

osTicket — Business Analysis Specification (Reverse-Engineered)

Scope & version

As of **September 1, 2025**, the latest stable release is **v1.18.2** (released February 4, 2025), with security/bug fixes and **PHP 8.2–8.4** support. Requirements include IIS/Apache + MySQL; language packs and plugins are available. ([GitHub](#), [osticket.com](#))

1) Business context & goals

What it is

osTicket is a helpdesk/ticketing system that consolidates support requests from **email, web forms, and phone** into a multi-user web interface, with routing, SLAs, queues and a user portal/knowledge base to reduce agent load and improve response times. ([docs.osticket.com](#))

Primary goals

- Provide a consistent intake for issues (email piping or mailbox polling) and turn them into trackable tickets. ([docs.osticket.com](#))
 - Route/triage automatically to the right department/agent/topic using **Ticket Filters** and **Help Topics**. ([docs.osticket.com](#))
 - Let agents work efficiently via **Queues/Custom Columns/Quick Filters** and department/team access controls. ([docs.osticket.com](#))
 - Offer **self-service** via the user portal (open a ticket, check status) and **Knowledgebase** content. ([docs.osticket.com](#))
 - Track and enforce **SLAs**; escalate/notify when overdue. ([docs.osticket.com](#), [osticket.com](#))
-

2) Actors & personas

- **End User (Requester):** Submits issues via portal or email; can check status; may be in an **Organization** with optional ticket sharing. (docs.osticket.com, [GitHub](#))
 - **Agent (Support staff):** Works tickets; permissions are role-based per-department; may belong to **Teams** for cross-department collaboration. **MFA** optional/required. (docs.osticket.com)
 - **Department Manager:** Oversees tickets in a department; manages SLAs/queues/canned responses visibility. (docs.osticket.com)
 - **Admin:** Configures system-wide settings (email, API keys, queues, plugins, visibility permissions). (docs.osticket.com)
 - **External System:** Creates tickets via **HTTP API** (ticket-creation only; API key bound to source IP). (docs.osticket.com)
-

3) Core user stories & acceptance criteria

3.1 End User: submit a ticket via portal

Story: As an end user, I can open a new ticket from the portal, choose a **Help Topic**, provide details, and receive a confirmation.

Acceptance

- Given I navigate to the client portal, when I click **Open a New Ticket**, then I see a form for contact details, topic, subject, and message.
- When I submit, then a ticket is created and I see a confirmation/number. (docs.osticket.com)

3.2 End User: submit via email

Story: As an end user, I can email support; the system converts it to a ticket via **piping** (real-time) or **POP/IMAP polling** (scheduled).

Acceptance

- If **email piping** is configured, inbound mail is delivered to `pipe.php` and becomes a ticket.
- If **POP/IMAP** is configured, a **cron/scheduler** triggers `cron.php` to fetch messages; **auto-cron** only runs while a staff member is active. (docs.osticket.com)

3.3 Auto-triage: route new tickets

Story: As an admin, I can define **Ticket Filters** (IF...THEN) to auto-assign department/agent/SLA/topic or reject/auto-reply.

Acceptance

- Filters apply **on ticket creation only**.
- Filters can inspect user data and custom fields; actions include department/agent assignment, canned reply, or rejection. (docs.osticket.com, [osTicket Forum](https://osTicketForum.com))

3.4 Agent: work tickets efficiently

Story: As an agent, I can view tickets in system/global/personal **Queues**, add **Custom Columns**, and **Quick Filters** to focus my work.

Acceptance

- I can create personal queues and adjust columns/sorts; admins can create global queues visible to all agents.
- I can quickly filter tickets by predefined fields from the queue UI. (docs.osticket.com)

3.5 SLA management

Story: As a manager, I can assign **SLA Plans** by **Help Topic** or **Department** to ensure tickets close within specified time; overdue tickets trigger notices.

