PRD: GAISE Insights

1. Product overview

1.1 Document title and version

• PRD: GAISE Insights
• Version: 0.1.0

1.2 Product summary

GAISE Insights is a lightweight web application that collects user feedback and product telemetry, automatically synthesizes themes, and surfaces prioritized product insights for product managers, support teams, and executives. The tool reduces time-to-insight by automating tagging, scoring impact, and delivering contextual reports and alerts.

GAISE Insights is designed for rapid setup with common helpdesk and analytics integrations, secure single-sign-on access, and an intuitive dashboard that highlights the highest-impact issues and trends. It focuses on actionable recommendations rather than raw data exploration.

2. Goals

2.1 Business goals

• Reduce time product teams spend triaging user feedback by 50% within 3 months of launch. (Priority: High)
• Decrease repeat support tickets related to top 5 identified issues by 20% in six months. (Priority: Medium)
• Drive adoption among product and support teams, targeting 40% weekly active use in the first quarter. (Priority: Medium)

2.2 User goals

• Quickly understand the most important user problems and their estimated impact.
• Filter and surface feedback by product area, severity, and source.
• Export concise reports and send alerts for high-severity items.

2.3 Non-goals

• This is not a full BI or dashboarding platform for arbitrary analytics.
• Not intended to replace a data warehouse or long-term historical analytics store.
• Not a customer support ticketing system (it integrates with ticketing systems but does not replace them).

3. User personas

3.1 Key user types

• Product manager
• Support agent / triage owner
• Data analyst
• Executive / stakeholder

3.2 Basic persona details

• **Product manager**: Prioritizes roadmap items and needs quick, evidence-backed insights.
• **Support agent**: Needs to tag and escalate common issues; uses the tool to confirm recurring problems.
• **Data analyst**: Validates automated tagging and pulls exports for deeper analysis.
• **Executive**: Reviews summarized trend reports and high-level KPIs.

3.3 Role-based access

• **Admin**: Full access — manage integrations, users, retention settings, and export capabilities.
• **Analyst**: Access to raw data exports, tagging rules, and advanced filters.
• **User (Product/Support)**: View dashboards, receive alerts, annotate items, and change item priority.
• **Guest/Viewer**: Read-only access to dashboards and exported reports.

4. Functional requirements

• **Ingest feedback from sources** (Priority: High)
• Integrate with Zendesk, Intercom, email, and CSV upload.
• Schedule periodic imports and support manual one-off imports.
• **Automatic topic tagging and sentiment** (Priority: High)
• Auto-tag feedback into themes and detect basic sentiment.
• Allow manual override of tags.
• **Priority scoring and impact estimation** (Priority: High)
• Score issues by frequency, sentiment, and criticality to produce a ranked list.
• **Dashboard and filtered views** (Priority: High)
• Provide default dashboard cards: top issues, trend lines, and recent critical items.
• Filters by date range, source, tag, and product area.
• **Alerts & notifications** (Priority: Medium)
• Send Slack/email alerts for items above a configurable severity threshold.
• **Export & reporting** (Priority: Medium)
• Export filtered views to CSV and generate a PDF summary report.
• **Role-based access control & SSO** (Priority: High)
• Support SAML/OAuth for enterprise SSO and enforce role permissions.
• **API** (Priority: Low)
• Provide read-only REST endpoints for integration with downstream systems.

5. User experience

5.1. Entry points & first-time user flow

• Entry points: web app URL, SSO redirect, Slack alert link.
• First-time flow: quick onboarding wizard that connects one data source, runs a short import, and shows the "Top insights" card with example items.
• In-app tips explain scoring and tagging and link to documentation.

5.2. Core experience

• **Connect data source**: User links Zendesk/Intercom or uploads a CSV.
• Onboarding shows example items so users see immediate value.
• **Auto analysis**: System ingests data, auto-tags themes, and assigns priority scores.
• Provide a progress indicator and short summary after the import.
• **Review & triage**: Users review ranked issues, apply manual tags, and mark items for follow-up.
• Triage actions are persisted and feed back into the scoring model.
• **Share & act**: Users export reports or send alerts to teams.
• Allow quick Slack message creation from an insight card.

5.3. Advanced features & edge cases

• Manual tagging and rule creation for custom topics.
• Bulk edit and bulk export for large imports.
• Graceful handling of malformed CSV rows with an import error report.
• Data sampling mode for very large datasets (first 100k rows by default).

5.4. UI/UX highlights

• Compact KPI cards with clear primary metric and supporting context.
• Inline explanations for scoring and tag confidence.

• Accessible color choices and keyboard navigability for lists.
• Lightweight onboarding that produces a visible insight in < 2 minutes.

6. Narrative

María is a product manager who wants to quickly identify and prioritize the most frequent and severe user problems because she needs to focus the team on the changes that will reduce churn and support load; she finds this tool and immediately sees ranked issues with evidence and suggested next steps, allowing her to assign follow-up without deep data-analysis work.

7. Success metrics

7.1. User-centric metrics

• Time to first actionable insight after onboarding (target: < 10 minutes).
• Weekly active users among product/support teams (target: 40% in 3 months).
• Average time spent triaging per issue (target: reduction by 50%).

7.2. Business metrics

• Reduction in repeat support tickets for top issues (target: −20% in 6 months).
• Feature adoption uplift where fixes are shipped (measured per release).
• Paid adoption or upgrade rate for teams using the product (if applicable).

7.3. Technical metrics

• Ingest pipeline latency (target: < 2 minutes for typical imports).
• Dashboard load time (target: < 2 seconds for common views).
• System availability (target: 99.9% uptime).

