THE FDE CUSTOMER

DISCOVERY QUESTION BANK

65 Questions to Uncover Real Customer Problems Before You Build

How to Use This Question Bank

Forward Deployed Engineers spend a large part of their week with customers before any code is written. The quality of what you build depends almost entirely on the quality of what you learn in those early conversations. A solution built on a misunderstood problem will be technically correct and practically useless.

This question bank gives you a structured set of questions to run discovery calls with confidence. It is not a script to read word for word. Treat it as a menu: pick the questions that fit your customer, your project, and the stage you are in.

When to Use It

- **Kickoff calls:** Use Sections A, B, and C to understand why the project exists, how work gets done today, and who is involved.
- **Scoping workshops:** Use Sections D, E, F, and G to map data, systems, AI requirements, and constraints in detail.

- **Pre-build validation:** Use Section H to lock down success metrics before development starts, so everyone agrees on what "done" means.
- **Mid-project check-ins:** Revisit Sections B and H to confirm the workflow hasn't changed and the targets are still right.

How to Adapt the Questions

The questions are written to work across industries, but small adjustments make them land better. Replace generic terms with the customer's own vocabulary. If a hospital calls its records system "the EMR," ask about "the EMR," not "your data source."

- **Healthcare:** Spend more time on Sections D and G. Patient data, consent, and HIPAA or DPDP requirements shape every design decision.
- **Retail and e-commerce:** Focus on Sections B and E. Data is usually spread across POS, inventory, and e-commerce platforms that don't talk to each other.
- **Legal and financial services:** Go deep on Section F. Accuracy thresholds, citations, and what the AI must never do are central.

- **Internal tools and knowledge assistants:** Prioritize Sections C and F. Adoption depends on users trusting the answers.

Question Priority Legend

Every question is tagged so you can decide quickly what to cover when time is short.

Tag	Meaning	When to Use
MUST-ASK	Critical to scoping. Skipping it creates a real risk of building the wrong thing.	Cover in every discovery engagement, ideally in the first or second call.
SITUATIONAL	Important depending on the industry, project size, or solution type.	Use when the topic is relevant to this customer or project.
FOLLOW-UP	Deepens an earlier answer or moves the engagement to a next step.	Use after a must-ask question opens a thread worth pulling.

Three Rules for Every Discovery Call

1. **Listen more than you talk.** Aim for the customer speaking 70% of the time. Your job is to learn, not to pitch a solution in the first ten minutes.
2. **Ask "why" at least twice.** The first answer is usually the symptom. "We need a chatbot" becomes "Agents spend too long finding policy answers" becomes "Policy documents are scattered across five systems." The third answer is what you build for.
3. **Confirm what you heard.** Summarize before moving on: "So the main delay is the manual reconciliation every Monday, is that right?" Customers will often correct you, and that correction is valuable.

> **Pro tip**
>
> Send the customer a short agenda 24 hours before the call that lists the topics, not the questions. It helps them bring the right people and data, and it signals that you are there to understand, not to sell.

The 45-Minute Discovery Call Framework

A discovery call without structure tends to drift. The customer talks about what is top of mind, and you leave with half the information you need. This framework divides a 45-minute call into five stages, each with a clear objective and a set of question sections to draw from.

STAGE 1	STAGE 2	STAGE 3	STAGE 4	STAGE 5
Context 5 min	**Workflow** 15 min	**Data & Systems** 10 min	**Success Criteria** 10 min	**Next Steps** 5 min
Sections A, C	**Section B**	**Sections D, E, F, G**	**Section H**	**Section I**

Stage 1: Context (5 minutes)

Open by understanding why this conversation is happening now and who is in the room. Introduce yourself briefly, confirm the time available, and ask what prompted the project. Two or three questions from Section A are enough here.

- Objective: understand the trigger, the sponsor, and the business goal.

- Example opener: "Before we get into details, what made this a priority for your team this quarter?"

