Taazaa.

REAL ESTATE & FINANCE PRACTICE

The AI Implementation Layer for REITs

10 practical AI workflows you can build this week. With step-by-step instructions, copy-and-paste prompts, and the guardrails that keep them safe.

A field guide for acquisitions, asset management, investor relations, and finance teams.

By **Mike Hammerslag**, Real Estate Technology Advisor
Taazaa · Human-centered software and AI engineering

INSIDE THIS GUIDE

Contents

A short book with a narrow goal. Read the front matter once, then keep the 10 workflows on your desk.

Foreword: AI Has to Survive the Underwriting Model 3

What the Implementation Layer Actually Is 4

How to Use This Book 6

THE 10 WORKFLOWS

01 Lease Abstraction and Key-Term Extraction 8

02 First-Pass Deal Screening Memo 10

03 Budget vs. Actual Variance Narrative 12

04 Rent Roll Risk and Rollover Analysis 14

05 Submarket Comp and Market Brief 16

06 Investor Relations Q&A and Earnings Prep 18

07 ESG and Sustainability Data Extraction 20

08 Investment Committee Narrative Builder 22

09 REIT Compliance Checklist Assistant 24

10 Pipeline and Email Triage with CRM Hygiene 26

GOING FURTHER

From Workflow to System 29

The Universal Guardrails 30

About the Practice and Your Next Step 31

FOREWORD

AI Has to Survive the Underwriting Model

Most real estate teams already have AI open in a browser tab. It sits right next to Argus and the rent roll, answering questions all day long. Useful? Sometimes. Trustworthy on a Monday morning, when the deal is real and the numbers have to hold? That is a different question.

I have spent more than 25 years in commercial real estate, on the research side, the finance side, and the ownership side. I have underwritten assets, forecast property taxes, and sat across from owners who wanted a straight answer. So here is mine. AI in real estate is not magic, and it is not snake oil. It is a tool that has to survive the underwriting model. If a workflow does not improve NOI, shorten a cycle, sharpen a decision, or lower risk, it does not belong in your budget. Full stop.

The gap I keep seeing is not a model gap. The models are good enough. The gap is the implementation layer, the quiet connective work that turns a clever answer into a workflow you can run the same way every time, with checks that catch the mistakes before they reach a memo or an investor.

This book is deliberately small in scope. We are going to build at the individual level. Ten everyday tasks, the kind that quietly eat your week. Lease abstraction. Deal screening. Variance narratives. Rent roll risk. Each one comes with steps, a prompt you can paste, and the part most guides skip, the guardrails. Where the model lies. What you must verify. What you should never hand to a machine without a person reading it first.

Will this give you the whole system? No, and I will not pretend it does. A portfolio-wide implementation layer, wired into your data room and your systems of record, governed and audited, is a much bigger build. That is the work my team does. But you cannot specify a system you have never felt. So feel it first. Build a few of these, see where the value actually sits, and you will know exactly what to ask for next.

The asset does not care how impressive the technology sounds. It cares whether the work got faster, cleaner, and more reliable. Let's start there.

MH

Mike Hammerslag

Real Estate Technology Advisor, Taazaa

Leads Taazaa's Real Estate and Finance practice. More than 25 years across CRE research, strategy, financial forecasting, property management, leasing, and ownership.

THE CORE IDEA

What the Implementation Layer Actually Is

Here is a pattern worth noticing. The teams getting real value from AI rarely have a better model than everyone else. They have a better setup around the model. That setup has a name, even if nobody says it out loud. We call it the implementation layer.

Picture three layers stacked on top of each other. The model sits at the bottom. The interface sits at the top. The implementation layer is the part in the middle that most people skip, and it is where the value lives.

TOP LAYER

The interface

Where you interact. A chat box, a button inside your CRM, an internal app. Easy to see, easy to overrate.

THE MIDDLE LAYER, WHERE ROI LIVES

The implementation layer

The connective work that turns a smart answer into a reliable workflow.

- Structured prompts
- Context & retrieval
- Output schemas
- Guardrails
- Human checkpoints
- Integration
- Monitoring

BOTTOM LAYER

The model

The reasoning engine (GPT, Claude, and the rest). Powerful, improving fast, and increasingly a commodity.

Think of it like a car. The model is the engine. The implementation layer is the drivetrain, the brakes, the dashboard, and the seatbelts. The interface is the steering wheel. A brilliant engine with no brakes is not a car you want anywhere near a live deal. The reason most AI pilots stall is not a weak engine. It is everything that was never built around it.

So what does that connective work look like, concretely? Every workflow in this book shares the same seven parts. Learn the shape once, and you will see it in all 10.

