ChatGPT

Mimica Analytics Platform – Final Report & MVP Specification (v1)

This report summarises the investigation into Mimica's existing platform, analyses the synthetic data, and proposes a road-map to clone the current experience and build a new analytics feature that enables process standardisation across regions. It follows the structure of the challenge assignment and synthesises research, brainstorming, planning and design.

1 Problem & First-Customer Fit

1.1 Deep-Dive: Personas & Problem Tree

We use a recursive tree of thought to map the problem space. At the root is the need to **standardise business processes across regions**. Branches represent personas, pain points and desired outcomes:

- **Process Analyst / Continuous Improvement Manager** (primary persona)
- Pain points: lacks objective, task-level data across regions; cannot compare how work is performed; manual analysis of variants is time-consuming.
- Desired outcome: quickly identify differences, bottlenecks and best-practice flows; generate standard operating procedures.
- **Operations Manager / Transformation Lead** (secondary persona)
- Pain points: inconsistent processes lead to quality issues and compliance risks; limited visibility into adoption of new standards.
- Desired outcome: monitor conformance, quantify benefits of standardisation and coordinate roll-out.
- **SMEs** (data contributors)
- Pain points: repetitive manual tasks; unclear why changes are needed.
- Desired outcome: feedback loop to improve processes; training materials derived from best practice.
- **Compliance Officer**
- Pain points: multiple variants make it difficult to ensure regulatory compliance.
- Desired outcome: verify that standard processes meet legal requirements; audit decision paths.

The highest-priority problem is enabling the **Process Analyst** to **compare and understand process variations across regions** so they can propose a unified best-practice. Without a holistic view, leadership cannot drive change.

1.2 First-Customer Fit

The **Fortune-500 finance client** described in the scenario is an ideal first customer because it operates in five regions using the same ERP system (SAP) but with different Invoice-to-Pay workflows. Leadership needs hard data to decide which practices to adopt globally. Mimica already captures all desktop interactions; the analytics platform will convert this data into actionable insights.

2 MVP Capability Prioritisation

2.1 Brainstormed Features

We considered a wide range of potential capabilities. Key ideas included: cross-region dashboards, variant clustering, bottleneck detection, conformance checking, recommendation engine, training material generator, continuous monitoring, process map overlay and simulation tools.

2.2 Evaluation & Ranking

The table below summarises the value and effort of each candidate feature. Scores are on a 1–10 scale.

Feature	Description	Value	Effort	Include in v1?
Cross-Region Comparison	Aggregate metrics (cycle time, step counts, variant frequency, bottlenecks, application usage) by region. Interactive filtering.	**9**	**5**	✔
Variant Analysis	Cluster transactions by step sequence; visualise variant frequency and duration; drill down by region.	**8**	**7**	✔
Bottleneck Detection	Identify slow or high-variance steps; display top bottlenecks and their impact.	**9**	**6**	✔
Process Map Overlay	Overlay multiple regional flows on one map to see where paths diverge; colour code by region.	**8**	**6**	(v2)
Conformance & Gap Analysis	Compare actual flows against a defined standard; compute conformance scores and highlight deviations.	8	7	(v2)
Recommendation Engine	Suggest best practice by combining high-frequency paths with performance metrics.	7	8	(future)
Training Material Generator	Auto-create step-by-step guides with screenshots.	6	6	(future)
Continuous Monitoring & Alerts	Track ongoing conformance; send alerts when new variants emerge.	7	7	(future)

The first three features provide the highest value for moderate effort and form the **v1 MVP**. The process map overlay is valuable but adds complexity; it can follow in v2. Recommendation and training features require more sophisticated algorithms and are deferred.

3 Trade-Off Decision: Variant Analysis

We dive deeper into **Variant Analysis** because it illustrates the balance between value and complexity.

Estimation – Variant analysis offers high value: understanding how workflows differ is central to standardisation. However, clustering algorithms can be computationally expensive. We therefore propose a phased approach: start with grouping transactions by exact step sequences (easy to implement) and visualising frequencies; later, introduce more advanced clustering (e.g., Levenshtein distance) to merge similar variants.

