IT Glue

WE ❤ DOCUMENTATION

Blueprints

Table of Contents

Escalation Rules 3

Incident Management 5

Problem Management 11

Risk Management 15

Service Desk Process Flow 18

Response Recovery 20

Escalation Rules

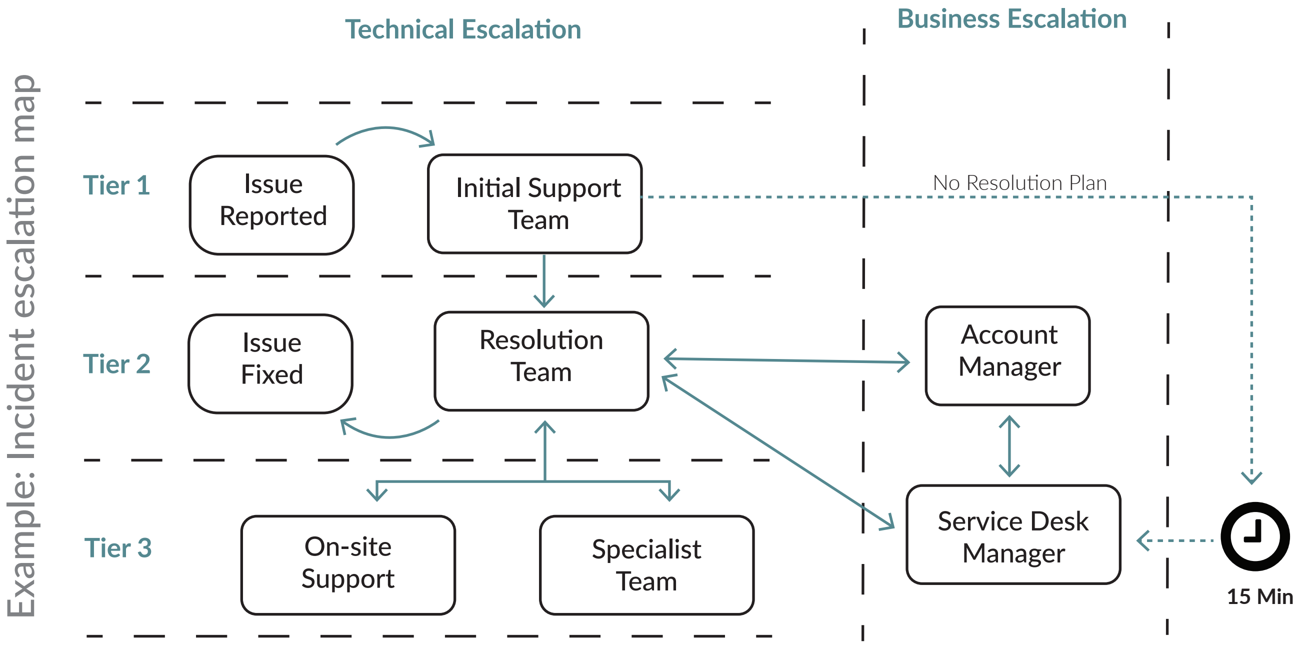
You may have defined escalation levels in the form of an escalation ladder, for example:

- Tier 1 Technicians
- Tier 2 Technicians
- Tier 3 Technicians
- On-site Support
- Service Desk Manager (decision point)

Generally speaking, first-line support is provided by tier 1 technicians, second-line support provided by tier 2 technicians, and third-line support provided by tier 3 technicians. The smaller the organization, the fewer tiers you will have.

Note that issues can be escalated to a particular level within the hierarchy, when needed. For example, a tier 1 technician can immediately escalate to the service desk manager if the technician knows they lack the administrative permissions needed to fix a P1 incident.

The service desk manager is escalated to when there is a decision to be made. The decision point could also be an account manager depending on the decision and your structure. The purpose of having a decision point is to address the issue with the client. Typically, this is a conversation with the client about making a purchasing decision, such as replacing a device or ordering a part, in order to continue resolving the issue.