Acceptance

- SLA is selectable at topic/department; queues and alerts reflect overdue status. (docs.osticket.com, osticket.com)

3.6 Knowledgebase & canned responses

Story: As an agent, I can create **FAQs** and **Canned Responses**; end users can search KBase; canned responses can be department-scoped.

Acceptance

- FAQ entries (attachments/HTML) can be associated with help topics for targeted search; canned responses can be limited by department. (docs.osticket.com)

3.7 Organizations & ticket sharing

Story: As a supervisor, if ticket sharing is enabled for my organization, I can view colleagues' tickets.

Acceptance

- Organization members can optionally **share** tickets; primary contacts may configure sharing. ([GitHub](https://github.com))

3.8 Security hardening

Story: As an admin, I can require **MFA for agents** and ensure API usage is restricted by **IP-bound keys**.

Acceptance

- Toggling “require agents to turn on 2FA” enforces MFA; API calls use **X-API-Key** and must originate from the allowed IP. (docs.osticket.com)

3.9 External ticket creation (API)

Story: As an external system, I can **create tickets** via HTTP JSON with **X-API-Key**.

Acceptance

- Only **ticket creation** and tasks are supported at present (not full read/update). (docs.osticket.com)

4) Business rules

1. Ticket intake

- **Email piping** delivers in real-time to **api/pipe.php**.
- **POP/IMAP** requires scheduled execution of **api/cron.php** (CLI/URL); **auto-cron** fetches only when staff are active/logged in. (docs.osticket.com)

2. Ticket Filters

- Evaluate **only at creation**; may set department/agent/topic/SLA or reject/respond. (docs.osticket.com, [osTicket Forum](https://osticket.com/forum))

3. SLA assignment

- **SLA Plans** may be set by **Help Topic** and/or **Department**. (docs.osticket.com)

4. Access control

- **Roles** grant permissions **per department**; agents have a primary department and can have extended access. **Visibility Permissions** can hide departments, agents, help topics. (docs.osticket.com)

5. Queues

- **Global queues** are admin-defined; **personal queues** are agent-defined; custom columns/sorts supported. (docs.osticket.com)

6. Knowledgebase & canned responses

- FAQs can target help topics; **canned responses** can be limited to a department's visibility. (docs.osticket.com)

7. API

- Only **create** (tickets/tasks) supported; API keys are **IP-bound** (**X-API-Key**). (docs.osticket.com)
-

5) High-level process flows

A. Portal submission → routing → agent response → closure

1. User opens ticket via portal and selects a **Help Topic**. (docs.osticket.com)
2. System creates ticket; **Ticket Filters** (if any) set department/assignee/SLA. (docs.osticket.com)
3. Ticket appears in relevant **Queues**; agent responds (optionally using **canned responses**). (docs.osticket.com)
4. SLA timer tracks due/overdue; alerts/notices fire per configuration; ticket closed or reopened on user reply. (docs.osticket.com, osticket.com)

B. Email intake (piping or POP/IMAP)

1. Email routed via **piping** (real-time) or fetched by **cron/auto-cron**; converted to ticket. (docs.osticket.com)
2. Same routing/queues/SLA steps as above.

C. Organization ticket visibility

1. If **ticket sharing** enabled, organization members can see one another's tickets. ([GitHub](https://github.com))

D. API created ticket

1. External system POSTs to `/api/tickets.json` with **X-API-Key** from an allowed IP; ticket created. (docs.osticket.com)
-

6) Data dictionary (essential entities)

(The project publishes official ERDs for Tickets, Users, SLAs, Queues, etc. below is a BA-level summary; see linked ERDs for full structures.) (docs.osticket.com)