Stage 2: Workflow (15 minutes)

This is the most important stage, so give it the most time. Ask the customer to walk you through the process as it happens today, using a real recent example rather than a description of how it should work. Draw the steps as they talk, and note every handoff, tool, and workaround.

- Objective: map the current process and find the true bottleneck.
- Example: "Can you take me through the last invoice dispute your team handled, from the moment it came in?"

Stage 3: Data & Systems (10 minutes)

Once you understand the workflow, trace where the information comes from and where it goes. Identify systems, data owners, access paths, and any compliance or security limits. If an IT contact isn't on the call, note it as an action item.

- Objective: build a rough map of data sources, integrations, and constraints.

- Example: "When the agent checks the order status, which system are they looking at, and does it have an API?"

Stage 4: Success Criteria (10 minutes)

Agree on how success will be measured before anything is built. Ask for today's baseline and a target number. A project without agreed metrics will be judged on opinion, and opinions change.

- Objective: define a measurable outcome and who will evaluate it.
- Example: "If we cut report preparation from six hours to one, would that be a clear win for you?"

Stage 5: Next Steps (5 minutes)

Close with clear actions, owners, and dates. Summarize the three most important things you learned and ask the customer to correct anything you misunderstood.

- Objective: leave with sample data requests, follow-up meetings, and named owners.
- Example: "I'll send a summary by tomorrow. Could your IT lead share the API documentation by Friday?"

> **When you only get 20 minutes**
>
> Cover only the must-ask questions from Sections A, B, and H. These tell you why the project matters, what the real problem is, and how success will be judged. Schedule a separate technical session for Sections D through G with the customer's IT team.

The Question Bank

The 65 questions below are grouped into nine sections that follow the natural order of a discovery engagement. Each section explains why it matters, shows an example from a real-world style scenario, and lists the questions with what a good answer sounds like and the warning signs to watch for.

Section A: Business Context & Goals

Every technical project exists to serve a business goal. If you don't know what that goal is, you can't make good trade-offs later, such as choosing

between speed and accuracy or deciding which features to cut when time is short.

These questions help you understand why the project exists, who cares about it, and how urgent it is. They also reveal whether the project has real executive backing or is an experiment that may lose priority in a month.

> **Example in practice**
>
> A logistics company asked an FDE to build "an AI tool for the operations team." When asked what triggered the project, the operations head explained that a major client had threatened to leave after three late shipment notifications. The real goal was not a general AI tool but proactive delay alerts for top accounts. That single answer narrowed the scope from months to weeks.

How to ask these well

- Start broad and let the customer describe the goal in their own words before you suggest anything.

- Listen for numbers, deadlines, and named people. These signal real commitment.
- If the answer is vague, ask for a recent example: "Can you tell me about a time this problem cost you something?"

Question	What You're Listening For	Red Flag
A1 MUST-ASK **What triggered this project now?**	A specific event: a lost client, an audit finding, a cost spike, or a leadership mandate with a date.	"Everyone is doing AI, so we should too."
A2 MUST-ASK **What business outcome would make this project a success?**	An outcome tied to revenue, cost, time saved, risk reduced, or customer satisfaction.	Goals like "be more innovative" with no way to measure them.
A3 MUST-ASK **What happens if this problem isn't solved in the next six months?**	A clear cost of inaction, such as lost revenue, compliance penalties, or staff burnout.	"Nothing much would change." Urgency is low and priority may slip.

A4 SITUATIONAL **Have you tried solving this before? What happened?**	Past attempts, why they stalled, and what was learned.	Several failed attempts blamed entirely on vendors, with no internal lessons.
A5 MUST-ASK **Who is sponsoring this project, and how involved are they?**	A named leader with budget authority who attends key meetings.	No sponsor, or a sponsor who is never available.
A6 SITUATIONAL **How does this project fit into your wider priorities this year?**	Alignment with a stated company strategy or annual objective.	The project competes with larger initiatives for the same people.
A7 SITUATIONAL **What budget or resource commitment has been approved?**	An approved range, plus named internal people who will support the work.	"We'll find budget after the pilot proves itself."
A8 FOLLOW-UP **If you could fix only one part of this problem, which would it be?**	A clear single priority that tells you where to start.	"All of it, equally." Priorities haven't been agreed internally.