Escalation triggers

To escalate a ticket to a higher-tiered technician, there needs to be rules or guidelines that define the escalation triggers.

A large number of tickets will start with a tier 1 technician. In many cases, the tier 1 technician can resolve the request given the right amount of time. However, in this business, the amount of time you have is dictated by many factors including SLAs.

Normally, the escalation between the levels of the hierarchy are triggered by some combination of:

- **time trigger** - the ticket wasn't completed within some pre-determined timeframe that's within the target resolution time defined by the SLA.
- **complexity or severity** - complex or severe incidents will require more expertise and authority and may be immediately escalated.

Other possible triggers:

- **client satisfaction** - the decision to escalate may be based on client satisfaction issues.
- **external vendor** - the ticket needs to be escalated to an external vendor.
- **administrator privileges** - the ticket needs to be escalated to someone with admin level access.

Important:
Make sure your tier 1 staff know the amount of time they can spend on a ticket before asking for help or passing the issue up to the next support tier. As soon as it becomes clear that they are not able to resolve the issue or when target times for 1st level (2nd level, 3rd level, etc.) resolution are exceeded, the ticket must be escalated.

Incident Management

This document describes a process for handling incidents and applies to all incidents and all clients.

Goals

The goal of incident management is to restore normal service operation within SLA limits and as quickly as possible using workarounds or solutions.

Out of scope

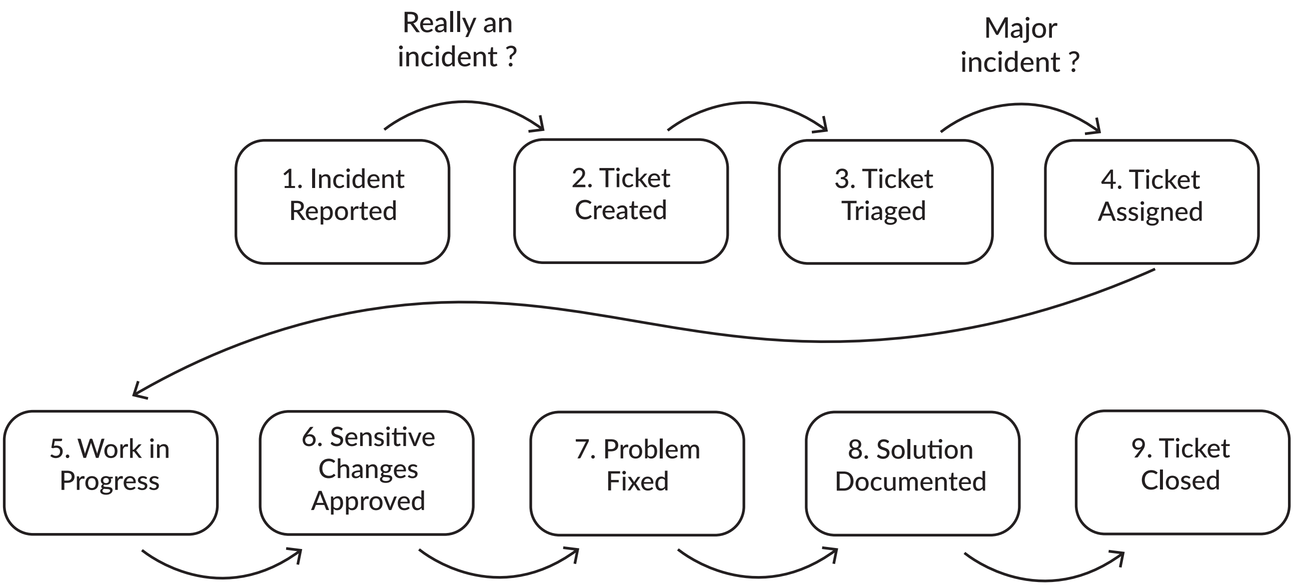
If it's a request for new or altered services

- then it's the responsibility of the Request Fulfillment process

If the cause of a reoccuring issue needs to be investigated at the same time

- then the root cause analysis is the responsibility of the Problem Management process

Process overview



Procedure Steps

1 Incident reported

Someone identifies and reports an incident.