8. Technical considerations

8.1. Integration points

• Zendesk, Intercom, email-to-CSV ingestion, Segment, Slack notifications, SAML/OAuth for SSO.
• Optional export to Snowflake/BigQuery via CSV/FTP or API.

8.2. Data storage & privacy

• Store only metadata and redacted user content by default; PII is hashed/removed during ingestion.
• Configurable retention policy with automatic purging for GDPR compliance.
• TLS in transit and AES-256 at rest for stored data.

8.3. Scalability & performance

• Asynchronous ingestion pipeline with horizontal worker scaling.
• Cache computed scores and tag lookups for read-heavy dashboard traffic.
• Batch processing for large imports with progress tracking.

8.4. Potential challenges

• Ensuring tagging accuracy for short or ambiguous feedback.
• Onboarding friction for teams with multiple, fragmented data sources.
• Data privacy and right-to-be-forgotten compliance across integrated systems.

9. Milestones & sequencing

9.1. Project estimate

• Medium: 6–8 weeks.

9.2. Team size & composition

• Medium team: 3–5 people
• Product manager, 1–2 engineers, 1 designer, 1 QA specialist

9.3. Suggested phases

• **Phase 1:** Core ingestion, automatic tagging, and dashboard (4 weeks)
• Key deliverables: Zendesk/CSV ingest, basic auto-tagging, dashboard with top issues.
• **Phase 2:** Priority scoring, alerts, and SSO (2 weeks)
• Key deliverables: impact scoring, Slack/email alerts, SAML/OAuth sign-on.
• **Phase 3:** Exports, advanced filters, and hardening (1–2 weeks)
• Key deliverables: CSV/PDF export, role-based access, performance tuning.

10. User stories

10.1. User authentication via SSO

• **ID**: US-001
• **Description**: As a user, I want to sign in via SSO so that access is controlled and I can use my corporate credentials.
• **Acceptance criteria**:
• Users can authenticate using SAML or OAuth providers configured by an admin.
• Role assignment occurs on first sign-in based on admin mapping.
• Unauthorized users are denied access and receive an explanatory message.

10.2. Import feedback from Zendesk

• **ID**: US-002
• **Description**: As a product manager, I want to import tickets from Zendesk so that feedback appears in the insights dashboard.
• **Acceptance criteria**:
• Admin can connect Zendesk with OAuth credentials.
• System imports tickets, deduplicates by ticket ID, and displays them on the dashboard.
• Import summary shows total imported, skipped, and error rows.

10.3. Automatic tagging and sentiment

• **ID**: US-003
• **Description**: As an analyst, I want the system to auto-tag feedback and surface sentiment so I can quickly group similar issues.
• **Acceptance criteria**:
• At least 80% of common themes in sample dataset are assigned an appropriate tag.
• Sentiment (positive/neutral/negative) is shown with each item.
• Users can override tags and the override is persisted.

10.4. Prioritized insights list

• **ID**: US-004
• **Description**: As a product manager, I want issues ranked by impact so I can prioritize what to address first.
• **Acceptance criteria**:
• Issues are ranked by a composite score (frequency × negative sentiment × severity).
• The top-10 list updates within two minutes of a new import.
• Users can pin/unpin items and add follow-up notes.

10.5. Filter and search results

• **ID**: US-005
• **Description**: As a support agent, I want to filter feedback by source, date, and tag so I can find relevant items quickly.
• **Acceptance criteria**:
• Filters can be combined and the list updates in < 1s for typical datasets.
• Search supports keyword and tag-based queries.
• Users can save filter presets.

10.6. Slack alert for critical items

• **ID**: US-006
• **Description**: As a product manager, I want critical items to trigger Slack alerts so my team can act quickly.
• **Acceptance criteria**:
• Admin configures Slack webhook and severity threshold.
• When an item crosses the threshold, a formatted Slack message is posted.
• Alerts include item title, score, and a direct link to the item in the app.

10.7. Export a PDF summary report

• **ID**: US-007
• **Description**: As an executive, I want to download a PDF report of top insights so I can share them in meetings.
• **Acceptance criteria**:
• Users can export the current dashboard view to PDF and CSV.
• PDF includes top issues, counts, trend charts (if available), and export timestamp.
• Exported file is downloadable within 30 seconds for typical dashboards.

10.8. Manual tagging and bulk edit

• **ID**: US-008
• **Description**: As a product manager, I want to manually tag items and bulk-edit groups so I can correct or refine automated tags.
• **Acceptance criteria**:
• Users can select multiple items and apply a tag change in a single action.
• Manual tags are flagged as user-curated and stored separately from auto-tags.

10.9. Role-management by admin

• **ID**: US-009
• **Description**: As an admin, I want to invite users and assign roles so team permissions are controlled.
• **Acceptance criteria**:
• Admins can invite users by email and set role (Admin/Analyst/User/Viewer).
• Role changes take effect immediately.
• Audit log records invite and role-change events.

10.10. Data deletion on request

• **ID**: US-010
• **Description**: As a compliance officer, I want to delete user-specific data on request to comply with privacy laws.
• **Acceptance criteria**:
• Admins can submit a delete request for a user identifier; system removes or redacts PII within SLA (48 hours).
• Deletion events are logged for auditability.

10.11. Read-only REST API

• **ID**: US-011

• **Description**: As an analyst, I want a read-only API to fetch insights so I can build reports or sync with other tools.
• **Acceptance criteria**:
• API supports authenticated GET requests for insights, tags, and exports.
• Rate limit is documented; sample queries available in API docs.

10.12. Import failure handling

• **ID**: US-012
• **Description**: As an operator, I want clear error reporting for failed imports so I can resolve data issues.
• **Acceptance criteria**:
• Import failures generate a human-readable error report with row-level details.
• System retries transient failures and surfaces persistent errors to the admin dashboard.