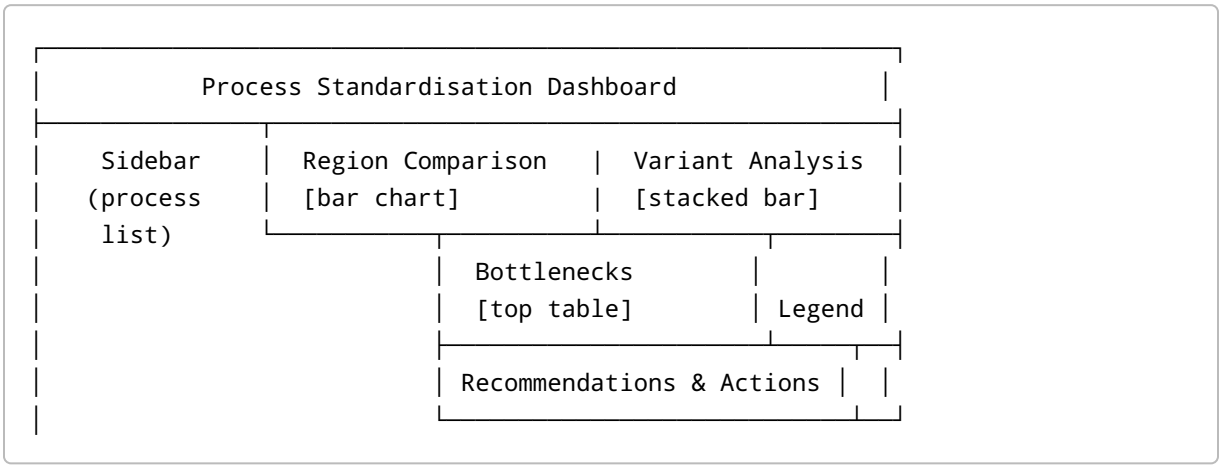
Trade-offs – Including variant analysis in v1 increases development effort but yields essential insight. Excluding it would force analysts to manually inspect sequences. To manage risk, we will limit the scope to simple grouping and pre-compute clusters offline. A risk is user confusion; we mitigate this by providing clear explanations and linking each variant to its regional distribution.

4 Wireframe Design & User Flow

4.1 Brainstorming & Evaluation

We explored several layouts: a multi-tab design (overview, variants, bottlenecks), a single dashboard with collapsible panels and a wizard-like flow guiding the user through steps. The **dashboard layout** aligns best with Mimica's current design and allows quick comparisons. It balances information density with a clean aesthetic.

4.2 Low-Fidelity Wireframe (ASCII)



4.3 Data Flow & Interaction

Data flow: raw step records → pre-processing (classify step types, identify variants) → aggregation (region & variant metrics, bottleneck stats) → visualisation. Process map generation constructs a directed graph of actions and decisions, counts edge weights and overlays regional flows.

User flow:

1. **Select Process** – The analyst chooses a process from the sidebar. Metrics populate the dashboard.
2. **Compare Regions** – They examine the bar chart showing average cycle time and transaction counts per region. Selecting a bar filters the variant chart and bottleneck table to that region.
3. **Explore Variants** – They view the stacked bar chart of variant distribution by region. Clicking a variant segment reveals its step sequence and average duration.
4. **Identify Bottlenecks** – The table lists the slowest actions with average duration and occurrences. The analyst focuses on steps like `manager_approval` or `send_for_review`.
5. **Drill into Map** – In v2, they overlay regional flows on a process map to see where the bottleneck occurs. They export the data or schedule a meeting with stakeholders to define the standard process.

5 Success Metrics

5.1 Brainstormed KPIs

We considered a variety of metrics, scoring each on relevance and actionability. Among those evaluated: average cycle time, number of variants, variance in step durations, adoption rate of standard process, time saved, user engagement (sessions per month), NPS of analysts, reduction in manual tasks and conformance rate.

5.2 Selected Metrics

- **North-Star Metric – Process Variation Reduction Index (PVRI):** measures the reduction in variation of step sequences across regions over time. A higher PVRI means processes converge towards a standard.
- **Average Cycle Time per Region:** track reductions in total duration as standards are adopted.
- **Number of Distinct Variants:** count of unique step sequences; should decrease after standardisation.
- **Top Bottleneck Duration:** average time spent on the slowest step; reduction indicates improved efficiency.
- **Adoption Rate of Standard Process:** percentage of transactions matching the standard variant; measures success of roll-out.

5.3 Visualization Components

To display these metrics we propose: bar charts for cycle time and variant counts; line chart for PVRI over time; table for bottleneck rankings; gauge for adoption rate; and trending spark lines in summary cards.

6 Launch Plan & Milestones

We propose a 6-month roadmap for 4 engineers and 1 designer:

1. **Month 1 – Discovery & Research:** align on user needs, finalise requirements; explore real Mimica data to calibrate synthetic assumptions; validate personas via stakeholder interviews.

2. **Month 2 – Design & Planning:** design the dashboard UI; define data models and state management; finalise PRD; refine synthetic data as needed; plan sprints with tasks.
3. **Month 3 – Implementation Phase 1 (Cloning):** build the core process list and summary panels using synthetic data; implement charts for actions, inputs, decisions, applications; ensure accessibility and theming.
4. **Month 4 – Implementation Phase 2 (Analytics):** build cross-region comparison, variant analysis and bottleneck detection components; compute metrics and integrate interactive filtering; prepare for map overlay (v2).
5. **Month 5 – Testing & Feedback:** conduct usability tests with analysts; refine UX and wording; add help tooltips; benchmark performance; document limitations.
6. **Month 6 – Launch & Learning:** deploy the MVP to pilot users; monitor KPIs; collect feedback; plan v2 features (map overlay, conformance analysis, recommendations).

