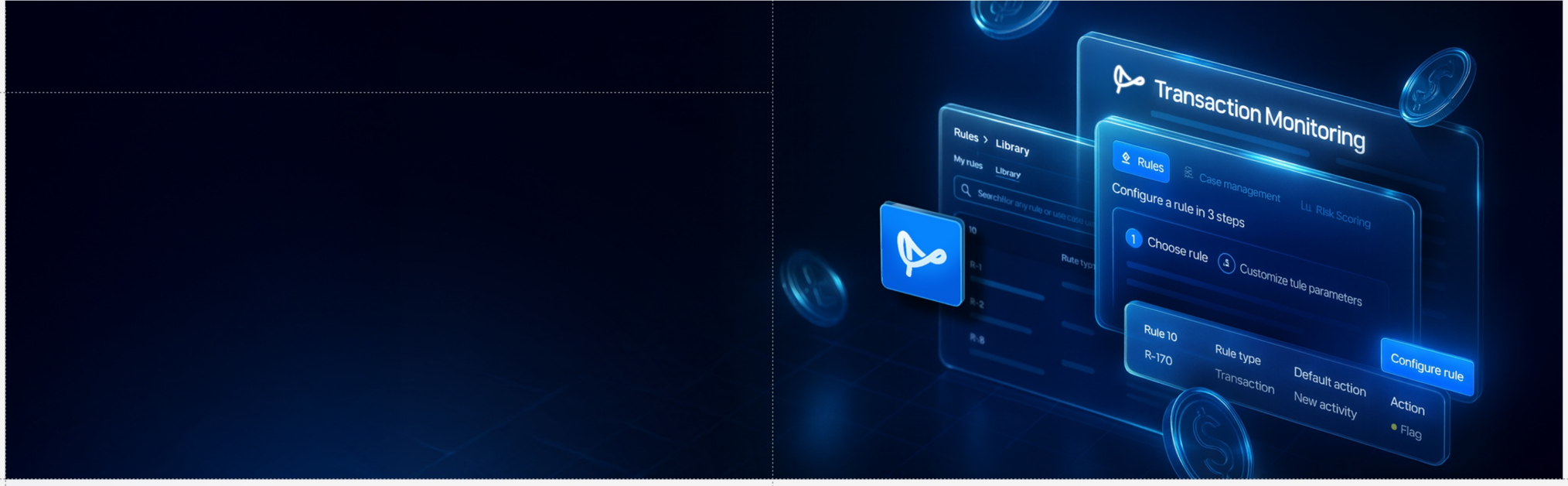
Designing a Real-World Transaction Monitoring Pilot

In 30 Days Without Breaking Production

Flagright



INTRODUCTION

A 30-day playbook to prove TM readiness

Launching a transaction monitoring (TM) pilot in 30 days might sound ambitious, but with the right scope and tools it's entirely achievable. This guide walks you through designing a one-month real-world TM pilot that delivers rapid insights without disrupting your live banking or fintech operations.

We'll cover how to tightly scope a 30-day pilot, leverage Flagright's latest features for speed and explainability, define success metrics, and decide a clear Go/No-Go at the end. The result is a practical playbook to validate an AI-driven TM solution in one month; reliably, transparently, and in line with regulatory expectations.

Why a 30-Day Pilot (Not 60-90) and How to Scope It

Traditional compliance pilots often span 2-3 months, but a well-planned 30-day pilot forces focus and yields faster feedback. The key is scoping the pilot tightly so it's realistic to execute in one month. Rather than trying to overhaul your entire monitoring program at once, zero in on a specific product line, risk typology, or segment of transactions that will serve as the pilot's testbed. For example, you might start with monitoring domestic wire transfers for structuring, or crypto wallet transactions for sanctions risks, not both at once. By limiting scope, you ensure the pilot can be set up and run end-to-end in 30 days, and any learnings can later expand to broader coverage.

Equally important is defining clear objectives. What do you hope to achieve in 30 days? It's not enough to say "see if AI works." Instead, frame a targeted hypothesis such as: "By augmenting our TM with Flagright's platform, we expect to reduce false positives by 50% while detecting at least 30% more truly suspicious cases, all in real time." Defining these goals up front anchors the pilot. It clarifies what success looks like (e.g. false positive reduction, improved detection, speed of alerts) and keeps the scope disciplined. Trying to tackle every possible issue in one short pilot is a recipe for floundering; a narrow focus delivers actionable results.

Real-time monitoring

Millions of events in 30 days

99.99% uptime

Sub-second latency, no blind spots

Rich data for precision

Measure false positives & rule accuracy

Don't worry that a 30-day pilot is too short; when done in real-time, a month of data can be incredibly rich. Flagright's platform operates in real-time with sub-second response times and 99.99% uptime, so you won't lose any observations to batching or downtime. In one month, a real-time system will process millions of events (if not more), plenty to assess performance. By the end of 30 days, you should know if the solution meets your objectives or if it needs tuning (or if it's a no-go). And if more time is truly needed for borderline cases, you can always extend the pilot, but the plan is to get answers in 30 days and avoid a drawn-out trial.

Ensuring No Impact on Production: Pilot in Parallel

A core principle is “do no harm” to your production environment. The pilot must run without breaking production, meaning no customer disruptions, no regulatory gaps, and no burden on your existing systems. The safest approach is to run the TM pilot in parallel to your current monitoring process rather than replacing it outright during testing. In practice, this means Flagright’s platform will ingest a copy of your transaction data (via API or data stream) and generate alerts independently, while your existing monitoring system continues as usual. The Flagright alerts are used for evaluation purposes in the pilot, not to actually block transactions or fulfill regulatory reporting (at least not yet during the test).

Running in parallel ensures that even if the new system misfires or produces unexpected results, it won’t interrupt normal operations. This addresses a common concern: will integrating a new real-time system slow down our transactions or overload our stack? With Flagright, that risk is minimal, the platform is built for high throughput (it’s been proven to handle thousands of transactions per second) and won’t introduce latency to customer transactions. Still, to be safe, the pilot can be set up in a shadow mode: the rules run and alerts trigger on the Flagright side, but they don’t automatically impact users.

Flagright even offers a “Shadow Rules” feature that lets you test new detection rules alongside live ones in a risk-free way. In a pilot context, you might run all your pilot rules as shadow rules, they will evaluate every transaction and log alerts, but not stop any transaction or create noise for customers. This way, you get full insights into how the system performs without any chance of false positives affecting users or legitimate payments.

