zero beside a climbing rate is not a contradiction; it is momentum and distance, both true, neither lying about the other.

The delta

Your completion reading is a subtraction, honestly come by: **today's rate, on the assembled page, minus your Module 1 baseline**: both counted under the same formula and the same rules, because the formula was never redefined along the way. That is why the delta can be believed: not because it is large, but because nothing between the two readings was allowed to move except the work.

What Happens Next

The map is in your hands; the lessons now open, one a week

The courses open weekly. Module 1's taught edition (the lesson you have just read in full, with its audio) opens first. Each following module opens one week after the last, in the order of this book, and opens for you when you log the current module's field-exercise artifact. Opened modules stay open. The series is included whole with the Team Tess Playbook subscription; there is nothing further to buy to finish this course.

This book stays free. It is the whole map, and maps should be free: anyone considering the course can read every plan, judge Module 1 complete, and know exactly what the fourteen weeks ask and build, before the first taught edition opens.

What may be said about this course: the claims discipline, in one line: the method, the sources, and your own number, nothing more is claimed, and nothing more needs to be.

Cost of this book. Printed here because a course that teaches honest counting counts itself.

External cash spent to build this book: **\$0.00**.

Compute: **UNMEASURED**: the house meters its compute in aggregate only, so no per-book compute figure exists, and none is claimed.

Raise the Rate: The Complete Lesson Plan Book · First public edition · October 2026 · Team Tess Playbook.
Sources of record: the Series Bible for Raise the Rate (October 24, 2026), product definition, four-beat module structure, capstone arc, fourteen-module map, cadence, tiers, completion instrument, and claims discipline · Module 1, "Your Number," complete (lesson, worked example, field exercise; audio is the narration of the same lesson) · the doctrine's eight articles, including Article 7, the Tendrils Principle. Every figure in this book traces to those sources; times for Modules 2–14 are design figures from the format specification and are labelled as such wherever they appear.