01 **Inputs.** The documents and data you feed in. A lease PDF, a T-12, a rent roll, an offering memo.

02 **Instructions (the prompt).** The role, the task, the rules, and the exact format you want back.

03 **Context and retrieval.** The reference material the model should ground in. Your definitions, your templates, your real comps.

04 **Output schema.** The shape of the answer. A table, a memo with set sections, fields that map to your systems.

05 **Guardrails.** The checks that catch errors before they spread. Source citations, confidence flags, math you verify.

06 **Human checkpoint.** Who reviews what, and signs, before any output counts.

07 **Integration and memory.** Where the output goes and how it gets reused. Optional when you work solo, essential at the system level.

At the individual level, you supply most of these by hand. You paste the inputs, you write the prompt, you verify the math, you do the review. That is fine. That is the point. Once you feel how much the structure matters, the case for building it into a real system makes itself.

BEFORE YOU START

How to Use This Book

Each workflow follows the same template, so you can skim to the part you need. The task. Where AI helps and where it breaks. What you need. The build steps. A prompt to paste. What good output looks like. The guardrails. The human checkpoint. And a short note on what the system-level version looks like.

What you need (for almost all 10)

- **A capable AI assistant.** Any current frontier chat model that accepts file uploads will do.
- **Your real documents.** The lease, the rent roll, the OM, the variance report. Bring the actual work.
- **A spreadsheet.** For checking math and holding structured output.
- **About 30 minutes per workflow.** Less, once you have tuned the prompt to your house style.

READ THIS FIRST: DATA AND DISCLOSURE SAFETY

- Do not paste confidential leases, tenant data, investor data, or **material non-public information** into consumer AI tools that may train on your inputs.
- Use an enterprise tier with data protection, or your firm's approved environment. When in doubt, ask compliance before you paste.
- For investor relations and compliance work, treat **every** output as a draft for review, never a final answer. We will repeat this where it matters.

One more thing on the guardrails. We have written most of them in a simple pattern: what to **verify**, what to **never** do, and what to **flag** for a person. The prompts are starting points, not scripture. Tune the wording, the fields, and the thresholds to match how your team already works. Then run the same version every time. Consistency is half the value.

PART TWO

The 10 Workflows

Each one is a real task you already do by hand. Build it once, run it the same way every time, and keep a person on the parts that matter. Start anywhere. Start with the one eating your week.

WORKFLOW 01

Lease Abstraction and Key-Term Extraction

Function: Asset Management | **Best for:** Lease admin, AM analysts | **Build:** 30 min

Reading a 60-page lease to pull the terms that actually matter: commencement, expiration, base rent and escalations, options, expense recoveries, renewal and termination rights, exclusives. By hand, it is slow, and it is easy to skim past the one clause that bites you later.

Where AI helps, and where it breaks

HELPS

- Fast first-pass extraction into a consistent structure
- Catching clauses you would otherwise skim
- Standardizing abstracts across a portfolio

BREAKS

- Dates and dollar figures (transcription slips)
- Legal interpretation and ambiguous language
- Scanned or poorly OCR'd documents

What you need

- The lease as clean text (if scanned, run OCR first and check the quality)
- An AI assistant that accepts file uploads
- Your standard abstract field list

Build it

1. Define your abstract fields once: commencement, expiration, base rent schedule, escalations, free rent, options, recovery type, and so on.
2. Convert the lease to clean text and confirm it is readable end to end.
3. Paste the prompt below with your field list, then upload the lease.
4. Require a clause reference and page number for every extracted field.
5. Spot-check the high-stakes fields (dates, rent, options) against the source.

The prompt

PROMPT: COPY, PASTE, AND ADD YOUR FIELDS

```
You are a commercial lease analyst. Extract the following fields from the attached lease into a
table: [paste your field list].

Rules:
• For every field, quote the exact clause and give the section number and page.
• If a field is not present, write "NOT FOUND". If it is ambiguous or hand-amended, write
"AMBIGUOUS, review".
• Never guess a date, dollar amount, or term. No inference dressed up as fact.

Output a table with columns: Field | Value | Source clause | Section & page | Confidence (High/Med/
Low). Below the table, list every item marked AMBIGUOUS or Low for legal review.
```

What good output looks like

A structured abstract table where each value carries a clause citation, a page, and a confidence flag, plus a short "flagged for legal" list at the bottom. If a value has no citation, treat it as not yet verified.

GUARDRAILS

- **Verify** every date and dollar amount against the cited clause. Recompute the rent schedule by hand.
- **Never** accept a number without a page citation, and never abstract from a draft. Use the signed, final lease.
- **Flag** amendments, side letters, and hand-amended language. They often override the base lease.