Data integration for the pilot should also be handled carefully to avoid production risk. Fortunately, Flagright’s deployment is flexible, many teams integrate via an API gateway or event bus that simply forwards a copy of transactions to Flagright in real time. This read-only integration means the pilot can monitor real transactions as they occur, without the ability to interfere. If an API integration in production is sensitive, an alternative is feeding Flagright with a near-real-time data export (e.g. streaming transactions to a secure cloud bucket that Flagright pulls from). The goal is to start monitoring actual live data from day 1 of the pilot, but in a one-way flow. It doesn’t require deep core-banking changes; often it’s as simple as a minor config to forward JSON messages of transactions. By the end of the first week, you should have the pilot environment receiving production data continuously.

PILOT TIMELINE

30-Day Plan Breakdown

To manage the pilot efficiently, it helps to break the 30 days into phases with specific deliverables. Here's a high-level pilot plan structure you can follow:

WEEK 1

Setup and Baseline Configuration

In the first 7 days, complete the integration and initial setup. This includes hooking up the data feed (API or batch stream) and confirming Flagright is receiving all required data fields (transaction details, customer info, etc.). Spend a day or two configuring baseline rules: enable Flagright's out-of-the-box scenarios relevant to your focus area, and turn on anomaly detection rules to cover unknown risks. Also set up user access, case workflows, and QA checklists in the system. Essentially, by the end of week 1 the pilot environment should be live, ingesting data, and triggering alerts on default settings. Pro tip: identify a handful of historical suspicious cases and input them as test data (if possible) to verify the system would catch them, this builds confidence early.

WEEK 2

Run Pilot & Calibrate

In week 2, let the pilot run and start collecting results. Analysts should begin reviewing Flagright alerts as they come in (in parallel to handling production alerts in the old system, if any). Hold a mid-week check-in to gather observations: Are alert volumes roughly as expected? Any obvious false positives to tweak? Are analysts understanding the flag explanations? Use Flagright's real-time monitoring to adjust on the fly; for instance, if a particular rule is firing too often on benign events, refine its threshold or scope (and document that change with the required comment in the platform). Conversely, if something risky wasn't flagged, consider adding a new rule or risk factor to cover it. Thanks to version control and shadow mode, you can make these adjustments safely and see their effect quickly. By the end of week 2, you might run a mini "QA audit" on the first batch of alerts: have a QA reviewer check a sample of closed alerts for quality, using the checklist and pass/fail system. Any feedback from QA (e.g. analysts need to attach blockchain evidence for crypto alerts) can be fed into training or process updates in week 3.

WEEK 3

Ramp-Up and Ongoing Monitoring

The pilot really proves its mettle under sustained operation. By now, the system has likely seen most common scenarios, and some rarer events. Continue running at full data volume. If your business has weekly cycles (e.g. higher volumes on weekends), this week will show how the system handles peak periods. Monitor system performance metrics closely: throughput, latency, uptime. Flagright's dashboard or support can provide these stats to verify everything is running smoothly (expect near 100% uptime and ms-level processing times; in our pilots, there's been no slow-down to client systems). From a compliance perspective, this is a good time to measure detection outcomes: Did the system catch any incidents that would have been missed? How many alerts are high-quality vs. false alarms? Engage your investigators and ask if they trust the alerts, are there fewer "cry wolf" situations? You should see improvement given the advanced filtering and anomaly logic, but gather their qualitative feedback. You can also start compiling data for metrics (we'll detail in the next section). If needed, perform a simulation of a potential change now that you have two weeks of data (for example, test a stricter risk model vs. the current one to see if it would overload with alerts or not). This lets you consider recommendations for production: maybe the pilot is running relatively conservative settings and you find you could tighten rules further to catch even more without too many extra alerts.

WEEK 4

Evaluation and Documentation

In the final week, focus on capturing results and deciding next steps. Continue monitoring through day 30, but start analyzing against success criteria. This week is about measurement, documentation, and decision. Pull stats from Flagright: alert counts, QA pass/fail rates, average risk scores, etc. Also export logs of any changes, QA reports, and sample case narratives. Gather evidence that the pilot met (or didn't meet) objectives. Create a scorecard (see next section) summarizing outcomes vs. initial targets. By mid-week, convene key stakeholders (compliance, risk, IT, internal audit) to review findings. Discuss whether results are strong enough for production rollout or if extending the pilot is warranted. By Day 30, aim for a clear recommendation: Go live, No-go, or Extend. Document lessons learned — note what to do differently or any challenges (valuable when scaling up). End the pilot with a brief report or presentation that serves as both a thought leadership piece and implementation plan. If successful, this documentation bolsters the case for further investment; if not, it clarifies rationale and next steps (like trying another approach or more data).

By structuring the 30-day pilot in these phases, you ensure a logical progression and avoid last-minute scrambling. Each week has its focus, and you continuously improve as you go, which maximizes the short timeframe. Moreover, this structure signals to observers that the pilot was systematic and well-managed, reinforcing trust in the process and in Flagright's technology.

Leveraging Advanced Flagright Features in the Pilot

A major advantage of using Flagright for this pilot is the plethora of out-of-the-box features that accelerate setup and enhance reliability. In a 30-day timeframe, you want to avoid heavy custom coding or waiting on data science teams to develop models. Flagright comes with AI-native monitoring capabilities that you can turn on from day one, ensuring the pilot isn't a bare-bones test, but rather a showcase of a mature, robust system. Here's how you can utilize Flagright's latest features at each stage of the pilot:

