# Service Request Management Process

| | |
|---|---|
| Document name | Process_Design_Document_SRM_V0.1.docx |
| Version | 0.1 |
| Status | ☒ In progress ☐ Approved ☐ Validated |
| Update date | 2025/05/02 |
| Owner | |
| Level of confidentiality | ☐Public ☒Internal ☐Limited ☐Confidential |

Document Control

CHANGE HISTORY:

| AUTHOR(s) | VERSION | DATE | CHANGES |
|---|---|---|---|
| Alvin MERRE | V0.1 | 2025/05/02 | Document creation |
| | | | |
| | | | |

APPROVAL

| ROLE | NAME | DATE |
|---|---|---|
| Team Lead | Nicolas FRANCOIS | |
| | | |

# Table of Contents

**1. Introduction** **4**
1.1. Main definition 4
1.2. Objectives, challenges & benefits 5
1.3. Actors 6
**2. Process description** **9**
2.1. Process trigger, inputs, outputs 9
2.2. Process activities 10
2.3. Link with other ITSM process 11
**3. Process Workflow** **13**
3.1. Service Request Opening 13
3.2. Service Request Approval 16
3.3. Service Request Fulfillment 17
3.4. Service Request Closure 21
**4. Roles & responsibilities** **23**
**5. Metrics & KPIs** **25**
5.1. Efficiency & SLA Compliance 25
5.2. Authorization & Security 26
5.3. User Experience & Satisfaction 27
**6. Governance** **28**
6.1. Governance objectives 28
6.2. Operational Committee 29
6.3. Continuous Improvement Committee 30
**7. Tools & Deliverables** **31**
7.1. Tools 31
7.2. Tickets, work orders and task status 32
7.3. Deliverables 33

# 1. Introduction

## 1.1. Main definition

**<u>A *Service Request*</u> is a formal request from a user or an authorized representative that initiates a service action which is predefined and considered a standard part of service delivery.**

Service requests are a critical component of the user experience and typically include:

- A request to perform a service action (either by the service provider or in collaboration with the user),
- A request for access to a resource, system, or application,
- A request for information,
- Submission of feedback, compliments, or complaints.

*<u>Common examples of Service Requests include:</u>*

- *Request to create or reset a password,*
- *Request for access to a shared folder or business application,*
- *Ordering IT equipment (e.g., laptop, mobile phone, screen),*
- *IT onboarding for a new employee (account creation, equipment delivery, access rights),*
- *Request for a physical access badge or building access,*
- *Inquiry about an internal process or policy,*
- *Submission of feedback or dissatisfaction regarding an IT service.*

These requests follow **standard, predefined, and approved workflows** to ensure consistency, efficiency, and traceability. While some requests involve simple tasks (e.g., providing information), others, like onboarding, may be more complex and require coordination across multiple teams or tools. Many service requests may involve standard changes to services or components.

So, **<u>Service Request Management</u>** is the process of managing a service request through its life cycle from submission through delivery & follow up. It consists of defining the underlying workflow & processes that enable a service request to be reliably submitted, routed, approved, monitored and delivered. It is a single point of entry for requesters to create, view and manage their requests.

## 1.2. Objectives, challenges & benefits

**The purpose of service request management practice is to handle efficient user requests and customer inquiries.**

**To achieve this, the process must:**

- Provide a clear and accessible channel for users to submit requests,
- Ensure accurate and timely communication regarding the status of service request tickets,
- Close or escalate requests in alignment with user confirmation and expectations,
- Meet service level targets for responsiveness and quality of user interactions,
- Accurately classify, articulate, and route requests to the appropriate service line or support team.

Implementing an effective Service Request Management process brings **significant value to the business**:

- Enables quick and efficient access to standard services, enhancing user productivity and improving the quality of business services and products
- Reduces administrative overhead by simplifying the process for requesting and receiving services, which also contributes to lower service delivery costs
- Improves control over service delivery through a centralized fulfilment function
- Supports cost optimization by enabling centralized negotiations with suppliers
- Helps reduce support costs through standardization and streamlined handling of recurring requests

## 1.3. Actors

### 1. Delivery Lead

The Delivery Lead is accountable for the overall design, performance, and continual improvement of the Service Request Management process. He is usually a senior manager with the ability and authority to ensure the process is rolled out and used by all stakeholders.

He is responsible for:

- Defining and maintaining the process framework, roles, and policies
- Ensuring alignment with overall ITSM strategy and governance standards
- Monitoring process performance and ensure it delivers expected value
- Identifying areas of improvement and lead process optimization initiatives
- Validating major process evolutions, KPIs, and governance models
- Supporting the Service Request Process Manager and ensure coordination with other ITSM practices
- Representing the process in audits and executive-level discussions

*ITIL role: Service Request Process Owner*

### 2. Service Request Process Manager

The SRM Process Manager, represented by **iQo**, is responsible for the day-to-day facilitation of the global Service Request Management process. They ensure the quality and consistency of the process and lead key governance activities at the global level.