Human checkpoint: a lease admin or analyst confirms dates, rent, and options before the abstract enters your system of record.

LEVEL UP: THE SYSTEM VERSION

At the system level, this runs across your entire lease library with a retrieval layer, writes straight into your lease administration platform, and surfaces upcoming critical dates automatically, so nothing depends on someone remembering to look.

WORKFLOW 02

First-Pass Deal Screening Memo

Function: Acquisitions | **Best for:** Analysts, associates | **Build:** 30 min

A broker sends an offering memorandum or a teaser. Before anyone builds a model, you need a fast, structured read: what is it, what is the story, what are the obvious risks, and does it fit the box. Done well, this saves hours of underwriting on deals that were never going to clear.

Where AI helps, and where it breaks

HELPS

- Turning a 40-page OM into a one-page screen
- Surfacing missing information fast
- Drafting sharp first questions for the broker

BREAKS

- Takes the broker's numbers at face value
- Cannot underwrite or judge returns
- Misses what the OM does not say

What you need

- The OM or teaser as a file
- Your investment box, written down: geographies, product types, deal size, hold, and return thresholds

Build it

1. Write your investment box once and save it. You will reuse it on every deal.
2. Upload the OM and paste the prompt with your box.
3. Ask for a fit verdict: in box, out of box, or needs info, with reasons.
4. Ask for a "claims to verify" list of every figure the OM asserts that you would confirm independently.
5. Ask for five sharp questions for the broker.

The prompt

PROMPT: COPY, PASTE, AND ADD YOUR BOX

```
You are an acquisitions analyst at a REIT. Read the attached offering memorandum and produce a
screening memo with these sections: Snapshot, The Story, Fit vs. Our Box, Top Risks, Claims to
Verify, Questions for Broker.

Our box: [paste geographies, product types, deal size, hold, return thresholds].

Rules:
• Treat every figure in the OM as an unverified claim, not a fact. The OM is a sales document.
• Do not estimate returns or fill gaps with assumptions. Flag missing data instead.
• In "Claims to Verify", list each asserted number (rents, occupancy, T-12, market rent, exit
assumptions) that we must confirm.
```

What good output looks like

A one-page memo with a clear fit verdict, a ranked risk list, a verification checklist that calls out every load-bearing OM number, and five questions worth a broker's time. It tells you whether to spend the afternoon, not whether to buy.

GUARDRAILS

- **Verify** occupancy, the T-12, and the rent roll independently before the deal advances. The screen is a filter, not a decision.
- **Never** let a screening memo stand in for underwriting, and never trust an OM's exit cap or comp set on its own.
- **Flag** cherry-picked comps, aggressive growth assumptions, and anything the OM is conspicuously quiet about.

Human checkpoint: an associate validates the fit verdict and pressure-tests the risk list before it reaches the deal lead.

LEVEL UP: THE SYSTEM VERSION

The system version screens every inbound deal from your deal inbox the moment it lands, scores fit against your live criteria, drafts the broker questions, and logs it all to your pipeline, so nothing sits unread.

WORKFLOW 03

Budget vs. Actual Variance Narrative

Function: Asset Management, FP&A | **Best for:** AMs, property accounting | **Build:** 25 min

Every month, someone has to explain why each property's actuals diverged from budget. Pulling the variances is mechanical. Writing the narrative, line by line, is the slog that eats a day around close.

Where AI helps, and where it breaks

HELPS

- Drafting commentary straight from your numbers
- Grouping variances by driver and size
- Flagging the material movers quickly

BREAKS

- It does not know why a variance happened
- It will invent plausible causes if allowed
- It can miscount if totals are not clean

What you need

- Your variance table: line item, budget, actual, dollar variance, percent variance
- A materiality threshold so the model explains signal, not noise

Build it

1. Export the variance table to a clean, labeled format.
2. Set a materiality threshold (for example, variances above a set dollar amount or percent).
3. Paste the prompt and the table.
4. For each material variance, supply the known cause, or let the model offer a clearly labeled hypothesis.
5. Have the property manager confirm or correct every cause before it ships.

The prompt

PROMPT: COPY, PASTE, AND SET YOUR THRESHOLD

```
You are an asset manager writing monthly variance commentary. Using the attached variance table,
write commentary for every variance above [threshold].

Rules:
• Never state a cause as fact unless I provide it. If you infer a cause, label it "HYPOTHESIS,
confirm with PM".
• Reconcile to the totals I gave you. If a total does not tie, say so rather than smoothing it
over.

Output: a short narrative grouped into Revenue, Operating Expenses, and NOI, with a one-line
callout per material item.
```

What good output looks like

A clean, grouped narrative draft where confirmed causes and labeled hypotheses are visibly different, ready for the PM to verify in minutes rather than write from scratch.