Dynamic Rules & Anomaly Detection

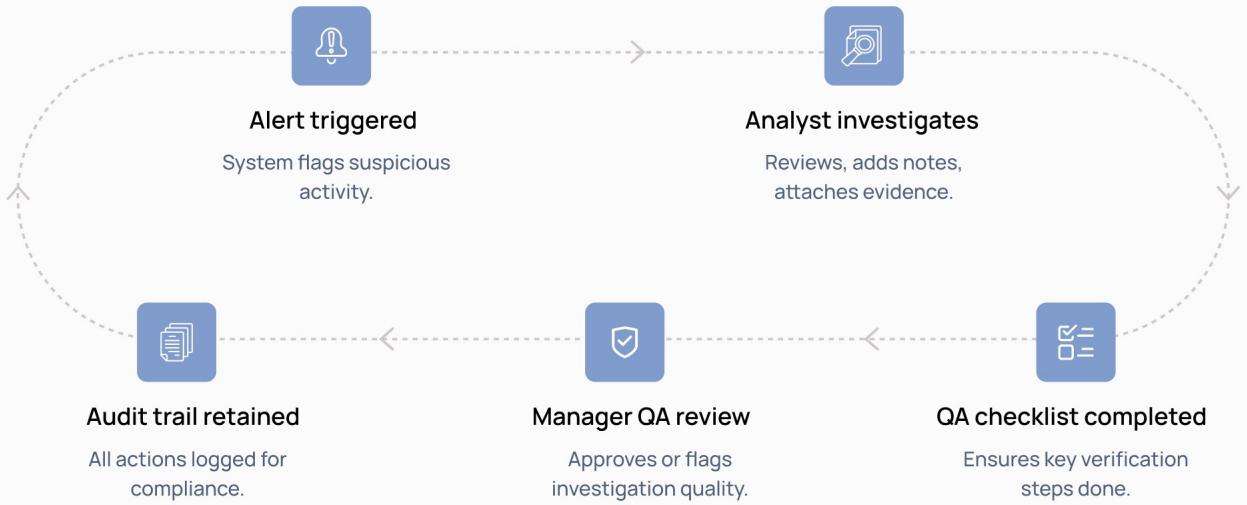
Static threshold rules often require fine-tuning (and can be too rigid or noisy). In this pilot, take advantage of Flagright's dynamic behavioral analytics, essentially anomaly detection rules that auto-calibrate to each customer's normal activity. For example, instead of a flat rule like "alert on transfers > $10k", you can enable rules that compare each user's transactions against their own historical average and standard deviation. If a usually low-volume customer suddenly sends five $5,000 transfers, the system will flag that spike as an anomaly, even if each transaction alone isn't huge. These dynamic rules greatly reduce false positives and catch emerging patterns faster. In the pilot, this means you spend less time fiddling with thresholds and more time seeing meaningful alerts. It's a quick win, just by enabling Flagright's pre-built anomaly detection policies, one bank saw a 93% reduction in false positives on a previously noisy scenario. Early in the 30-day run, you'll likely witness the difference: fewer trivial alerts and more focus on genuinely deviant behavior that might signal risk.

Real-Time Detection & Alerts

As the pilot runs, every transaction flows into Flagright's cloud in real time. The system's high-performance rule engine evaluates each event in milliseconds. This means by the time a transaction completes, any suspicious pattern is already flagged in the dashboard. Real-timeness is crucial to prove in a pilot, it's one of the big improvements we're doubling down on. You can demonstrate that alerts are popping up within seconds of risky activity, versus end-of-day or weekly reports in legacy setups. This immediacy not only helps catch fast-moving threats (imagine stopping a fraudster on their second try instead of their 50th), but it also shows stakeholders the potential to intervene sooner. Use the pilot to log the detection latency: e.g. "average time from transaction to alert: 5 seconds". If you have a baseline (like an older system that flagged things hours later), the contrast will be powerful.

Case Management, Investigation Workflow, and QA

Running a pilot isn't just about the tech catching alerts, it's also about how your team handles those alerts. Flagright includes a full case management system with built-in investigation workflows. During the pilot, every alert can be managed in Flagright: analysts can triage, add notes, attach evidence, and mark resolution. Critically, the platform has Quality Assurance (QA) tools baked in to maintain high investigation standards. You can configure investigation checklists that analysts must complete before closing an alert (for example, steps like "Verify customer KYC info" or "Review last 10 transactions"). These checklists ensure nothing falls through the cracks. Even better, a compliance manager can review closed alerts and mark a QA outcome (Pass/Fail) with comments, and those QA results are visible right in the case interface. For the pilot, this means you can actively measure the quality of alert handling. If an alert was closed improperly, the QA reviewer's note (e.g. "Fail, source of funds not actually verified") will be logged, and the analyst learns from it. This feedback loop is invaluable in a short pilot: by week 2 or 3 you can correct investigative approaches if needed. It also demonstrates to regulators and auditors (should they inquire) that your process has oversight. All QA reviews and checklist records are retained, so by pilot's end you have an audit-ready trail of every alert's journey and the rationale behind each decision.



Risk Scoring and Customer Profiles

Flagright doesn't just generate alerts – it continuously scores the risk of each customer and transaction. Throughout the 30 days, the system will produce a Customer Risk Assessment (CRA) for each user, which combines their inherent risk factors (e.g. country, KYC info) with their behavioral patterns (transaction trends). This ongoing risk scoring provides an "at-a-glance" indicator of which customers are trending high-risk as the pilot progresses. It's very much a human-in-the-loop design: your compliance officers can review a customer's profile in the dashboard and see the risk score, what contributed to it, and how it changed over time. For explainability, the platform surfaces the specific factors, e.g. "High risk because 3 risk indicators triggered: unusual transaction pattern, sanction-hit counterparty, use of mixing service". Those risk indicators are clear and traceable, so your team can understand why the system deems something risky. This directly addresses the "black box" worry that often accompanies AI. In fact, any time Flagright's AI flags an alert, it also provides context (which rule or anomaly triggered, which profile factor was out of bounds, etc.), making every alert explainable by design. Use this in the pilot to evaluate if analysts feel comfortable with the automated decisions, can they easily see why an alert was generated and does that rationale make sense? That feedback is key, because a system that catches everything but can't be explained to regulators or senior management won't fly. With Flagright, explainability is built-in via these risk factor breakdowns and rule annotations.

Screening Integration and Enhanced Profiles

Transaction monitoring rarely lives in isolation, often an alert will involve sanctions screening or watchlist hits. Flagright's platform provides integrated AML screening, and you can configure screening profiles during the pilot to tailor what lists are checked for different scenarios. For example, for low-risk customers you might only screen against major sanctions lists, but for high-risk customers or large transfers you screen against every global list and adverse media. The ability to create these nuanced screening profiles means the pilot alerts will be more precise and relevant. It's another way to reduce noise: you won't waste time on an alert just because a low-impact watchlist matched a name loosely, you choose what's important. Also, any screening hits that do occur are linked in the case, so an analyst can directly see "This alert also matched John Doe to a PEP list" and even fetch the profile details. For the pilot's sake, this shows how a unified platform streamlines investigations (no more jumping between a monitoring tool and a separate sanctions screening tool). It keeps the human investigator in the loop with all the info at their fingertips, which speeds up resolution while maintaining thoroughness.