- **Ticket:** number/id, requester (User), Organization, Department, Help Topic, SLA Plan, Status, Priority, Assignee (Agent/Team), Due/Overdue, Subject, Thread (messages, notes), Attachments, Collaborators, Created/Updated. (docs.osticket.com)
 - **User (End User):** name, email (unique), phone; Org link; tickets. (docs.osticket.com)
 - **Organization:** name, members, **ticket sharing** flag/settings. ([GitHub](https://github.com))
 - **Agent:** profile, primary department, extended access, role permissions, MFA settings. (docs.osticket.com)
 - **Department:** name, email/template/SLA defaults; visibility status (Active/Archived/Disabled). (docs.osticket.com)
 - **Team:** cross-department collections of agents (for assignment/visibility). (docs.osticket.com)
 - **Help Topic:** taxonomy used to guide intake, route, associate forms/FAQs; can be nested. (docs.osticket.com)
 - **SLA Plan:** target closure interval (business hours vs schedules), escalations/alerts. (docs.osticket.com)
 - **Queue/Custom Column/Quick Filter:** saved ticket list definitions and UI columns/filters. (docs.osticket.com)
 - **Canned Response:** templated agent replies (optionally department-scoped). (docs.osticket.com)
-

7) Non-functional requirements (NFRs)

- **Security:** MFA for Agents; per-department roles; visibility permissions restricting agents/departments/help topics; API keys **IP-bound**. (docs.osticket.com)
 - **Availability & ingest:** Email intake requires either piping (mail server integration) or scheduled fetch (cron/Scheduled Task/host scheduler) and **does not run if no staff are active under auto-cron**. (docs.osticket.com)
 - **Performance/UX:** Queues/columns/quick filters enable focused views, reducing time-to-action. (docs.osticket.com)
 - **Compatibility:** v1.18.2 supports PHP 8.2–8.4; standard LAMP/WIMP stacks. (osticket.com)
 - **Extensibility:** Plugins for attachments, MFA policies, audit, etc. (see plugin catalog). (docs.osticket.com)
-

8) Out-of-scope, risks & open questions

Known limits / risks

- **API surface** is limited: official docs state **ticket creation only** (and tasks), not full CRUD; this impacts integration/reporting unless you access DB directly or scrape UI. (docs.osticket.com)
- **Email auto-cron** doesn't fetch unless a staff member is active; many installs need a reliable **cron/scheduler** for POP/IMAP fetch. (docs.osticket.com)
- **Help Topic visibility & departmental scoping:** Supported via **Visibility Permissions** for agents; per-department topic visibility in client UI is a recurring forum ask and may require customization. (docs.osticket.com, osTicket Forum)
- **Upgrade path:** The team has been working on a **2.0 rewrite** (React frontend, themeing, REST API) with public roadmap items open; timelines have been fluid. Plan for change management and data migration. ([GitHub](https://github.com))

Open questions for stakeholders

1. **Intake mix:** What % of tickets should come via portal vs email? Is **pipng** feasible on your MTA, or is **POP/IMAP + cron** preferred? (docs.osticket.com)
 2. **SLA policy:** Define tiered SLAs per **Help Topic** and/or **Department**; who owns escalation? (docs.osticket.com)
 3. **Visibility model:** Which departments/agents/help topics should be hidden using **Visibility Permissions**? (docs.osticket.com)
 4. **Organization sharing:** Should org-wide ticket sharing be ON by default? Who is the primary contact? ([GitHub](https://github.com))
 5. **Queues:** Which **global queues** do we pre-define (e.g., **Overdue High-Priority, VIP Org, New-Last-24h**)? (docs.osticket.com)
 6. **Security posture:** Require **MFA** for all agents? API usage behind reverse proxy (ensure **X-Forwarded-For** is correct). (docs.osticket.com)
-

9) Appendix — Install/operate highlights

- **Download/requirements** (v1.18.2; PHP 8.2–8.4; IIS/Apache, MySQL). (osticket.com)
- **Email setup:** Choose **pipng** (real-time) or **POP/IMAP** (cron-driven); understand **auto-cron** trade-offs. (docs.osticket.com)
- **Key admin surfaces:** Admin Panel (Settings/Emails/Agents/Manage), Agent Panel (Queues, Knowledgebase). (docs.osticket.com)
- **Datamodel references:** Official **ERDs** for Tickets, Users, SLAs, Queues etc. (docs.osticket.com)