GUARDRAILS

- **Verify** the actual cause of each material variance with the PM or controller. Reconcile totals to the source before sending.
- **Never** publish an AI-stated cause without human confirmation. This is the one that burns people.
- **Flag** confident-sounding explanations the model offers on its own. Confidence is not evidence.

Human checkpoint: the PM or controller confirms the cause of each material variance, and the AM signs off on the NOI commentary.

LEVEL UP: THE SYSTEM VERSION

The system version pulls variances directly from your accounting platform on close, drafts the commentary, and routes each material item to the right PM for confirmation, with a record of who confirmed what.

WORKFLOW 04

Rent Roll Risk and Rollover Analysis

Function: Asset Management | **Best for:** AM analysts | **Build:** 30 min

Reading a rent roll for what matters: tenant concentration, the lease expiration schedule, weighted average lease term, near-term exposure, and rents that sit above or below market. The insight is valuable. The math is where a careless run quietly goes wrong.

Where AI helps, and where it breaks

HELPS

- Summarizing a messy rent roll into a clear picture
- Building the rollover schedule by year
- Calculating concentration and weighted average lease term

BREAKS

- Non-standard or dirty rent roll formats
- Any calculation you do not verify
- Inconsistent dates and unit assumptions

What you need

- The rent roll as a clean table (CSV or spreadsheet is best)
- Your definitions: how you calculate weighted average lease term and what counts as near-term

Build it

1. Export the rent roll to a clean table: tenant, square feet, start, expiration, base rent, recovery type.
2. Give the model your definitions so it computes the way your team does.
3. Ask for concentration, a rollover schedule by year, weighted average lease term, and a near-term watch list.
4. Require it to show the calculation behind every number.
5. Recompute the top three figures yourself before trusting the rest.

The prompt

PROMPT: COPY, PASTE, AND ADD YOUR DEFINITIONS

```
You are an asset manager analyzing the attached rent roll. Produce:
1) Tenant concentration, top 5 by square feet and top 5 by base rent.
2) Lease expiration schedule by year, showing square feet and rent rolling each year.
3) Weighted average lease term, by square feet and by rent.
4) A 24-month rollover watch list.

Rules:
• Show the calculation for every number.
• Flag rows with missing or inconsistent data. Do not fill gaps or guess.
```

What good output looks like

A concentration table, a year-by-year rollover schedule, weighted average lease term shown two ways, a near-term watch list, and a separate list of data-quality flags. Every calculated figure shows its work.

GUARDRAILS

- **Verify** all totals and weighted average lease term against the source. Recompute the top concentrations by hand.
- **Never** trust calculations on dirty data without a check, and never let it silently fill a missing date or rent.
- **Flag** recovery types before drawing expense conclusions. Gross and net change the whole story.

Human checkpoint: an analyst recomputes weighted average lease term and the top concentrations, and confirms the watch list with the AM.

LEVEL UP: THE SYSTEM VERSION

The system version monitors rollover across the whole portfolio, alerts when concentration crosses a threshold, and ties upcoming expirations to your leasing pipeline before they become surprises.

WORKFLOW 05

Submarket Comp and Market Brief

Function: Acquisitions, Research | **Best for:** Analysts, researchers | **Build:** 35 min

Pulling together a quick market read for a submarket: supply, demand, rent trends, comps, and the narrative tying them together. The synthesis is genuinely useful. The sourcing is where this one gets dangerous, so read the guardrails twice.

Where AI helps, and where it breaks

HELPS

- Structuring a clean market brief
- Summarizing material you provide
- Drafting the narrative around your data

BREAKS

- This is the highest hallucination risk in the book
- It will invent buildings, rents, and dates that look real
- It cannot tell you what is current

Why this one is risky. Ask a model for "industrial comps in this submarket" with no sources, and it will happily produce a polished table of comps that do not exist. The output looks credible, which is exactly the problem. The fix is simple but strict: it synthesizes only what you give it.

What you need

- Your own source material: market reports, comp exports, broker data, your notes
- A hard rule in the prompt: use only the provided sources

Build it

1. Gather real sources yourself. The model does not get to bring its own.
2. Paste or upload only that sourced material.
3. Use the prompt with the rule that every claim cites a source, and gaps are marked.
4. Ask it to keep sourced facts and your team's interpretation visibly separate.
5. Verify every comp and every number against the original source.