Examples of incidents:

- System-down (server issue)
- Network issues
- Automatic alert
- Printer not printing
- Service failing
- Application bug

2 Ticket created

The service desk receives the report and creates a ticket in a PSA or ticketing system.

3 Ticket triaged

The incident is triaged (prioritizing and updating key ticket fields) to assist whoever works on the ticket in taking swift and effective action.

The assignment of priority is especially important, as it will determine how quickly the issue will be resolved. Major incidents cause serious interruptions of business activities and must be solved with greater urgency.

The priority assignment should consider:

- How quickly does the service need to be restored?
- What is the business impact of the incident? In most cases, a user who can't log in to a minor system due to a forgotten password will be a lower priority than a server that crashes at the start of the business day.

Note that if you allow your technicians to pull tickets from a queue, they will need to do the initial triaging on pulled tickets.

4 Ticket assigned

Technician experience levels may fall into different tiers. When a relatively common ticket of minimal complexity arrives, it's reasonable to assign the ticket to a tier 1 technician. The tier 1 technician will then escalate to a tier 2 or higher level technician if they are unable to resolve the ticket.

Other factors to consider when assigning tickets:

Expertise - Assign phone system related requests to the technician who has handled more phone system assignments.

Access to technology and information - Make sure the technician has the required software/hardware, system permissions, and knowledge base access.

Availability - Use the ticketing system and electronic calendars to check on workload and availability.

5 Work in progress

The goal is fast recovery of the disrupted IT service.

Request and review information (screenshots, knowledge base, event logs) to find a solution or workaround.

Escalate the ticket when an escalation will allow for a much more positive experience for the client (e.g. a business critical function can return to operation) or when target times for resolution by first level support are exceeded.

Define exactly who owns the ticket when it's escalated. The ticket owner is not necessarily the person currently working on solving the ticket.

Also, document the escalation reason. Because an escalation is inherently expensive, it's important to explain what prompted it.

Potential reasons for escalations:

- Technical skill limitation
- Permission/security/sensitive
- SLA or time exceeded
- Scope approval required

6 Sensitive changes approved

If the fix requires sensitive changes, including security changes and changes that could have a direct impact on IT services (rebooting a business critical server), the appropriate managers are consulted. The work is assessed and approved before continuing.

7 Issue fixed

When there is a solution or workaround, perform the necessary steps to fix the issue. Some fixes will require testing and deployment before the issue can be considered fixed.

8 Solution documented

Document the solution in the ticket. You may want to also document the solution so that it can be searched inside your knowledge base. If there is existing documentation, make sure it's up to date before closing the ticket.

9 Ticket closed

Record any final items and mark the ticket solved.

Leave the ticket in a solved status for another day or longer to make sure the issue is resolved and allow the client the option to re-open the ticket.

At the end of this period, close the ticket. At the same time, the client should receive an email that asks them to rate their experience.

Note that resolved incidents, service performance, response times, and re-opened tickets should get looked at as part of a reporting process.

Business rules

- All staff must receive training on the workflow for sensitive changes and requests.
- All staff must receive training on the workflow for handling unsupported (out of scope of the managed service agreement) requests.
- All incidents must be tracked using tickets.
- All client and internal correspondence related to the incident must be recorded in the ticket.

- Each time the ticket is worked on, the status and categorization must be reviewed and updated.

- As soon as the target resolution time for first level support is exceeded, the incident is escalated to a tier 2 or tier 3 technician.

- All escalations to on-site support must be approved by the service desk manager.

- Only ticket owners (the person responsible for handling a specific ticket through its lifecycle) are allowed to mark tickets completed/solved.

Problem Management

This document describes a process for handling reoccurring incident-related problems. This process applies to all clients. A problem is defined as the cause of one or more incidents.