Governance

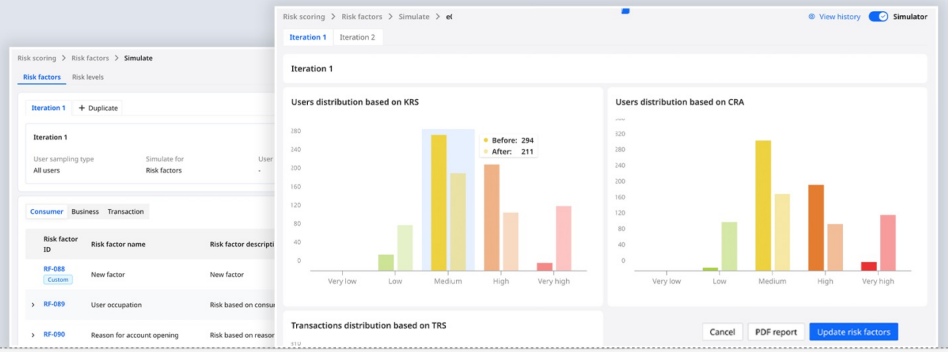
Version Control & Audit Trails: Compliance leaders and regulators will value how governance was maintained even in a pilot. Flagright has governance features to manage any rule changes or scoring model tweaks during the 30 days. Whenever a risk scoring parameter or rule condition is adjusted, the platform automatically creates a new version and logs who made the change with a comment. For example, if in week 2 you loosen a threshold to reduce alert volume, you'd enter a brief justification. This provides change traceability aligned with Model Risk Management (MRM) principles so anyone can review what changed and why. You can also require certain changes to go through an approval workflow. Any change affecting customer experience must be approved by a second manager, and the system enforces that maker-checker control. During the pilot, this might mean if an analyst adjusts a rule, a compliance lead must approve it before it applies. It might seem formal for a short pilot, but showing this capability is important because it proves the system can scale to production with strict oversight and complete audit logs. At the end of the pilot, you can pull a report of all configuration changes and approvals, proving the pilot was conducted with proper governance.

Rule Change Governance	Risk Score Evolution
Version control & approval workflow	Historical changes per customer
Maker-checker enforced	CSV/PDF export for validation
Complete audit logs	Audit-ready traceability

This is great material for internal audit or the model validation team to review, getting them comfortable early. Additionally, Flagright's risk scoring module allows you to download the full history of any customer's risk score changes over time. This means you can show, for example, how a user's risk rating went from Medium to High after certain alerts, with the exact factors listed. For audit readiness, you'll have a PDF/CSV of the entire risk evolution and can explain every uptick. Regulators often ask during exams, "Why did this customer's risk score change?", with Flagright you'll have that answer at your fingertips. Even in a short pilot, having this evidence builds confidence that the solution can meet regulatory scrutiny.

Simulation and “What-If” Testing

One cool thing you can do in a 30-day pilot is simulate potential changes without impacting the live pilot rules. Flagright includes a Risk Scoring Simulator tool where you can clone your current risk model, adjust weights or logic, and test it on a sample of historical data (up to 100k customers). Suppose midway through the pilot you notice too many alerts for a certain scenario, you might hypothesize that lowering the weight of that factor will help. Instead of guessing, you can run a simulation: create a variant model with that weight tweaked, and see how many users would drop from High risk to Medium, how many alerts would disappear, etc. The platform can output a report on the differences, which you could include in your pilot findings. This is a great way to show stakeholders that the team didn’t just passively run a vendor box, but actively experimented and optimized using data-driven methods (all within 30 days!). Any simulations or A/B tests should be documented as part of the pilot outcomes. It signals that when this goes live, the team will continuously refine the system rather than “set and forget”, a sign of a mature, principle-based compliance program.



Throughout these feature integrations, maintain effective communication with your team about the pilot. That means being expressive about the advantages (“We can instantly detect anomalies that used to slip through...”), technically confident (“The rule engine’s sub-second performance means no transaction goes unwatched.”), and value-oriented (“This will save us hours of manual review and let us focus on real risks”). However, avoid straying into hype; stakeholders appreciate that you focus on solving problems, not just touting shiny AI. In practice, the pilot updates you share internally might highlight concrete wins (e.g. “Already by week 2, the system flagged 3 likely SARs that our legacy rules missed, due to its dynamic profiling”) and back them up with data. Keep the tone confident and factual, showcasing the reliability and expertise built into the platform without sounding like a sales pitch. You’re demonstrating value, not just selling it.

Key Metrics and Success Criteria for the Pilot

From the outset, define the Key Performance Indicators (KPIs) that will judge the pilot's success. These should map directly to the pain points you're trying to solve and the goals you set. Based on internal priorities, we'll organize the metrics with the most impactful ones first (focusing strongly on real-time performance and productivity, per feedback). Below are essential metrics to track, and how to interpret them in the pilot context:

Detection Speed & System Reliability

Latency from transaction to alert should be near-instant. This is a top metric because real-timeness is a major value driver. For example, if previously an alert took hours or a day to generate (batch processing), now we expect it in seconds. Measure the average detection latency (and 90th percentile); e.g. "95% of alerts were triggered within 2 seconds of the triggering transaction." Also track system uptime and any processing lag. In a 30-day pilot, you ideally see 100% uptime with no missed data. Flagright's infrastructure is built for reliability (~99.99% uptime historically), so any downtime or delays would be notable (and likely a reason to pause).

A successful pilot will demonstrate that speed and stability are non-issues: the new system keeps up with production volumes without a hiccup. This metric reassures everyone that introducing real-time monitoring won't break anything, instead, it dramatically accelerates insight.

Investigator Productivity & Alert Quality

This metric measures how the pilot impacts the efficiency and effectiveness of the compliance team. It can be quantified through alerts handled per analyst per day and the SAR conversion rate (or true positive rate). If Flagright's AI prioritization works as intended, analysts should be able to review more alerts in the same time, with a higher proportion resulting in SARs. For example, if each analyst previously closed 10 alerts per day with 5% resulting in SARs, but during the pilot they close 15 alerts with 15% resulting in SARs, that's a significant improvement in both efficiency and quality. Track the SAR-to-alert ratio as a proxy for alert quality and collect analyst feedback to understand whether they're spending less time on irrelevant alerts and more on meaningful investigations. Productivity is not only about volume, but also about focusing on high-value cases.