The prompt

PROMPT: COPY, PASTE, AND ATTACH YOUR SOURCES

```
You are a real estate research analyst. Using ONLY the documents I provide, write a market brief
with these sections: Supply, Demand, Rent Trends, Notable Comps, Risks.

Rules:
• Use only the attached sources. Do not add facts, comps, rents, or statistics from general
knowledge.
• Cite the source for every factual claim. If something is not in the sources, write "NOT IN
SOURCES".
• Keep sourced facts and any interpretation in clearly separate sections. Label interpretation as
such.
```

What good output looks like

A brief where every number and comp traces to a document you supplied, gaps are stated plainly, and the team's read is fenced off from the facts. If a claim has no source, it does not belong in the memo.

GUARDRAILS

- **Verify** each comp and statistic against the original source, and date-check everything. Markets move fast.
- **Never** let it pull comps or stats from memory. Treat any uncited claim as false until proven.
- **Flag** the line between fact and interpretation. The two should never blur in a market brief.

Human checkpoint: a researcher verifies every comp and statistic against the source before the brief informs a memo or a price.

LEVEL UP: THE SYSTEM VERSION

The system version runs retrieval over your licensed data and internal research, so briefs are grounded in your real, current sources rather than the open internet, with citations baked into every line.

WORKFLOW 06

Investor Relations Q&A and Earnings Prep

Function: Investor Relations | **Best for:** IR teams, finance leads | **Build:** 30 min

Building an analyst Q&A bank and talking points ahead of an earnings call, drawn from your public filings. The drafting and the anticipation of tough questions are real time savers. The disclosure line is where this one demands discipline.

Where AI helps, and where it breaks

HELPS

- Structuring a clean Q&A bank
- Drafting plain answers from public material
- Anticipating the angles analysts take

BREAKS

- It cannot tell public from material non-public
- It will absorb confidential figures if you paste them
- It has no judgment on Regulation Fair Disclosure

DISCLOSURE CAUTION

Regulation Fair Disclosure (Reg FD) requires that material information reach all investors at once, not selectively. A model has no idea what your company has already disclosed. Feed it public filings only, and treat any answer that reaches past them as a legal question, not a drafting one.

What you need

- Public sources only: your latest 10-K and 10-Q, the prior earnings release, the public investor deck
- A hard rule in the prompt: public sources only, flag anything that reaches further

Build it

1. Gather public filings only. Nothing internal, nothing draft, nothing from the board packet.
2. Paste or upload that public material.
3. Prompt it to draft likely questions and answers grounded only in those documents.
4. Have it flag any answer that would require new disclosure rather than answering it.
5. Route the full bank to IR and legal before the call.

The prompt

PROMPT: COPY, PASTE, AND ATTACH YOUR PUBLIC FILINGS

```
You are an investor relations analyst. Using ONLY the public documents I provide, build an analyst
Q&A bank with likely questions and draft answers.

Rules:
• Use only the attached public sources. Do not add figures or claims from anywhere else.
• If an answer needs information not in the sources, write "REQUIRES DISCLOSURE REVIEW" instead of
answering.
• Do not speculate on guidance or forward numbers. Flag anything potentially material for legal
review.
```

What good output looks like

A Q&A bank sourced entirely to public filings, where disclosure-sensitive gaps are clearly flagged for IR and legal rather than quietly filled. The flags are the point. They route the hard calls to the people who should make them.

GUARDRAILS

- **Verify** every answer traces to an already-public document. If it is not public yet, it is not an answer.
- **Never** paste internal forecasts, draft results, or board materials into the tool.
- **Flag** any answer that would reveal something not yet public. Reg FD applies to all of it.

Human checkpoint: IR and legal review the full Q&A bank before the call. Nothing material reaches an investor without that review.

LEVEL UP: THE SYSTEM VERSION

The system version keeps a living Q&A knowledge base scoped to public disclosures only, with disclosure controls built in and an audit trail of who approved each answer and when.

WORKFLOW 07

ESG and Sustainability Data Extraction

Function: ESG, Operations | **Best for:** ESG leads, ops analysts | **Build:** 40 min

Pulling energy, water, and emissions data out of utility bills, meter exports, and property reports into one structured table for ESG reporting. The extraction saves hours of manual entry. The unit math is where quiet errors creep in.

Where AI helps, and where it breaks

HELPS

- Extracting numbers into a consistent schema
- Normalizing units across messy sources
- Flagging missing months and meters

BREAKS

- Silent unit errors (kWh versus MWh, therms versus kBtu)
- OCR misreads on scanned bills
- Inventing values for periods that are missing

Why this one needs care. ESG figures get audited and scrutinized by investors. A single wrong conversion factor does not stay small. It compounds across every property in the portfolio. Build the schema first, make every value cite its source, and reconcile a sample by hand.