Section B: Current Workflow & Pain Points

This is the section where most real problems are found. Customers usually describe the solution they want, not the problem they have. By asking how work actually gets done today, step by step, you uncover the bottlenecks, workarounds, and hidden manual tasks that the solution must address.

Always ask for a real, recent example rather than a general description. People describe processes as they are supposed to work. Recent examples reveal how they actually work.

> **Example in practice**
>
> A bank's compliance team requested "a faster document search tool." When the FDE asked them to walk through the last regulatory query they handled, it became clear that search took only 10 minutes. The remaining three hours went into copying findings into a spreadsheet, reformatting them, and emailing three reviewers for sign-off. The highest-value solution was automated report drafting and a review workflow, not better search.

How to ask these well

- Ask for the last real instance of the task, not the typical one.
- Sketch the workflow as they talk and show it back to them for correction.
- Pay close attention to every mention of spreadsheets, email, copy-paste, or "we check it manually." These are automation opportunities.

Question	What You're Listening For	Red Flag
B1 MUST-ASK **Walk me through the last time this task was done, step by step.**	Specific steps, tools, people, and handoffs in the order they actually happened.	A description of the official process that no one actually follows.
B2 MUST-ASK **How long does the process take from start to finish?**	A baseline time you can use later to measure improvement.	No one knows, and no one has ever measured it.
B3 MUST-ASK	Shadow processes that fill gaps in current systems.	Denial of workarounds while users clearly depend on them.

Where do people rely on spreadsheets, emails, or manual workarounds?		
B4 MUST-ASK **Which step causes the most delays or errors?**	A consistent answer across people, ideally supported by data.	Every stakeholder names a different step, with no data to settle it.
B5 SITUATIONAL **How often does this task happen?**	Volume per day, week, or month to estimate impact and size the solution.	Volume is so low that the build cost can't be justified.
B6 SITUATIONAL **What happens when something goes wrong in this process?**	How exceptions are handled, escalated, and resolved.	No defined exception process, so the edge cases are undocumented.
B7 MUST-ASK **Which parts of the process need human judgment?**	Clear boundaries for where AI can assist and where people must decide.	An expectation that AI will replace judgment in high-risk decisions.
B8 FOLLOW-UP **Can I observe someone doing this task, or see a screen recording?**	Willingness to give you access to real users and real screens.	Access to end users is refused or repeatedly delayed.

B9 SITUATIONAL **What has changed in this workflow in the last year?**	Stability of the process and any recent system or policy changes.	The process changes every month, so a build may be outdated at launch.
B10 FOLLOW-UP **If this process ran perfectly, what would it look like?**	A target state you can design toward and validate against.	The ideal described solves a different problem than the one discussed.

Section C: Users & Stakeholders

A solution that works but isn't adopted has failed. These questions identify who will use the solution, who approves it, who is affected indirectly, and who might resist it. Each group has different needs, and missing one can stall a deployment at the last minute.

Pay special attention to the gap between the buyer and the user. The person who asked for the project is often not the person who will use it every day.

> **Example in practice**

> An FDE built an AI assistant for a hospital's admin team based on requirements from the IT director. At rollout, the front-desk staff refused to use it because it required three extra clicks per patient compared with their existing process. Talking to two front-desk staff during discovery would have surfaced this in fifteen minutes.

How to ask these well

- Always ask to speak with at least one end user, not only managers.
- Map stakeholders into three groups: users, approvers, and affected teams.
- End each call by asking who else you should speak with.

Question	What You're Listening For	Red Flag
C1 MUST-ASK	Named user groups, their roles, and their current tools.	Users are described only in general terms, and no one has asked them.