If automation features like AI narratives, intelligent routing, or the QA module save time, quantify the impact, for instance, "~30 minutes saved per analyst per day in case reviews." The goal is to demonstrate that the pilot made the team more productive and improved outcomes, identifying more true cases with less manual effort. Flagright users have reported up to 87% reduction in manual monitoring workload through such automation, so evaluate whether your pilot shows similar results.

Detection Rate & Coverage (True Positives)

How many suspicious activities did the new system catch, especially those that your legacy process might miss? This is often measured in terms of false negatives avoided. If you had known cases or inserted test scenarios, check if Flagright detected them. More broadly, if during the pilot an incident occurred (e.g. a real fraud or AML issue), did the system flag it (and how quickly)? You may calculate a recall rate if you have a set of expected suspicious events: e.g. "The pilot detected 4 out of 5 of our test cases, 80% recall, versus 2/5 (40%) by the old system." Even without known test cases, you can reason qualitatively: the anomaly detection might surface a pattern that wasn't on your rule list at all, that counts as an expanded coverage win. Highlight any such "new risks identified".

For example, "Flagright's behavioral analytics alerted us to an unusual pattern of rapid transfers that we hadn't been monitoring for, which we deemed suspicious and escalated." That's evidence of broader risk coverage. The metric can be expressed as number of additional suspicious cases found, or percentage increase in SAR filings (if you confirm some through the pilot). Just be careful to differentiate true issues from false alarms, we only count it as improved detection if it really was something warranting investigation. If none of your existing cases slipped through, that itself is valuable to note: it means the system at least matches current coverage and likely provides other benefits (speed, efficiency). But ideally, a successful pilot will have at least one story of a bad actor caught or a scenario illuminated that was previously dark.

False Positive Reduction

One of the easiest ROI metrics is the reduction in false positives, i.e. alerts that turn out not to be suspicious. Flagright's approach of dynamic thresholds, screening profiles, and whitelisting should significantly cut down on noise. Quantify this by comparing the alert rate or alert-to-SAR ratio to your baseline. If historically you reviewed 100 alerts to find 1 case, and now it's 20 alerts to find 1 case, that's a 5x precision improvement (false positive rate plummeted). Or express it as "X% fewer alerts for the same outcome". In one example, unifying siloed monitoring on Flagright led to 93% fewer false positives, your results may vary, but even a 50% reduction is massive for workload. Make sure to account for any increase in true positives when calculating this; even if total alerts stayed the same, if more of them are real issues, that implicitly means fewer false ones.

This metric ties directly to efficiency and cost savings. It also has regulatory weight: fewer false positives means less "noise" and the ability to focus on real risk, which regulators encourage in a risk-based approach. If the pilot shows a notable drop in false alarms, emphasize that as a win for both compliance effectiveness and operational cost. Conversely, if false positives didn't drop, analyze why, maybe the scenarios were too sensitive or you need to adjust thresholds further. That insight would feed into either extending the pilot or tuning before go-live.

Rule Accuracy & Analyst Agreement

Here we consider agreement rate, how often did the system's alerts align with human judgments or with your existing system's alerts. We mention this metric last, because while it's useful to see consistency, a high disagreement isn't necessarily bad (the system could be finding things the old method missed, that's good, or it could be flagging junk the old method filtered out, that's bad). Still, track how many Flagright alerts overlapped with your legacy alerts, and how many were unique. Also track if analysts reviewing the pilot alerts generally concurred with the risk assessment. For instance, if the system flagged 100 alerts and your investigators agreed that 90 of them were indeed worth flagging (even if not all became SARs), that's a 90% analyst agreement. A few disagreements are normal and healthy, as they spur investigation into whether the system or the human missed something. During the pilot debrief, discuss any cases of disagreement in depth.

Often, you'll find either the AI spotted a subtle risk the human at first doubted (in which case the AI proved its value), or the AI flagged something benign that you need to refine (in which case you adjust the logic). Streamlining this metric's interpretation: if agreement rate is high, it gives confidence the AI and team are on the same page; if it's low, focus on why, it might highlight areas for improvement but doesn't outright mean failure. The goal is not 100% agreement (that could indicate the AI is just mimicking the old system), but a healthy alignment with some novel catches. For the decision makers, keep this explanation straightforward, e.g. "Out of X pilot alerts, the team deemed Y to represent true issues (Y/X = Z% alignment with risk experts' expectations). The remainder were false positives that we have identified for tuning." This shows you are critically evaluating the AI, not blindly trusting it.

Compliance & Audit Readiness

Though less of a numeric metric, it's crucial to assess whether the pilot met all compliance requirements and left an audit trail. This includes confirming no regulatory reports were missed (i.e. the pilot running in parallel didn't cause any SAR filing to slip through the cracks, it shouldn't, since your normal process was still active, but double-check). Also check if all the alerts and decisions in the pilot are well-documented in the system. Did the investigators fill the mandatory fields? Are QA notes present for all reviewed cases?

Essentially, quality assurance pass rate could be a metric, e.g. "100% of pilot alerts passed QA checks for proper procedure." If any failed, note the percentage and reasons. Another aspect is audit documentation: by the pilot's end, you should be able to export items like risk model history, CRA logs, and case records. The ease of producing this evidence can also be noted. For example, "All pilot alert decisions and rationale were documented within Flagright, and a full audit log was exported for our records, showing the system's readiness for regulatory review."