What you need

- Source documents: utility bills, meter exports, property reports
- A fixed schema: property, meter or account, period, value, unit, source
- A normalization rule so units are consistent and conversions are shown

Build it

1. Define your schema first, before the model sees a single document.
2. Feed it the source documents.
3. Prompt it to extract into the schema and cite the source document and page for each value.
4. Require it to show any unit conversion and mark missing data as "MISSING", never zero.
5. Reconcile a sample of values against the original bills.

The prompt

PROMPT: COPY, PASTE, AND ATTACH YOUR SOURCE DOCUMENTS

```
You are a sustainability data analyst. Extract energy, water, and emissions data from the attached
documents into this exact schema: property, meter/account, period, value, unit, source doc + page.

Rules:
• Cite the source document and page for every value.
• Show your unit conversion whenever you normalize (for example, kWh to MWh).
• If a value is missing, write "MISSING". Never use 0 or an estimate. Do not infer or fill values.
```

What good output looks like

A clean table where every figure carries a source and page, conversions are shown in the open, and gaps are labeled honestly rather than guessed. When the auditor asks where a number came from, the answer is already in the row.

GUARDRAILS

- **Verify** a sample of values and every unit conversion against the original bills.
- **Never** let it estimate or backfill a missing period. A labeled gap beats a confident guess.
- **Flag** low-confidence reads on scanned documents for manual entry.

Human checkpoint: an ESG analyst reconciles a sample against source documents before any figure goes into a GRESB submission or an investor report.

LEVEL UP: THE SYSTEM VERSION

The system version ingests utility data on a schedule, applies your emission factors consistently across the portfolio, keeps a full source-to-number audit trail, and reconciles automatically against prior periods.

WORKFLOW 08

Investment Committee Narrative Builder

Function: Acquisitions | **Best for:** Deal leads, analysts | **Build:** 30 min

Turning your underwriting model and deal facts into a clear Investment Committee narrative: the story around the numbers, without touching the numbers themselves. The writing is the easy win. The discipline is keeping the model away from the math.

Where AI helps, and where it breaks

HELPS

- Drafting the narrative and the memo structure
- Articulating the business plan clearly
- Stating risks in plain language

BREAKS

- Recomputing or "tidying" your numbers
- Softening risks to flatter the deal
- Inventing support for the thesis

The rule that matters. The model tells the story. It does not do the math. Every number comes from your underwriting, reproduced exactly as you ran it. The moment a model starts recalculating returns to be "helpful", you have a memo that no longer matches your model.

What you need

- Your finalized underwriting outputs and deal facts
- Your IC memo structure
- A rule that numbers are reproduced, never recomputed

Build it

1. Finalize your numbers in your model first.
2. Paste your final numbers and deal facts into the prompt.
3. Prompt it to draft the narrative using those numbers verbatim.
4. Require a candid risk section that does not soften anything.
5. Keep the buy or pass recommendation as a human decision.

The prompt

PROMPT: COPY, PASTE, AND ADD YOUR DEAL FACTS

```
You are an investment analyst. Using the deal facts and underwriting outputs I provide, draft an
Investment Committee narrative with these sections: Opportunity, Business Plan, Returns, Key Risks,
Sensitivities.

Rules:
• Reproduce my numbers exactly. Never recalculate, round, or adjust them.
• Write a candid Key Risks section. Do not soften or bury anything.
• Do not state a buy or pass recommendation. Leave that to the committee.
```

What good output looks like

A sharp narrative where the numbers match your model to the dollar, risks are stated honestly, and the recommendation is left to the people accountable for it. The committee reads a clear story and still makes its own call.

GUARDRAILS

- **Verify** every figure in the narrative matches your underwriting exactly.
- **Never** let the model recompute returns or clean up your numbers.
- **Flag** any risk it downplays. The risk section exists to protect the committee, not the deal.

Human checkpoint: the deal lead confirms numbers tie to the model and owns the recommendation before the memo reaches the committee.

LEVEL UP: THE SYSTEM VERSION

The system version pulls numbers directly from your underwriting model, so the narrative and the math can never drift, with version control on every memo and your thesis logic preserved from deal to deal.

WORKFLOW 09

REIT Compliance Checklist Assistant

Function: Compliance, Tax | **Best for:** Compliance, finance ops | **Build:** 25 min

Organizing and tracking the recurring REIT compliance calendar, and drafting questions for your tax advisor. Note the verbs: organizing and drafting questions. This workflow never decides whether you comply with anything.

READ THIS BEFORE YOU BUILD

This is the most caution-heavy workflow in the book. A REIT's tax status depends on income tests, asset tests, and a distribution requirement, with real consequences if they are missed. A general-purpose model is not a tax authority and cannot tell you whether you pass any of them.

