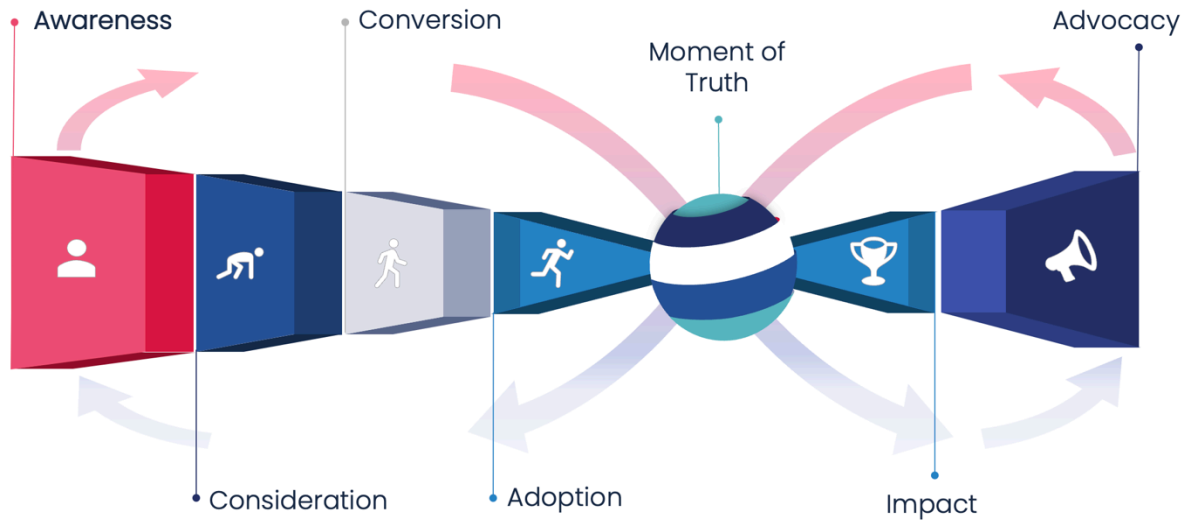
Worksheet: Discover Data User Journeys, Expectations & Blockers

Purpose: Identify your real data users (not just "stakeholders"), map the end-to-end user journey, and surface blockers + expectations at each stage.

Time: 60–90 minutes **Group size:** 4–12 **Materials:** Whiteboard / Miro / sticky notes, pens, timer



---

1) Define the "Data Product" You're Investigating (5 mins)

What data / capability are we mapping today?

Name: _____

Example: "Workforce Absence Dashboard (Monthly)"

What does it enable people to do? (1 sentence)

_____

Example: "Monitor sickness trends and compare absence rates across teams and regions."

What's the output? (tick one or more)

☐ Dashboard / report

☐ Search / catalogue

☐ Dataset / extract

☐ Data pipeline / feed

☐ API / service

☐ Other: _____

Who “owns” it today?

Team / role: _____

Example: “People Analytics / HR Data Team”

---

2) Identify Your Data Users (10–15 mins)

List every type of person who interacts with this data (directly or indirectly).

Don’t worry about job titles — capture roles “in the wild

User Type / Role	Example person/team	What they use it for (job to be done)	Frequency
Service manager	Ops Team Lead	Understand performance and identify where to intervene	☐ Daily ☑ Weekly ☐ Monthly ☐ Ad hoc
Analyst	BI Analyst	Build a report combining this dataset with others	☐ Daily ☐ Weekly ☑ Monthly ☐ Ad hoc
Researcher	Policy Research Unit	Explore trends and test hypotheses	☐ Daily ☐ Weekly ☐ Monthly ☑ Ad hoc
Developer / data engineer	Platform Team	Automate pipelines / improve reliability	☑ Daily ☐ Weekly ☐ Monthly ☐ Ad hoc
Governance / IG	Information Governance Lead	Ensure use is compliant and controlled	☐ Daily ☐ Weekly ☐ Monthly ☑ Ad hoc

Prompt questions (if you get stuck):
Who requests it? Who consumes it? Who interprets it? Who signs off on decisions from it?
Who gets blamed when it’s wrong? Who supports it when it breaks?

3) Segment Users by “Mode of Use” (5–10 mins)

Assign each user type to a segment:

Segments (tick one per user type):

- ☐ **Explorer** (finding & assessing what exists)
- ☐ **Consumer** (using outputs like dashboards/reports)
- ☐ **Analyst** (querying, combining, validating)
- ☐ **Builder** (pipelines, models, APIs)
- ☐ **Decision-maker** (using insights to act)
- ☐ **Governor** (controls, approvals, policy)
- ☐ **Operator** (monitoring reliability + incidents)

User Segment Map

User Type	Segment	Notes
Service manager	Consumer / Decision-maker	Needs fast answers and confidence in “what changed?”
BI Analyst	Analyst	Needs definitions, lineage, and join guidance
Platform engineer	Builder / Operator	Needs stable schemas, monitoring, and clear ownership