Who will use this solution daily, and how comfortable are they with technology?		
C2 MUST-ASK **Who approves the output or signs off on decisions?**	Named approvers and the review process they follow.	Approval authority is unclear or shared by a committee that rarely meets.
C3 SITUATIONAL **Which teams are affected even if they won't use the tool directly?**	Downstream teams such as IT, legal, finance, or customer support.	Teams like IT or legal are unaware the project exists.
C4 SITUATIONAL **Who is most likely to resist this change, and why?**	Honest concerns such as job security, workload, or trust in AI.	"Everyone is on board." Resistance often surfaces late.
C5 FOLLOW-UP **Who else should I talk to for a complete picture?**	Names and roles, with an offer to introduce you.	Reluctance to involve anyone beyond the immediate contact.
C6 SITUATIONAL **How will users be trained and supported after launch?**	A plan for training, internal champions, and a support channel.	No plan, with an assumption that users will figure it out.

Section D: Data Sources & Quality

AI and automation solutions are only as good as the data behind them. Many FDE projects are delayed not by engineering problems but by data that is incomplete, outdated, scattered, or inaccessible. These questions help you find those issues before you commit to a timeline.

Ask for honest answers about data quality. Customers often overestimate how clean their data is until someone looks at a sample.

> **Example in practice**
>
> A manufacturing company wanted predictive maintenance alerts. The plan assumed five years of sensor data. When the FDE asked for a sample, the data showed that sensors had been replaced twice, units of measurement changed in 2023, and 30% of records had missing timestamps. The project was re-scoped to start with one production line where data was reliable.

How to ask these well

- Ask for a sample dataset as early as possible. A small sample reveals more than an hour of discussion.
- Identify the data owner by name. Access requests without an owner can take weeks.
- Ask about formats explicitly. Scanned PDFs and handwritten forms change the technical approach entirely.

Question	What You're Listening For	Red Flag
D1 MUST-ASK **Where does the data for this process live today?**	Specific systems, databases, shared drives, or document stores.	"It's somewhere in SharePoint." Data location is unknown.
D2 MUST-ASK **Who owns this data, and who can grant access?**	A named owner and a known access approval process.	No clear owner, so access requests have nowhere to go.
D3 MUST-ASK **How often is the data updated?**	Update frequency that matches how fresh the answers need to be.	Data is updated monthly, but users expect real-time answers.

D4 MUST-ASK **How accurate and complete is the data, honestly?**	A frank estimate, including known gaps and duplicate records.	"It's perfect." Almost no business data is.
D5 SITUATIONAL **What formats is the data in?**	Structured tables, PDFs, emails, scans, images, or audio.	Critical information exists only in scanned images or handwritten notes.
D6 MUST-ASK **Is there sensitive or personal data involved?**	Clear identification of personal, financial, or health data.	Uncertainty about whether sensitive data is present.
D7 SITUATIONAL **How much historical data is available?**	Enough history to test, evaluate, and, if needed, fine-tune.	Only a few weeks of data, or history that is no longer relevant.
D8 FOLLOW-UP **Can you share a sample dataset this week?**	A realistic, anonymized sample delivered on a committed date.	Repeated delays in sharing even a small sample.

Section E: Systems, Integrations & APIs

FDEs spend a large share of their build time connecting systems. An AI assistant that can't reach the CRM, or an automation that can't write back to the ERP, delivers little value. These questions identify every system involved, how it can be accessed, and who on the customer side will help.

Integration surprises are the most common cause of timeline overruns. A system with no API, strict rate limits, or a six-week security approval process must be known on day one.

> **Example in practice**
>
> A retail chain wanted an assistant to answer store managers' stock questions. The inventory system had an API, but it allowed only 100 requests per hour across the whole company. With 300 stores, live queries would have failed within minutes. The FDE designed a nightly sync into a separate database, which the assistant could query freely.

How to ask these well

- List every system mentioned in Section B and ask about each one individually.

- Ask for API documentation and a test account, not just a yes or no answer.
- Confirm early whether the solution needs to write data back or only read it, since write access requires more approvals.