If any compliance gaps were found (say a certain data field needed for a regulator wasn't captured), that's critical to flag and would likely mean a "no-go" until resolved. But if everything is in order, this metric is more of a checkmark indicating the pilot didn't just catch more bad guys, it did so in a way that holds up to scrutiny. With these metrics defined, plan how you will collect and analyze them during and after the pilot. Many can be tracked via Flagright's dashboards or with simple queries/export: e.g. count of alerts, average risk score, number of QA fails, etc. Some metrics like agreement rate or SAR conversion will require human input (knowing which alerts led to SARs, etc.), so designate someone to compile that info. It's wise to create a simple spreadsheet or table as your scorecard of results. Let's illustrate what that might look like:

Metric	Baseline (BEFORE)	Pilot Result (30 DAYS)	Target/Benchmark	Met?
Alert Latency (avg time from transaction to alert)	~24 hours (batch process)	**5 seconds** (real-time)	**< 1 minute**	Yes (Exceeded)
Uptime/Reliability	N/A (legacy not real-time)	**99.99%** (no downtime)	**99%+**	Yes
Analyst Alerts per Day	~10 per analyst	**15 per analyst** (with AI help)	**> 12**	Yes
SAR Conversion Rate (true positives)	~5% of alerts → SARs	**15% of alerts → SARs**	**10% (2× increase)**	Yes
False Positive Rate	95% (only 5% alerts useful)	**85%** (15% alerts useful)	**< 50% false positive**	Yes (improved precision)
Notable Missed Cases	2 known cases missed in past	**0 missed by Flagright**	**0**	Yes
Analyst Agreement	-	**~90%** (most AI alerts concurred)	**~80%+ expected**	Yes
QA Pass Rate (procedural)	~90% (some checklist steps missed)	**95% (few QA fails, all minor)**	**100% (no critical fails)**	Almost (minor issues)

Presenting the results in a structured way like this makes it easy for stakeholders to see the value. In our example, most targets were met or exceeded (green checkmarks), and any near-misses are highlighted with an explanation (perhaps the QA process needs a bit more training). This balanced view builds credibility: you're transparent about the pilot's outcomes, not hiding any shortcomings, but clearly the positives dominate.

The Go/No-Go Decision Rubric

At the end of the 30 days, you need to make a recommendation: Go to Production, No-Go (do not proceed), or in some cases Extend/Tune Pilot for a bit longer. To ensure this decision is well-founded, use a Go/No-Go rubric that weighs the pilot results against must-have criteria. Essentially, these criteria come from your success metrics but focus on the minimum acceptable levels you require to be confident in a production rollout. Below is a sample Go/No-Go rubric with typical criteria:

Detection Efficacy

Go if the pilot caught all or most high-risk incidents (e.g. detected ≥ 90% of test scenarios or known suspicious behaviors). No-Go if it missed critical known issues or showed < 80% coverage of expected risks. Even one miss of a major typology might justify no-go unless clearly fixable, because it indicates a gap in the system's rules that could be exploited.

False Positive Management

Go if false positives dropped significantly or are at least manageable for the team (e.g. alert volume reduced by 50%, or precision improved to company's target level). No-Go if false positive rate remained as high as before or worse, since that would just carry over inefficiency to the new system. Gray area: if false positives didn't improve enough but other benefits are huge, you might lean "extend pilot" to tweak rules and try again.

Real-Time Performance & Reliability

Go if the system maintained real-time processing without outages (e.g. no noticeable downtime, and latency under X seconds). This is usually a go/no-go factor because any instability in production is unacceptable. No-Go if the pilot encountered crashes, significant delays, or data losses. Those would need to be resolved and perhaps re-piloted, as reliability is non-negotiable in monitoring critical financial activity.

Analyst Acceptance & Workflow Integration

Go if the compliance team finds the platform usable and an improvement. This can be gauged by feedback and that agreement rate we discussed, if analysts broadly agree with the alerts and find the interface and workflow (case management, QA) to add value, that's a positive sign. No-Go if analysts fundamentally distrust the alerts or struggled to use the system correctly (e.g. if QA found many mistakes or confusion). User buy-in is important; a tool that confuses or frustrates analysts could backfire in production, so any major usability issues should be ironed out first.

Governance & Audit Readiness

Go if all compliance checks are satisfied, audit logs complete, approvals in place for changes, and no regulatory requirements unmet. Basically, ensure the pilot demonstrated that using this system will keep you in line with regulators. No-Go if there were compliance gaps, for example, if some alerts didn't have complete audit trails or if a required report couldn't be generated. Those gaps need addressing before you trust the system fully.

ROI and Business Justification

This is a more strategic criterion. Go if the pilot showed clear value (e.g. X hours saved, Y% risk reduction, improved metrics across the board) that justifies the cost and effort of switching to the new system. No-Go if the improvements were marginal or the pilot raised more questions than it answered. While 30 days is short, you should have enough evidence of potential ROI. If it's not convincing, say the system worked but only modestly better than status quo, you might decide it's not worth proceeding (or maybe extend to see if improvements can be made).

Using such a rubric formalizes the decision. In practice, you might score each area (e.g. 1-5 scale) and see if the overall score passes a threshold for Go. Or simply discuss each point: if all the "Go" conditions are met, then green light; if any critical "No-Go" condition is present, then red light. In cases of mixed results, you might do a conditional go: proceed to production only after certain fixes or changes are implemented. For instance, you could conclude: "Go live with Flagright next quarter, contingent on adjusting the anomaly rule that caused a few false positives and providing additional analyst training on the new workflow."

One more aspect of Go/No-Go is planning the rollout if you choose "Go". Use pilot insights to decide how to deploy. Maybe you'll roll it out in phases (cover one product at a time) or big bang if confidence is high. Leverage what you learned: e.g. if the pilot showed strong results for domestic transfers, you might go live there first and pilot cross-border in a second phase. Also, address any outstanding issues as action items in the go-forward plan (e.g. "Before go-live, incorporate an additional rule for XYZ typology that was discovered.").

If the decision is "No-Go" (the pilot failed), it's equally important to document why and what comes next. Perhaps the data integration was incomplete, or the solution didn't meet a crucial need. These findings can either justify looking for a different solution or doing a second pilot after adjustments. A no-go after 30 days isn't a failure if it steers you in the right direction, it's better to know early than after a costly implementation. But ideally, with the careful planning and Flagright's capabilities, you'll reach a Go decision with confidence.

Calls to Action for Key Stakeholders

A successful TM pilot and subsequent rollout requires a concerted effort from multiple parts of the organization. Here are clear calls to action for each major stakeholder group to drive this initiative forward:

BSA/AML Officer & Compliance Team

Lead the charge on defining pilot success from a compliance standpoint. Set non-negotiables (e.g., no degradation in SAR quality or timeliness) and ensure the pilot design respects all regulatory requirements. Use your influence to secure internal buy-in and explain to leadership that the pilot strengthens the AML program, not weakens it. Be prepared to liaise with regulators: inform them of the pilot and later present the results.

Action: Draft a one-page brief for senior management and regulators on the pilot's purpose and safeguards, demonstrating that compliance is firmly in the driver's seat of this innovation.