The tool organizes and reminds. It never concludes. Every actual judgment goes to your qualified tax advisor. No exceptions, no shortcuts.

Where AI helps, and where it breaks

HELPS

- Building a tracking checklist and calendar
- Organizing supporting documentation
- Drafting questions for your advisor

BREAKS

- It is not a tax authority
- It cannot tell you if you pass a test
- It must never quantify thresholds or give a determination

What you need

- Your compliance calendar and prior filings
- Your tax advisor in the loop from the start
- A hard rule: organize and track only, no conclusions, no thresholds

Build it

1. List the recurring requirements at a high level: income tests, asset tests, distribution requirement, filing dates.
2. Prompt it to build a tracking checklist and a list of questions for your advisor.
3. Explicitly forbid any threshold numbers or pass and fail language.
4. Add a "confirm with tax advisor" line to every item.
5. Route everything to your tax advisor for the actual judgments.

The prompt

PROMPT: COPY, PASTE, AND ADAPT TO YOUR CALENDAR

```
You are a compliance coordinator, NOT a tax advisor. Help me organize a REIT compliance tracking
checklist covering recurring requirements: income tests, asset tests, the distribution requirement,
and filing deadlines.

Rules:
• Organize and track only. Do NOT state any thresholds, percentages, or numeric tests.
• Do NOT assess whether anything passes or complies.
• For every item, draft a question I should ask my tax advisor rather than an answer. Add "Confirm
with tax advisor" to each line.
```

What good output looks like

A clean tracking checklist and a sharp set of questions for your tax advisor, with zero conclusions and zero numbers presented as tax guidance. It tells you what to track and what to ask. It never tells you the answer.

GUARDRAILS

- **Verify** nothing here is treated as tax advice. It is a tracking aid, full stop.
- **Never** accept a threshold, percentage, or pass and fail statement from the tool as authoritative.
- **Flag** every compliance judgment to a qualified tax advisor, always.

Human checkpoint: a qualified tax advisor makes every compliance determination. The tool never decides REIT status.

LEVEL UP: THE SYSTEM VERSION

The system version maintains the compliance calendar with reminders and a documentation trail, while keeping every actual determination with your tax and legal advisors, where it belongs.

WORKFLOW 10

Pipeline and Email Triage with CRM Hygiene

Function: Business Development, Acq Ops | **Best for:** BD, deal ops | **Build:** 30 min

Triaging inbound deal flow and broker emails into structured CRM fields, and drafting timely replies. The extraction and the first-draft reply save real time on a busy desk. The rule that keeps it safe: draft and stage, never send or auto-write.

Where AI helps, and where it breaks

HELPS

- Extracting deal details into CRM fields
- Prioritizing what came in overnight
- Drafting reply templates fast

BREAKS

- Sending anything on its own
- Writing to your CRM without review
- Misreading deal terms from a quick email

Why the approval gate matters. This workflow touches both your systems and your relationships. An auto-sent reply with a wrong number, or a CRM polluted with misread terms, costs more than it saves. Keep a person between the draft and the send, every time.

What you need

- The inbound emails
- Your CRM field structure
- An approval step before anything sends or saves

Build it

1. Paste the inbound broker email.
2. Prompt it to extract your key fields and draft a reply.
3. Keep every reply as a draft, never an auto-send.
4. Require your approval before any send or CRM write.
5. Have it flag anything ambiguous instead of guessing.

The prompt

PROMPT: COPY, PASTE, AND ADD YOUR CRM FIELDS

```
You are a deal operations assistant. From the broker email I paste, extract these fields: deal
name, market, asset type, size, asking price, broker/contact, and suggested next step. Then draft a
short, professional reply.

Rules:
• Extract only what is stated in the email. Mark anything unstated as "UNKNOWN".
• Produce the reply as a draft only. Do not send anything.
• Do not assume terms. Flag anything ambiguous for me to confirm.
```

What good output looks like

Clean structured fields ready to review, plus a tight draft reply you can approve or edit, with nothing sent or saved automatically. You stay fast on the desk and still keep your hand on every send.

GUARDRAILS

- **Verify** extracted deal terms against the email before they touch the CRM.
- **Never** let it send a reply or write to the CRM without your approval.
- **Flag** ambiguous or unstated terms instead of guessing them.

Human checkpoint: you approve every CRM write and every outbound reply. The tool drafts and stages. A person sends.

LEVEL UP: THE SYSTEM VERSION

The system version connects to your CRM with approval gates, dedupes and enriches inbound deal flow automatically, and keeps a record of what was suggested versus what you actually approved.

PART THREE

From Workflow to System

Ten workflows on your own desk will save you real hours. The bigger gains show up when these stop being personal habits and become how the whole organization works. That shift is the implementation layer.