Goals

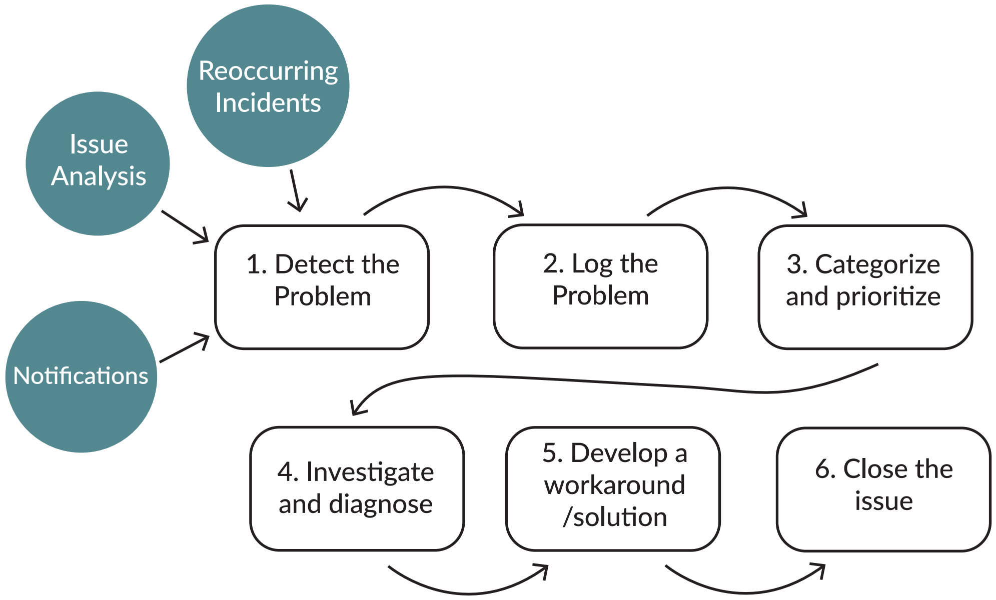
Problem Management is a methodology designed to prevent problems and resulting incidents from happening, to eliminate reoccurring incidents, and to minimize the impact of incidents that cannot be prevented.

Process overview

Problem Management consists of activities used to diagnose root causes of incidents and determine resolutions to these problems. It ensures that resolutions are implemented through appropriate control procedures such as Change Management and Release Management.

The process also ensures that information about problems and appropriate workarounds are documented in order to reduce the number of incidents over time.

All workarounds/resolutions are recorded. This information is used to help identify permanent solutions as well as provide quick temporary fixes while solutions are being implemented.



Procedure Steps

1 Detect the Problem

Problems may be raised through escalation or through analysis of incident patterns and alerts. Signs of a problem include incidents that repeat after successful trouble-shooting and incidents not resolved at the service desk.

2 Log the Problem

Log all information in a **problem ticket**. Information should include:

- Time and date of occurrence
- Incident details (from Incident Management)
- Service Details
- Equipment Details
- Related Incidents
- Symptoms
- Previous fixes (diagnostic and recovery actions)
- Problem Category

3 Categorize and prioritize the Problem

Use the same methodology as Incident Management. Prioritization is determined by user impact and urgency. Urgency is based on how quickly the client requires a solution. Impact is a measure of the extent of potential damage that might be caused. Priority determines how soon the problem is investigated.

4 Investigate and diagnose

Investigate to determine the root cause of the issue. Once the root cause is determined, it becomes a Known Issue. Log all actions in the Problem Record.

5 Develop a workaround/solution

Create a workaround. The workaround enables the Service Desk to restore service to users while the problem is being resolved. The workaround should always be considered temporary.

6 Post the workaround and close the issue

Post the known issue and workaround to a Known Issue Database (sometimes called a Known Error Database). This allows the service desk to search for issues and workarounds to quickly close related incidents.