Question	What You're Listening For	Red Flag
E1 MUST-ASK **Which systems must this solution connect to?**	A complete list, including legacy and third-party tools.	New systems keep appearing in later conversations.
E2 MUST-ASK **Do these systems have APIs, and are they documented?**	Available APIs with current documentation and examples.	No API, or documentation that is years out of date.
E3 SITUATIONAL **Are there rate limits, usage costs, or licensing restrictions?**	Known limits and costs that can be planned around.	Limits discovered only after the build starts.

E4 SITUATIONAL **How do users sign in today?**	SSO, Active Directory, or OAuth that the solution can reuse.	Each system has separate logins with no central identity provider.
E5 MUST-ASK **Where should the solution be hosted?**	A clear choice: customer cloud, vendor cloud, or on-premises.	Hosting is undecided and depends on an unscheduled security review.
E6 MUST-ASK **Should the solution only read data, or also write back to systems?**	A clear scope for read and write access.	Write access is expected but no one has approved it.
E7 MUST-ASK **Who on the IT team will support integration work?**	A named IT contact with time allocated to the project.	IT is not involved yet or has no capacity this quarter.
E8 FOLLOW-UP **Do you have a test or sandbox environment?**	A non-production environment with realistic data.	Testing must happen directly in production.

Section F: AI-Specific Requirements (LLM, RAG, Agents)

AI projects have requirements that traditional software projects don't. You need to understand what the AI should answer, what it must never do, how correctness will be judged, and what a wrong answer costs. These questions shape decisions about model choice, retrieval design, guardrails, and human review.

Ask for real examples of questions or tasks. Twenty real questions from users are worth more than any written requirement, because they become your first evaluation set.

> **Example in practice**
>
> A law firm wanted a research assistant for contract review. During discovery, the FDE asked what the AI should never do. The partners answered immediately: it must never present a conclusion without citing the exact clause. That single requirement led to a RAG design that returned clause-level citations with every answer and a confidence flag for anything uncertain.

How to ask these well

- Collect real example questions early and use them as your test set.
- Ask what a wrong answer costs. A wrong product recommendation and a wrong medical answer need very different safeguards.
- Clarify whether the AI should act or only suggest. Agent-based actions need approval flows and audit logs.

Question	What You're Listening For	Red Flag
F1 MUST-ASK **What questions or tasks should the AI handle? Can you share 20 real examples?**	Real user questions that define scope and form the test set.	Only hypothetical examples, with no access to real queries.
F2 MUST-ASK **What should the AI never answer or do?**	Clear boundaries such as legal advice, pricing promises, or medical diagnoses.	No boundaries considered, which suggests risk hasn't been discussed.
F3 MUST-ASK **How will users check that an answer is correct?**	Citations, source links, or a review step built into the workflow.	Users are expected to trust answers without any way to verify them.

F4 MUST-ASK **What accuracy is acceptable, and what does a wrong answer cost?**	A realistic target linked to business impact.	An expectation of 100% accuracy with no human review.
F5 SITUATIONAL **Should answers include citations or links to source documents?**	A clear preference that shapes retrieval and response design.	Citations are needed, but source documents are not accessible.
F6 SITUATIONAL **Should the AI take actions, or only recommend them?**	Clear scope for actions such as updating records or sending emails.	Full automation expected with no approval or audit trail.
F7 SITUATIONAL **Are there approved or banned AI models or vendors?**	A known policy on approved models and data processing.	Policy is unknown and will be decided after the build.
F8 FOLLOW-UP **How fast does a response need to be?**	A latency expectation linked to the user's workflow.	Instant responses expected for complex multi-step tasks.

Section G: Security, Compliance & Constraints

Security and compliance requirements can change the architecture of a solution completely. Knowing early that data can't leave a specific region, or that every AI decision must be logged for auditors, prevents expensive redesigns later.

These questions also help you plan the timeline. Security reviews in large organizations can take weeks, so they need to start in parallel with development, not after it.

> **Example in practice**
>
> A healthcare provider in India wanted to summarize patient discharge notes using an LLM. During discovery, the compliance officer confirmed that patient data could not be processed outside the country under their data protection policy. The FDE selected a model deployment hosted in an Indian cloud region and added automatic masking of patient identifiers before processing.