THE BIGGER PICTURE

What Changes When It Becomes a System

Everything in this book works at the individual level, and that is the point. You can build any one of these today. But notice what each workflow leaned on: a vetted prompt, your own sourced documents, a guardrail you had to remember, a human approval step. At the individual level, all of that lives in your head and your habits. At the system level, it gets built in and runs the same way for everyone, every time.

Here is what actually changes in the jump from a personal workflow to an organizational system.

The shift, point by point

- **Shared, versioned prompts.** Instead of ten people each keeping their own copy, one vetted prompt becomes the standard, improved over time and run the same way across the team.
- **Retrieval over your real data.** The system reads your actual lease library, data room, and filings, so answers are grounded in your documents rather than the open internet. The hallucination risk from Workflow 5 mostly goes away.
- **Guardrails that do not depend on memory.** The checks in this book stop being something each person has to remember and become rules the system enforces on every run.
- **Connected to the tools you already run.** It plugs into the systems where work happens, from Yardi, MRI, RealPage, and Argus to HubSpot and your data warehouse, so output lands in the right place instead of a chat window.
- **Audit trails and access controls built in.** Role-based access, disclosure controls, and a record of who approved what. This is what compliance, IR, and tax need to sign off, and exactly what a chat box cannot give them.
- **Measured, not assumed.** You track cycle time, error rate, and hours returned, so the value is visible on a dashboard instead of living in anecdotes.

> **Where to start.** You do not need all of this at once. Start where the portfolio leaks the most time, margin, or confidence. That one becomes your first system, and the results pay for the next one. The workflows in this book are a good map of where the leaks usually are.

KEEP THIS PAGE

The Universal Guardrails

Every workflow in this book has its own guardrails, but the same handful of rules sits underneath all of them. If you remember nothing else, remember these. They are what separates AI that helps from AI that quietly creates risk. Print this page and keep it next to your screen.

01 **Cite the source**
Every factual claim should trace to a document you provided. No citation, no claim, no exceptions.

02 **Never paste confidential or non-public information**
If it is material non-public information, or not yours to share with an outside tool, it does not go in. Ever.

03 **Label inference**
Keep what the documents say separate from what the model thinks. In a memo or a brief, the two must never blur.

04 **Verify the math**
Treat every number as unverified until you check it against the source or your own model. Models are confident, not correct.

05 **Keep humans on decisions**
The model drafts, organizes, and extracts. People decide, approve, and send. Do not blur that line.

06 **Set a materiality threshold**
Spot-check low-stakes work. Verify high-stakes work line by line. Match the scrutiny to the consequence.

07 **Re-check dates and currency**
Markets and filings move fast. Confirm you are working from current information, not a stale snapshot.

08 **Use the final version**
Run the signed lease, the final model, the filed document. Never an early draft dressed up as the real thing.

09 **Approve before send or save**
Nothing reaches an investor, a broker, or your CRM without a human approval step in front of it.

10 **Escalate tax, legal, and disclosure**
When it touches REIT status, Reg FD, or legal exposure, route it to a qualified advisor. The model is not one.

THE PRACTICE

About the Practice and Your Next Step

The 10 workflows in this book are a working sample of the implementation layer, scaled down to what one person can build in an afternoon. They are deliberately individual. The real version is the organizational system underneath them: shared prompts, retrieval over your data, guardrails enforced on every run, and integration with the platforms you already use.

That is what our Real Estate and Finance practice builds. We help REITs and real estate firms move from clever one-off uses of AI to systems that survive the underwriting model, hold up under audit, and actually return hours to the team.

MH

Mike Hammerslag

Real Estate Technology Advisor, Taazaa

Leads Taazaa's Real Estate and Finance practice. More than 25 years across CRE research, strategy, financial forecasting, property management, leasing, and ownership. He works with REITs and owners to put AI where it improves NOI, cycle time, and decision confidence, and nowhere it does not.

See What Your Implementation Layer Could Look Like

If a few of these workflows hit close to home, that is usually a sign there is a system worth building. Let us walk through where AI would pay off first across your portfolio, grounded in your real tasks and your real data. No pitch, just a working session on your highest-impact use case.

Book a working session at taazaa.com

A note on sources and scope. This guide is educational and does not constitute tax, legal, accounting, or investment advice. REIT qualification rules carry real consequences. Always confirm with a qualified tax advisor. For primary references, see the REIT provisions of the Internal Revenue Code (IRC Section 856 and following) at **irs.gov**, Nareit at **nareit.com**, and GRESB at **gresb.com**. Product and platform names mentioned (Yardi, MRI, RealPage, Argus, HubSpot) are trademarks of their respective owners and are referenced for context only.