Review

Review the problem. This is a team activity designed to prevent future problems. The Problem Management Team evaluates the problem documentation to identify what happened and why. This is where having a Problem ticket works much better than trying to pull together details from memory. This outcome of this process should result in improved processes, staff training, and complete documentation.

Problem Management outputs

- Known issues
- Requests for Change (RFCs) through Change Management
- Updated Problem Ticket (including solutions and workarounds)
- Closed Problem Tickets (Archive)
- Known Issues Database (KIDB)
- KB content to use in Incident Management
- Management Reports

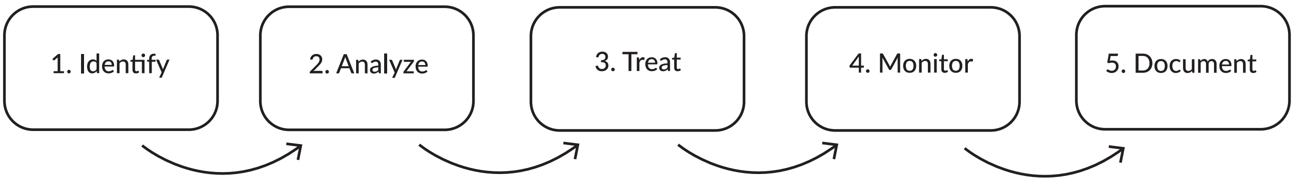
Risk Management

This document describes a process for handling risk. This process applies to all clients.

Goals

The goal is to identify and document risks and then put the proper controls in place to reduce or eliminate them. Risk Management is an ongoing process, but is especially important during an initial client walk-through so important decisions can be made early on.

Risk Management Process Overview



Procedure Steps

1 Identify the Risk

Identify and document the following:

- Assets
- Threats
- Existing and planned security measures
- Vulnerabilities
- Consequences
- Related business processes

Examples include:

- Guests can access the client's network
- Client isn't taking basic precautions, like keeping servers dry and cool
- Client isn't backing up critical data on a regular basis
- Client is using weak passwords
- Client has old and unsupported hardware on the network

2 Analyze the Risk

This step provides the basics for evaluation and treatment.

- Determine the cause and source of risk
- Determine the likelihood of the risk occurring
- Determine the consequences of the risk if it does occur

Classify the risk such as essential risk, monitored risk, observe the risk.

- Managed Risk - A treatment should be in place to manage or change the risk. Default for Low, Med, High measured risk.
- Monitored Risk - Controls should be in place to monitor potential risks or further signs of risk. No treatment at this time.
- Observed Risk - No treatment, but potential risks are observed on a regular basis

Document the assessment in a way that records a risk profile that:

- helps identify risk priority
- captures reason for decision on tolerable and intolerable exposure
- allows all involved to see the full profile
- helps in review and monitoring

3 Treat the Risk (Controls)

Risk treatment involves selecting one or more options to modify or control the risk. These include:

- **Transfer** - Share the risk with another party or parties.
- **Avoid** - Avoid the risk by deciding not to start or continue an activity that presents the risk.
- **Treat** - Change the likelihood of the risk or change the consequences (Impact)
- **Accept** - Retain the risk by informed decision. For example, taking or increasing the risk indefinitely or until another time.

The Treat option may be broken down into additional controls:

- **Preventive** - Limits the possibility of a threat.
- **Corrective** - Controls the threat.
- **Directive** - Ensures that a particular outcome is achieved.
- **Detective** - This is only possible after an event has occurred and a loss has occurred.

The overall purpose of a control is to restrain risk, but seldom does it eliminate it.

4 Monitor the Risk

Regular monitoring ensures that controls are effective and efficient. Monitoring is also required to ensure that additional vulnerabilities are not introduced.

5 Document the Risk

Clearly identify ownership of the risk.
In addition to identifying risk ownership:

- Record any historical decisions and consequential actions.
- Identify key milestone dates where risk can increase. For example, software becomes unsupportable, hardware warranties expire.
- Identify key deadlines when risks must be removed.

Documentation should be reviewed regularly as new risks or controls emerge.