Risk Management & Model Validation

Establish the governance framework around the AI model from day one. Document the pilot model as you would any high-risk model: define validation steps, monitoring metrics, and required documentation. Conduct an independent review of pilot outcomes, verify key metrics, and test sample scenarios to ensure the results are reliable. If the pilot moves to production, be prepared for full model validation.

Action: Create a model risk assessment plan outlining potential risks (e.g., explainability, data drift) and how they will be tested during the pilot to provide assurance to oversight committees.

IT & Product Teams

Enable the pilot technically and design for scale. In the pilot, your mission is to provide a stable, secure environment: set up the data feeds, sandbox, and integration points with minimal disruption to current systems. Ensure data security is paramount, coordinate with InfoSec to vet the AI platform (cloud security, data encryption, access controls). Looking ahead, start planning how this will integrate with your core systems if it goes live (case management systems, alert generation systems, etc.). Perhaps begin laying the API groundwork or selecting middleware to make the AI outputs feed seamlessly into analyst workflows.

Action: Draft an integration blueprint that outlines how the AI solution will connect to your transaction monitoring system and case management in production. Even if full integration is phase 2, having this blueprint ensures you can move quickly after a successful pilot and identifies any technical constraints or resources needed.

Operations & Analytics (AML Investigators)

Be the champions and the critics during the pilot. Embrace the opportunity to test a tool that could alleviate your pain points. Provide candid feedback on the AI's recommendations, are they useful, accurate, explainable? Identify where it helps and where it doesn't. Your perspective will shape model adjustments and the ultimate decision. Post-pilot, if moving forward, help update SOPs and mentor peers in using the new system.

Action: Document 2-3 case studies from the pilot, for instance, an alert the AI handled exceptionally well, and one where it stumbled but a human caught it. These real examples will be powerful when communicating the pilot's value (or limitations) to senior management and across the team.

Procurement & Vendor Management

Manage the external partnerships and ensure ROI justification. If this pilot involves an external vendor, you likely facilitated the initial agreements. Make sure the pilot agreement covers data usage, IP, and exit clauses (in case of no-go, data should be destroyed or returned). As results come in, work on the business case for full deployment: quantify the benefits (time saved, reduction in workload) in dollar terms versus the projected cost of the solution. Begin negotiating a scalable contract that can kick in if the pilot is successful; this might include volume-based pricing, support commitments, etc. Also, vet the vendor's compliance with relevant regs (are they SOC 2 compliant? Do they meet GDPR/CCPA if data's involved?).

Action: prepare a vendor due diligence report and ROI projection. This report should summarize how the vendor meets security/compliance standards and project the annual savings/benefits vs. cost if you deploy fully (e.g., “we estimate $X saved in analyst hours, avoidance of Y false alerts, etc., against a cost of $Z”). Having this ready will expedite approvals from finance and procurement committees post-pilot.

Senior Management & Board

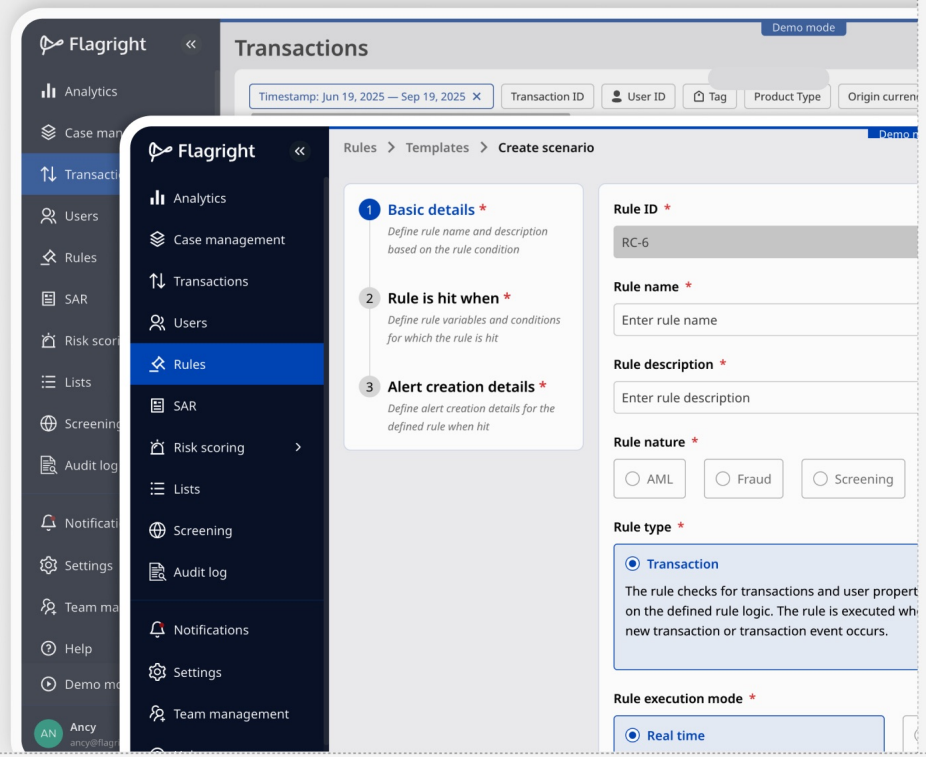
(If you are in this group or need to influence them) Support and oversight are key contributions here. Endorse the pilot as part of the institution’s innovation strategy – allocate the necessary budget and resources, and clear any organizational roadblocks. Ask the tough questions at the end: did it work, and is it worth scaling? Also consider the broader strategy: if successful, how will this AI capability give us a competitive edge or improve compliance assurance? Ensure that scaling plans align with the company’s risk appetite and strategic goals.

Action item for leadership: Establish a clear decision mandate for the pilot outcome. For example, resolve that “if the pilot meets predefined criteria, we empower the team to proceed with implementation, and if not, we will pivot or stop”. This sets the tone that the institution is committed to evidence-based decision making and avoids indecision after the pilot (which can demoralize teams).

By following these calls to action, each stakeholder can confidently play their part in the pilot’s success. The end result is a coordinated effort where everyone, from the compliance officer to the IT engineer to the procurement manager, is aligned on the mission: modernizing our transaction monitoring with AI, in a way that is safe, effective, and regulator-ready.