4) Map the Data User Journey (Core Exercise) (25–40 mins)

Select a single user type (for example, a Business Analyst) and run a short thought experiment: imagine they are a **brand-new recruit** joining your organisation with no prior knowledge of your systems, acronyms, or internal networks. Now walk with them through each stage of the data journey and ask: *what would they do, what would they expect, and where would they get stuck?*

Stages of the Data Journey

1. **Discover** – *If you needed workforce data, how would you even find out whether it exists — and where to look for it?*
2. **Access** – *Can I get it? What approvals, tools, or steps are required?*
3. **Understand** – *Can I trust it and interpret it correctly? What does each field actually mean?*
4. **Use** – *Can I apply it to my task? Can I query, join, analyse, and work with it without workarounds?*
5. **Share** – *Can I safely share results? What are the rules, limits, and disclosure requirements?*
6. **Improve** – *If something looks wrong or missing, can I report it, request changes, or contribute improvements—and will anything happen as a result?*

Expectations & Blockers Grid (print one per user type)

User Type: _____

Example: “BI Analyst (central team)” **Primary job-to-be-done:**

Example: “Create a dashboard showing regional service demand and forecast pressure points.”

Expectations & Blockers Grid (with examples)

Stage	What they are trying to do	Expectations ("It should…")	Blockers ("It fails when…")	Evidence / signal it's working	Owner who can fix it
1) Discover	Find relevant datasets fast	"I can search by topic and synonyms" / "It shows what's available"	No catalogue / search returns nothing / naming is internal jargon	Users find 2–3 usable datasets in <10 mins	Data catalogue owner / metadata team
2) Access	Get access or request it	"Access steps are clear" / "Approval doesn't take weeks"	Unclear process / stuck in mailbox / approvals require rework	Request approved first time, within defined SLA	Data owner + IG
3) Understand	Check meaning & quality	"Definitions are clear" / "Known issues are visible"	No data dictionary / unclear fields / conflicting metrics	Users interpret correctly without calling a colleague	Data steward
4) Use	Combine + analyse	"Examples exist" / "Joins are explained" / "Schema is stable"	IDs don't match / broken joins / unexpected changes	Fewer workarounds, fewer duplicated extracts	Data engineering
5) Share	Publish results safely	"Rules for sharing are clear" / "Outputs can be reused"	Fear of 'getting it wrong' / unclear disclosure rules	Outputs are reused, cited, and trusted	Governance + analytics lead
6) Improve	Report issues + request changes	"There's a clear feedback loop" / "Fixes don't disappear into nowhere"	No ticket route / no owner / fixes not prioritised	Issues tracked + communicated back to users	Product owner

6) Add “Hidden Blockers” (10 mins)

These are blockers that don’t show up in tickets because users work around them.

Hidden blockers checklist (with examples)

☑ People don’t know the data exists

☑ People don’t know who to ask for access

☐ Access approvals take too long

☑ Definitions vary by team (“What does ‘active’ mean?”)

☑ Data quality issues are discovered too late

☑ Data is technically accessible but unusable (format/tooling)

☐ People export to spreadsheets to finish the job

☑ Users don’t trust it even when it’s correct

☐ No feedback loop / fixes don’t reach upstream

☐ Users build shadow copies / parallel pipelines

Where do users create workarounds?

Example: “Analysts keep a private spreadsheet mapping ‘region names’ to official codes because the dataset doesn’t match system IDs.”

---

7) Rank the Biggest Pain Points (10 mins)

For each blocker, score impact and frequency, then pick “top 3”.

Pain Point Ranking (with examples)

Blocker	Stage	Impact (1–5)	Frequency (1–5)	Total	Notes
Access requires 3 approvals and no one knows the owner	Access	5	4	9	People abandon requests mid-way
Definitions of “active case” differ across teams	Understand	5	3	8	Results conflict between dashboards
Schema changes break dashboards silently	Use	4	4	8	Causes rework + loss of trust

Top 3 blockers to fix first:

1. _____
2. _____
3. _____

---

8) Outputs: Turn Findings into Actions (10 mins)

“Fix-It” Backlog Starter (with examples)

	Who it hurts	What success looks like	Quick win / longer-term	Owner
No clear access pathway	Analysts, researchers	One request link + named owner + SLA	☑ Quick win ☐ Longer term	Data owner + IG
Definitions inconsistent	Everyone using metrics	Single glossary + agreed definitions	☐ Quick win ☑ Longer term	Data stewardship
Schema breaks downstream	BI team	Change notifications + versioning	☐ Quick win ☑ Longer term	Data engineering

Facilitator Script (Quick)

1. “Let’s list every type of person who touches this data.”
2. “Pick the top 3 user types by importance.”
3. “For each, walk through the journey stages.”
4. “Write expectations as ‘it should…’ statements.”
5. “Write blockers as concrete failure modes.”
6. “Rank by pain and commit to 3 fixes.”