How to ask these well

- Invite the security or compliance contact to at least one call. Don't rely on secondhand answers.
- Ask how long the security review takes and start the paperwork early.
- Document every constraint in writing and confirm it with the customer.

Question	What You're Listening For	Red Flag
G1 MUST-ASK **Can data leave your cloud environment or geographic region?**	A clear data residency rule that guides hosting and model choice.	No one is sure, and no one can confirm.
G2 MUST-ASK **Which regulations or standards apply (HIPAA, GDPR, SOC 2, DPDP Act)?**	A specific list of applicable regulations and internal policies.	Regulated data is involved, but compliance hasn't been consulted.

G3 MUST-ASK **Is a security review required before deployment, and how long does it take?**	A known process and timeline that can run in parallel with the build.	A review lasting several months that hasn't been started.
G4 SITUATIONAL **Are there logging, audit trail, or data retention requirements?**	Defined logging and retention rules for inputs and outputs.	Audit requirements discovered after deployment.
G5 SITUATIONAL **Who needs to approve the use of AI in this process?**	A named approver or an AI governance group.	A pending company-wide AI policy that could block the project.
G6 FOLLOW-UP **Are there technical constraints we should know about?**	Legacy systems, network restrictions, approved languages, or tools.	Constraints mentioned only after the architecture is agreed.

Section H: Success Metrics & ROI

If success isn't defined before you build, it will be defined by opinion afterward. These questions establish a baseline, a target, and a review

process. They also help you prioritize features, because anything that doesn't move the success metric can wait.

Push for numbers. "Faster" is not a metric. "Reduce average report preparation time from six hours to one hour" is.

> **Example in practice**
>
> A customer support team wanted an AI assistant to "help agents." The FDE asked how they measured agent performance today. They tracked average handling time, currently 11 minutes per ticket. They agreed that reducing it to 8 minutes within 60 days of launch would justify a full rollout. Every later feature decision was tested against that number.

How to ask these well

- Always ask for the current baseline before discussing targets.
- Agree on a review date and the person who will decide whether to scale.
- Ask what failure looks like. It often reveals expectations that success questions miss.

Question	What You're Listening For	Red Flag
H1 MUST-ASK **How do you measure this process today?**	An existing metric, such as time, cost, error rate, or volume.	Nothing is measured, so improvement can't be proven.
H2 MUST-ASK **What number would make this a clear win?**	A specific target agreed by the sponsor.	"We'll know it when we see it."
H3 SITUATIONAL **How will you measure success 30, 60, and 90 days after launch?**	A staged evaluation plan with checkpoints.	Success expected immediately at launch, with no ramp-up period.
H4 SITUATIONAL **Who will review results and decide whether to scale?**	A named decision-maker and a review date.	No owner for the decision, so pilots never become rollouts.
H5 SITUATIONAL **What cost or time savings are you expecting?**	A realistic estimate the customer can justify internally.	Expectations far beyond what the process volume allows.

H6 FOLLOW-UP **What would make you consider this project a failure?**	Honest concerns that reveal hidden expectations.	Failure criteria that contradict the stated success metrics.

Section I: Timeline, Deployment & Handover

The final section covers how the solution will reach users and who will own it afterward. Many FDE projects succeed technically but struggle after handover because no one on the customer side is ready to maintain, monitor, or improve the solution.

Plan the handover from the first week, not the last. Knowing who will own the solution affects how you document it and which tools you choose.

> **Example in practice**
>
> An FDE delivered a document-processing pipeline for an insurance company in eight weeks. Two months later, the pipeline stopped working after an upstream system changed its file format, and no one internally knew how to fix it. On the next project, the FDE asked about

> ownership in the first call, paired with an internal engineer throughout, and delivered a runbook at handover.

How to ask these well

- Ask whether the deadline is fixed or flexible, and what is driving it.
- Recommend a pilot with one team when the risk or uncertainty is high.
- Agree on a communication rhythm, such as a weekly 30-minute update, from day one.