Business rules

- All risks must be conveyed to the client as soon as discovered.
- All risks must be documented with the risk treatment clearly identified and accepted by the client.

Service Desk Process Flow

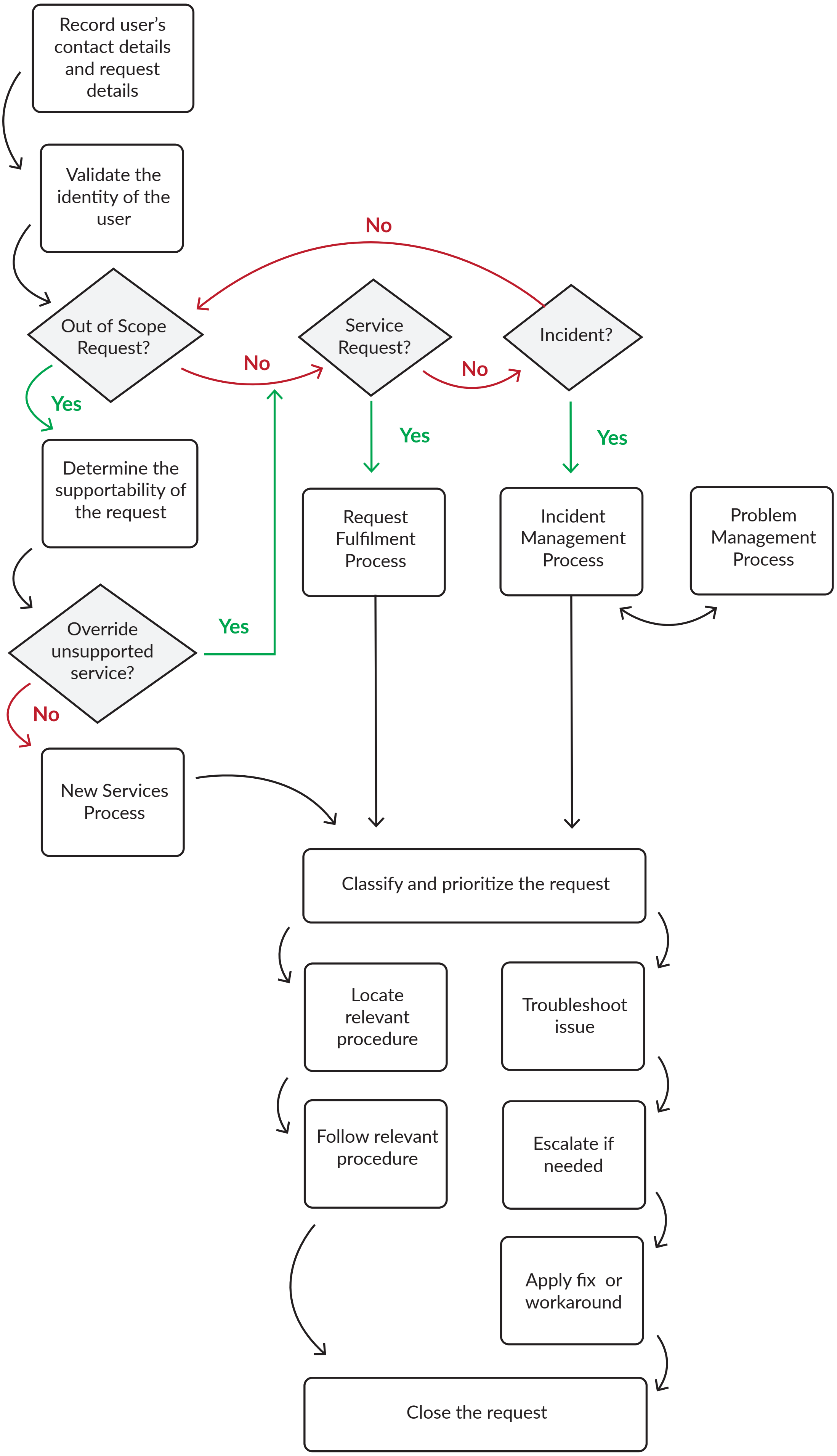
The service desk process flow provides an overall structure for handling user requests.