CONCLUSION

From Pilot to Production (and Beyond)



In just 30 days, you can go from uncertainty about a new transaction monitoring system to hard evidence of its impact. By explicitly limiting scope, running the pilot in parallel to avoid production risk, and leveraging Flagright's expressiveness (dynamic rules, integrated screening, real-time scoring, QA workflows, etc.), a one-month pilot can yield results that might take others 90 days to gather. The keys are real-timeness, reliability, and explainability, showing that the system works fast, stays dependable, and provides transparent reasoning for its alerts. We doubled down on those in our approach, and the payoff is a pilot that not only uncovers more risks and efficiencies, but does so in a manner that compliance officers, IT, and regulators can all get comfortable with.

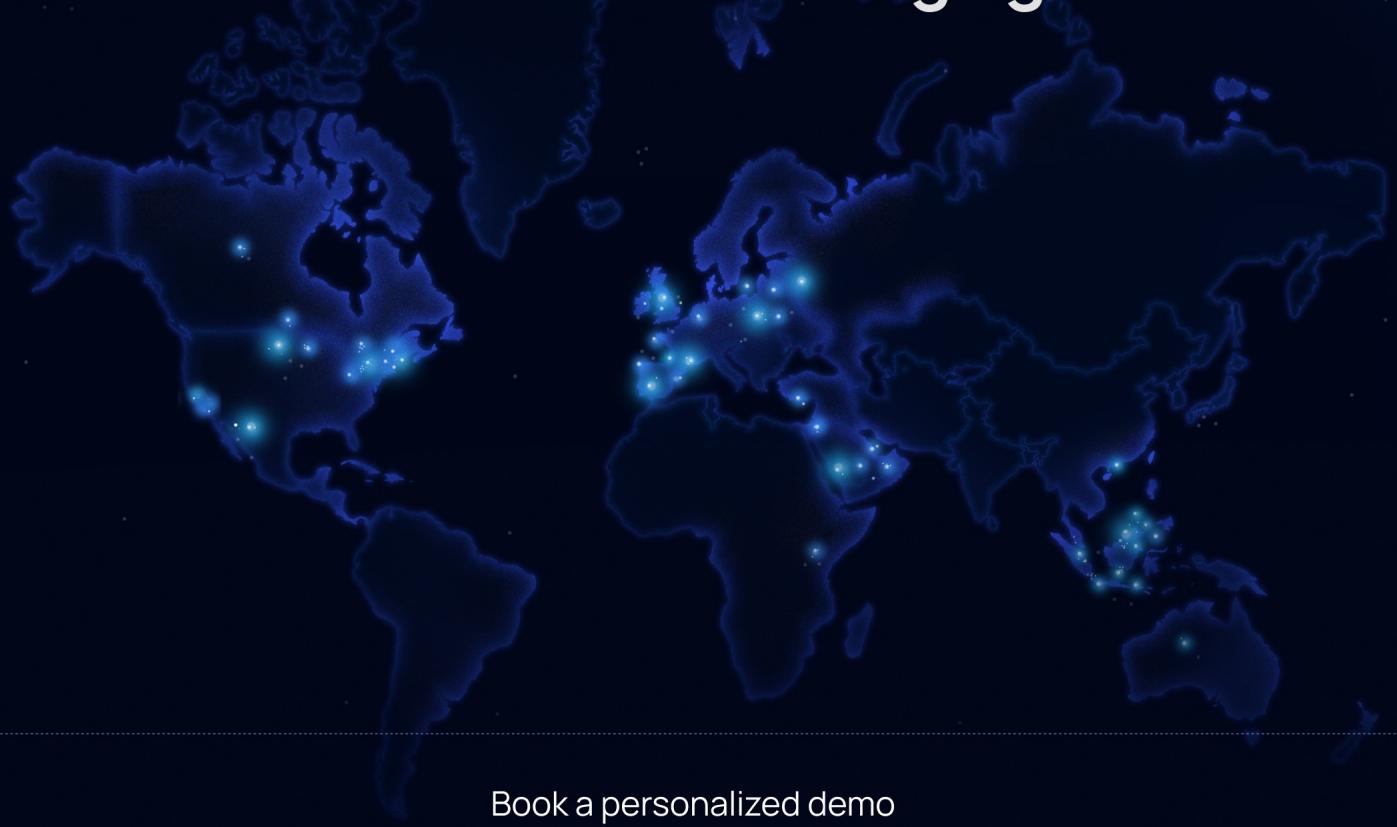
This pilot serves as both a thought leadership proof ("look what we accomplished in a month with cutting-edge AI monitoring") and a practical playbook for full deployment. If the outcome is a Go, you've essentially drafted the blueprint for implementation: you know which rules to use, which metrics to watch, how to train the team, and what value to expect. Even the documentation and scorecards you produced can roll directly into your production monitoring governance materials. You've also likely fostered buy-in across the organization by involving stakeholders early and often, they've seen the data and been part of the decision, so moving to production will feel like a natural next step, not a leap into the unknown.

Moving forward, maintain the momentum. The pilot is just Day 0 of continuous improvement. As you operationalize Flagright, keep iterating on scenarios (the platform makes it easy with no-code updates and shadow testing of new rules). Set up ongoing QA and periodic model reviews leveraging those version histories. And perhaps most importantly, capitalize on the real-time insights, use the fact that you can detect and act on risks immediately as a competitive and strategic advantage. For instance, if fraud patterns shift, you can respond within days (or hours) by tweaking a rule, rather than waiting for quarterly model recalibrations.

In summary, designing a 30-day TM pilot without breaking production is absolutely within reach. It requires focus, the right technology partner, and a clear eye on success metrics. By following this guide and utilizing Flagright's modern AML platform, you can achieve in one month what traditionally might take a whole quarter, all while keeping your business safe throughout. Whether you're a fast-moving fintech or a bank looking to modernize, this approach lets you test fast, learn fast, and trust the results. If the pilot meets its goals, you'll be ready to flip the switch to a smarter, faster transaction monitoring program that's battle-tested in your own environment. And if more tweaking is needed, you'll know exactly where to concentrate, with minimal time lost. Either way, you emerge with deeper insight into your compliance operations and how AI can enhance them in the real world. That knowledge is the true ROI of a pilot, and in today's ever-evolving financial crime landscape, it's worth its weight in gold.

Flagright

See why financial institutions around the world trust Flagright



Book a personalized demo

flagright.com/contact