Question	What You're Listening For	Red Flag
I1 MUST-ASK **Is there a hard deadline driving the launch?**	A real date tied to an event, such as an audit, season, or contract.	An aggressive date with no reason behind it.

I2 SITUATIONAL **Would you prefer a pilot with one team before a full rollout?**	Openness to starting small and learning quickly.	Full rollout across all teams expected on day one.
I3 MUST-ASK **Who will maintain and own the solution after go-live?**	A named internal owner with the right skills or a support agreement.	No one is assigned to maintain the solution.
I4 SITUATIONAL **What documentation or training does your team need at handover?**	Clear expectations for runbooks, training sessions, and support.	Handover expectations undefined until the final week.
I5 FOLLOW-UP **How should we share progress, and how often?**	An agreed rhythm, channel, and audience for updates.	Updates expected only at the end of the project.

Worked Example: Retail Inventory Reporting

This example follows a discovery engagement with a mid-sized retail chain of 40 stores. The initial request came from the head of operations: "We

need an AI dashboard for inventory reporting." Here is how six questions from this bank changed the direction of the project.

The Discovery Conversation

Question Asked	What the Customer Said	What It Revealed
A1. What triggered this project now?	"Last month we ran out of our best-selling item in 12 stores during a festival sale while the warehouse had plenty."	The real issue is stock visibility across locations, not reporting in general.
B1. Walk me through the last time this report was prepared.	"An analyst downloads data from the POS system, the warehouse system, and the e-commerce platform every Monday, then combines it in Excel."	Data sits in three disconnected systems. The report is built manually once a week.
B2. How long does the process take?	"About six hours every Monday. By the time managers see it, it's already outdated."	Baseline: six hours of manual effort, and data that is up to seven days old.
D3. How often is the data updated?	"POS updates in real time, the warehouse system nightly, and e-commerce every hour."	Daily refresh is achievable. Real-time across all three isn't required.

E2. Do these systems have APIs?	"POS and e-commerce do. The warehouse system only exports CSV files to a shared folder."	Integration needs two API connectors and one automated file pickup.
H2. What number would make this a clear win?	"If store managers see stock levels by 9 a.m. every day and stock-outs on top items drop by half."	Clear success metrics: daily data by 9 a.m. and a 50% reduction in stock-outs.

The hidden problem

The customer asked for an AI dashboard. What they actually needed was a reliable daily data pipeline that combined three systems, plus early alerts when a high-demand item was running low in one store but available elsewhere. The "AI" part was useful but secondary: a simple demand forecast on top of clean, combined data.

The resulting solution design

- Automated daily data pipeline pulling from POS and e-commerce APIs and picking up warehouse CSV exports each night.
- A combined inventory database refreshed by 7 a.m. every day.
- Stock-out risk alerts sent to store managers when top items fall below a threshold, with suggested transfers from nearby stores.
- A simple dashboard for the operations head showing stock levels, alerts, and transfer history.
- A 30-day pilot with 5 stores before rolling out to all 40.

Without discovery, the team would likely have built a visually impressive dashboard on top of the same six-hour manual process. With it, they removed the manual work entirely and targeted the outcome the business cared about.

Discovery Red Flags Cheat Sheet

Some answers during discovery signal problems that will grow if you ignore them. Use this cheat sheet to recognize the most common warning signs and respond before they turn into delays, scope creep, or failed adoption.

Red Flag	What It Usually Means	How to Respond
"Everyone is doing AI, so we need it too."	No specific business problem has been identified.	Ask about a recent incident that cost time or money. Anchor the project to that.
No one owns the data.	Access requests will stall and data quality issues will have no fixer.	Ask the sponsor to name a data owner before the build begins.
Success is defined as "make it smarter."	Results will be judged on opinion rather than evidence.	Propose one measurable target and a baseline, and get it agreed in writing.
IT isn't in the room.	Integration, security, and hosting decisions will arrive late.	Schedule a technical session with IT within the first week.
"The data is perfect."	Data quality hasn't been checked.	Request a sample and review it together before committing to a timeline.