When a user contacts the service desk, the service desk:

- Records the user's contact information and the details of the request.
- Validates the identity of the user.
- Categorizes, classifies, and prioritizes the request.
- Resolves the user's request.
- Closes the request.

The diagram on the next page illustrates the service desk process flow

Process Flow



Record user's contact details and request details

Validate the identity of the user

Out of Scope Request?

Service Request?

Incident?

No

Yes

No

No

Yes

Yes

Determine the supportability of the request

Request Fulfilment Process

Incident Management Process

Problem Management Process

Override unsupported service?

Yes

No

New Services Process

Classify and prioritize the request

Locate relevant procedure

Troubleshoot issue

Follow relevant procedure

Escalate if needed

Apply fix or workaround

Close the request

ITGlue

WE ♥ DOCUMENTATION

Library Preview

Response and Resolution Times

A key task in service level management is defining service level targets, such as response and resolution times.

Response times

Response time refers to how quickly you respond to technical issues. The higher your staffing levels, the more likely it is that you can promise a response within “x” minutes.

If you provide 24/7 on-call support, you will have to factor this into your promise. A best practice is to define priority levels and stipulate that only the highest priority (Priority 1) issues qualify for 24/7 support.

Priority levels example:

Priority 1 = complete criticial service outage, no realistic workarounds

e.g. internet/network down, email outage, LOB application outage

Priority 2 = critical service severely degraded, significantly affecting business, workarounds available

e.g. workstation hard drive failure, printer down, internet/network slowness

Priority 3 = minimal impairment of operational performance

e.g. workstation slowness, install/update an application, install a printer, add user permissions

Make sure that you clearly define which issues are only responded to during business hours, and how this is calculated in the response time. For example, if operating hours are 7am to 6pm, Monday to Friday, and a call is logged at 5:45pm on a Friday evening, then a response at 7:05am on the following Monday is a 20 minute response time.

You also need to define what a “response” means. Is it the first completed phone call? The first email from the service desk? This definition would not usually include the auto-response that comes from the ticketing system when the ticket is created.

Maximum resolution times

Resolution time refers to how long it takes from the time an issue is logged until it is fully resolved.

As with response times, the usual practice is to establish a priority classification system and assign a target resolution time to each priority, noting which ones are eligible for 24/7 support and which ones are only calculated based on business hours.

Offering fixed resolution times takes careful consideration because they require very defined written processes around ticket statuses, as well as a resolution time calculation that excludes any time spent “waiting.” Ticket statuses that stop the clock typically include: pending client input, pending vendor response, and pending parts.

Example: Response and resolution times by priority

The following table provides a high-level summary of the response and resolution time objectives for support services provided under this agreement:

Priority	Desription	Response	Resolution
1	Critical services are **down** and work cannot be completed by organization with no realistic workarounds	30 Minutes	2 Hours
2	Critical services are **severely degraded**, significantly affecting business operations with some workarounds available	30 Minutes	4 Hours
3	Operational performance is **impaired** while most business operations remain functional	30 Minutes	8 Hours

Important! Make sure you only agree to targets that are achievable. For example, a 30 minute response time for high priority issues requires adequate staffing to make this possible. Service level targets provide the opportunity to manage client expectations and protect your business.

For more on this topic, check out our April 11th blog article: How MSPs Achieve Great Service Level Agreements.

Uptime Provision

Some MSPs offer uptime guarantees of up to 99.9% (equals 43.2 minutes of unplanned downtime each month) for hosted network services or other business continuity services.

This may be acceptable to many clients but be prepared to pay for this guarantee.

You may want to add a caveat to the SLA that uptime guarantees only apply after “x” days. Or if upgrades are needed, agree to bring the client infrastructure up to date before putting the uptime guarantee into effect.