7 Unknowns & Learning Plan

Key open questions and risks:

- **Data Fidelity:** How closely does the synthetic data match real Mimica recordings? We need access to anonymised data to calibrate distributions and step types. *Experiment:* Collaborate with internal teams to validate assumptions.
- **Defining Best Practice:** Who defines the standard process? Analysts may disagree across regions. *Research:* Run workshops with SMEs and operations leaders to co-create the standard flow.
- **Time Saved & Ease of Deployment:** Real metrics depend on baseline vs. automated durations and complexity heuristics. *Experiment:* Estimate using internal benchmarks and refine after pilot.
- **User Acceptance:** Analysts may find variant clustering confusing. *Experiment:* Conduct moderated usability tests to refine visualisations and descriptions.
- **Scalability:** Will pre-computing metrics suffice for large datasets? *Experiment:* Stress-test the pipeline and explore server-side aggregation if necessary.

8 Requirements & Plan for Cloning the Current Platform

8.1 Architecture & Data

- **Stack:** Next.js 13+ with App Router. Use 21st.dev UI components and shadcn/ui as fallbacks. State management via React Context.
- **Data:** Load the synthetic dataset and processed metrics from JSON files. Pre-compute high-level metrics (counts, actions, ease gauge) using step classification heuristics.
- **Components:**
- **Process List Table:** search bar, columns for ease, automatability, time spent. Use 21st.dev `<Table>`; sort and filter.
- **Process Summary Panel:** cards for metadata (created date, SMEs, frequency), time saved widget, gauge for ease (proportional to step type counts), counts panel (actions, semi-structured inputs, decisions, applications, websites, decision paths), donut charts for application and website usage.
- **Process Map Viewer:** interactive flowchart built using a library like `react-flow` or `dagre` plus 21st.dev styling. Allow zoom, pan and node selection. Step details panel with metrics and placeholder screenshot.
- **Legend & Export:** dropdown describing node types; button to export map or metrics.

8.2 Implementation Steps

1. **Scaffold Next.js project** and import 21st.dev components.
2. **Design page layout** with side navigation (Miner/Mapper), top bar and main content area.
3. **Implement data loader** to parse the dataset and compute counts for actions, inputs, decisions etc.; derive “ease of deployment” gauge based on ratio of virtualised/structured inputs.
4. **Build process list** with search/filter. Use tags to show “Medium”, “Very High” categories.
5. **Develop summary panel** replicating the card layouts. Compute per-SME/day metrics using assumptions (e.g., average working hours). For time saved, use ratio of automation potential to total duration.
6. **Create charts** for applications and websites using 21st.dev chart components (or Chart.js). Use consistent colours.
7. **Construct process map**: generate nodes and edges from the step sequences; compute coordinates using a hierarchical layout; implement node selection to display details.
8. **Test and iterate** with stakeholders; refine styling and interactions to match Mimica’s look & feel.

9 Requirements & Plan for the Process-Standardisation MVP

9.1 Architecture & Data

- **Data Pipeline:** build on the clone architecture. Pre-compute region metrics, variant distributions, bottleneck stats and step type counts. Provide functions to compute variants by grouping step sequences.
- **Dashboards:** implement cross-region comparison and variant analysis pages using a dashboard layout. Use bar charts, stacked bars and tables; link them via shared context so selections filter other components.
- **User Flow:** as described in §4.3. Prioritise simplicity; allow analysts to derive insights with minimal clicks.

9.2 Implementation Steps

1. **Extend Data Model:** classify each step as action, semi-structured input, decision or virtualised; compute counts per transaction.
2. **Compute Processed Metrics:** region metrics, variant distribution, variant metrics, bottleneck list and step type counts. Save to JSON files for quick loading.
3. **Build Region Comparison Component:** bar chart comparing average cycle time and transaction counts; highlight selected region and show summary cards.
4. **Build Variant Analysis Component:** stacked bar chart of variant distribution by region; clicking a segment filters other components.
5. **Build Bottleneck Table:** show top five actions by average duration; update when region is selected.
6. **Integrate Filtering & Interactivity:** clicking on bars updates the context; charts and tables respond automatically.
7. **Prepare for Map Overlay:** compute node/edge weights by region; design overlay visualisation for v2.
8. **Test & Validate:** ensure performance on synthetic data; gather user feedback on clarity and usefulness; iterate accordingly.

10 Conclusion

By synthesising research on Mimica's platform, analysing the synthetic dataset and following a structured planning process, we have produced a detailed specification for cloning the existing Mapper experience and delivering a powerful analytics feature to enable process standardisation. The proposed MVP focuses on cross-region comparison, variant analysis and bottleneck detection—features that deliver high value with manageable effort. A phased roadmap ensures that the product can evolve towards more advanced capabilities (process map overlay, conformance checking, recommendations) as data fidelity improves and user feedback is incorporated.