End users haven't been consulted.	Adoption risk is high, even if the build succeeds.	Ask to shadow or interview two or three real users.
100% accuracy is expected.	Expectations about AI are unrealistic and risky.	Discuss the cost of errors and design a human review step.
The deadline has no reason behind it.	The scope may be cut in unplanned ways under pressure.	Ask what drives the date and propose a phased delivery.
No one will own the solution after launch.	The solution will degrade once you leave.	Agree on an internal owner and pair with them during the build.
Requirements change in every meeting.	Stakeholders haven't agreed among themselves.	Run a prioritization session with the sponsor and freeze the pilot scope.

> **Remember**
>
> A red flag isn't a reason to walk away. It is a risk to name, discuss openly with the customer, and plan around. Customers usually appreciate an FDE who spots problems early.

Templates

Print these templates or copy them into your notes tool. Use the first during the discovery call and the second to send a written summary to the customer within 24 hours.

Template 1: Discovery Call Notes

Customer	Date	Attendees

A. Business context and goals (trigger, sponsor, outcome)

B. Current workflow (steps, time taken, workarounds, bottleneck)

C. Users and stakeholders (users, approvers, affected teams)

D & E. Data sources and systems (location, owner, APIs, formats)

F & G. AI requirements and constraints (scope, boundaries, compliance)

H. Success metrics (baseline, target, review date)

I. Timeline and ownership (deadline, pilot, owner after launch)

Red flags noticed

Template 2: Post-Call Summary

Send this to the customer within 24 hours. It confirms your understanding, gives them a chance to correct mistakes, and creates a written record of agreed scope.

Problem statement (one or two sentences in the customer's language)

Constraints (security, compliance, budget, timeline)

Data and systems map (system, data owner, access method)

Success metrics (baseline and target)

Open questions

Next steps (action, owner, due date)

Example problem statement

"Store managers can't see accurate stock levels across locations because inventory data is combined manually once a week from three systems. This leads to stock-outs on high-demand items while stock is available elsewhere. The goal is daily stock visibility by 9 a.m. and a 50% reduction in stock-outs on top items within 60 days."

GSDC CERTIFIED FORWARD DEPLOYED ENGINEER

Turn Discovery Skills Into a Certified FDE Career

Asking the right questions is where every successful deployment begins. The GSDC Certified Forward Deployed Engineer program helps you build the complete skill set that companies look for in customer-facing engineers, from discovery and solution design to AI integration and production deployment.

What you'll build

- **Customer discovery and requirement scoping:** turn vague requests into clear, measurable problem statements.
- **AI solution design:** work with LLMs, RAG pipelines, and AI agents for real business use cases.
- **Integration and deployment:** connect APIs, data sources, and cloud environments into working systems.

- **Industry-recognized credential:** validate your FDE capabilities for employers and clients.

GSDC
Global Skill Development Council

CERTIFIED FORWARD DEPLOYED ENGINEER

LEARN DIRECTLY FROM INDUSTRY EXPERTS, AI PRACTITIONERS, AND ENGINEERING LEADERS WHO ARE BUILDING AND DEPLOYING SCALABLE AI DRIVEN SOLUTIONS FOR ENTERPRISES WORLDWIDE.



ABOUT GSDC CERTIFICATION

EBOOK

Extensive and exclusive Ebook created by world's experts to help you with understanding core concepts.

CREATED BY EXPERTS

GSDC certifications are created and authored by world's leading experts in the field.

EBOOK

Extensive and exclusive Ebook created by world's experts to help you with understanding core concepts.

LEARNING MATERIALS

Get access to learning materials such as videos, ebooks, templates, and practice exams, which will help you clear the certification exam.

LEARNING OBJECTIVE

- Validate your Forward Deployed Engineering expertise.
- Increase your earning potential with an in-demand credential.
- Get complimentary career assistance after certification.

Enroll now with the code **LEARN20** To avail **20%** discount

Enroll Now

www.gsdcouncil.org