They are responsible for:

- Governing the end-to-end execution of the Service Request Management process
- Leading and facilitating SRM governance meetings

- Managing the overall planning and lifecycle of service request categories and workflows
- Reviewing complex or high-impact service requests retrospectively
- Monitoring KPIs and driving continuous improvement of the request handling process
- Escalating or arbitrating in case of process exceptions or operational blockers
- Maintaining and updating process rules, service catalog entries, and related documentation
- Ensuring that user experience and communication standards are consistently applied

### 3. End user

The **end user** or their representative is the person who initiates the service request. They express a need and submit the request through an appropriate channel (e.g., services catalog via BMC Helix or phone).

*ITIL role: Requester*

### 4. Help Desk

The **Help Desk** plays a guiding role by assisting users who are unfamiliar with the service catalog or the request creation process. Their objective is to foster user autonomy by showing users how to navigate the catalog and submit their own requests. If needed, they must take remote control of the user's workstation to demonstrate the process or even submit the request on the user's behalf. They act as facilitators, helping users formulate and submit their service requests correctly, but do not process or fulfil the request directly.

*ITIL role: Support L1*

### 5. Leader Team, Delivery Managers, Service Owners or Business Representatives

Certain requests may require validation before execution, especially when involving access to sensitive systems, financial implications, or compliance concerns. Approvers can be **Leader Team, Delivery Managers, Service Owners, or Business Representatives** accountable for granting or denying such approvals.

*ITIL role: Approver*

**Service Owners also ensure that the request is properly managed throughout its lifecycle**. They are coordinators, they monitor progress, ensure the request follows the correct process steps, engages the appropriate fulfilment teams, and guarantees the final delivery meets the expected service quality.

*ITIL role: Request Coordinator*

### 6. Technical teams

Technical or operational teams are responsible for executing the service request once it has been approved. They can be **internal** (e.g., IT operations, infrastructure, workplace support) or **external** (e.g., third-party service providers or suppliers), depending on the nature of the service and the sourcing model in place.

*ITIL role: fulfilment teams*

# 2. Process description

## 2.1. Process trigger, inputs, outputs

1. **Triggers -** *What initiates the process ?*

The Service Request Management process is triggered by a user request.

This request can be submitted through two main channels:

- **Service Portal:** via BMC Helix Digital Workplace, where users can browse the service catalog and submit requests directly.
- **Phone Support:** by calling the help desk, which assists the user and logs the request on their behalf.

2. **Input -** *What formally enters the process ?*

A **Service Request** created and submitted in the ITSM tool (BMC Helix), containing all necessary information for evaluation and processing.

Typical input elements:

- Requested service or item from the catalog
- Description and justification of the request
- Desired fulfillment window
- Attached supporting documents or details
- User contact and contextual information

3. **Outputs -** *What does the process deliver ?*

*A **consistent and traceable fulfillment of service requests**, ensuring timely delivery of value to users while maintaining service quality and efficiency.*

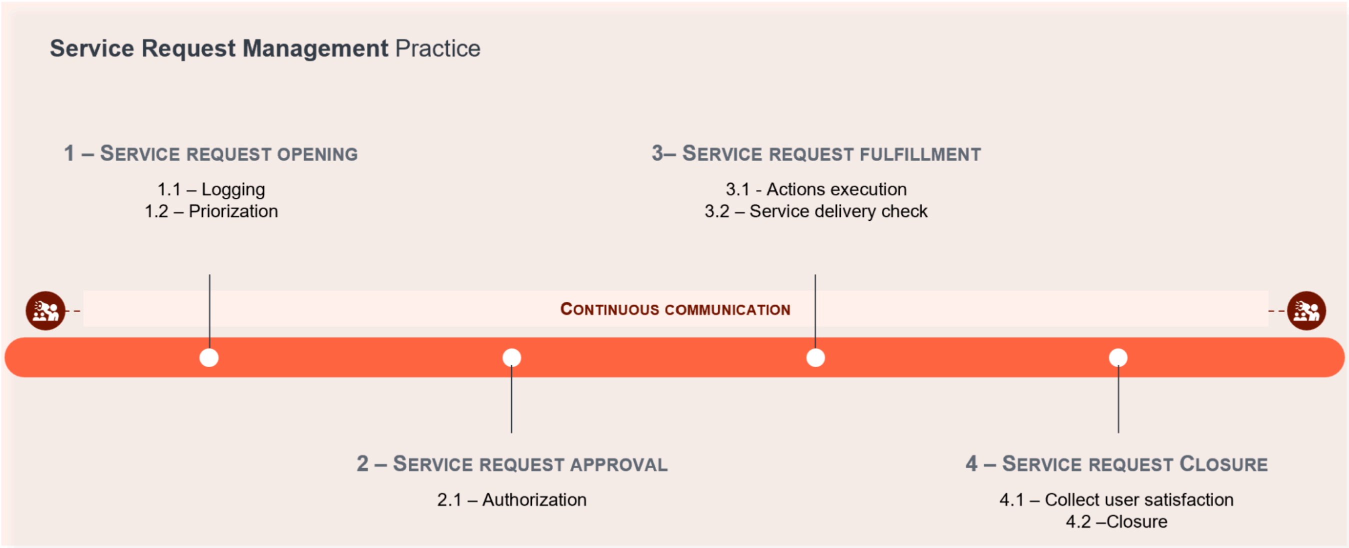
Examples of outputs:

- Fulfilled and closed service requests
- Communication and coordination with the requestor
- Satisfaction survey results or feedback collected
- Documentation updates in the CMDB or knowledge base (if applicable)

## 2.2. Process activities

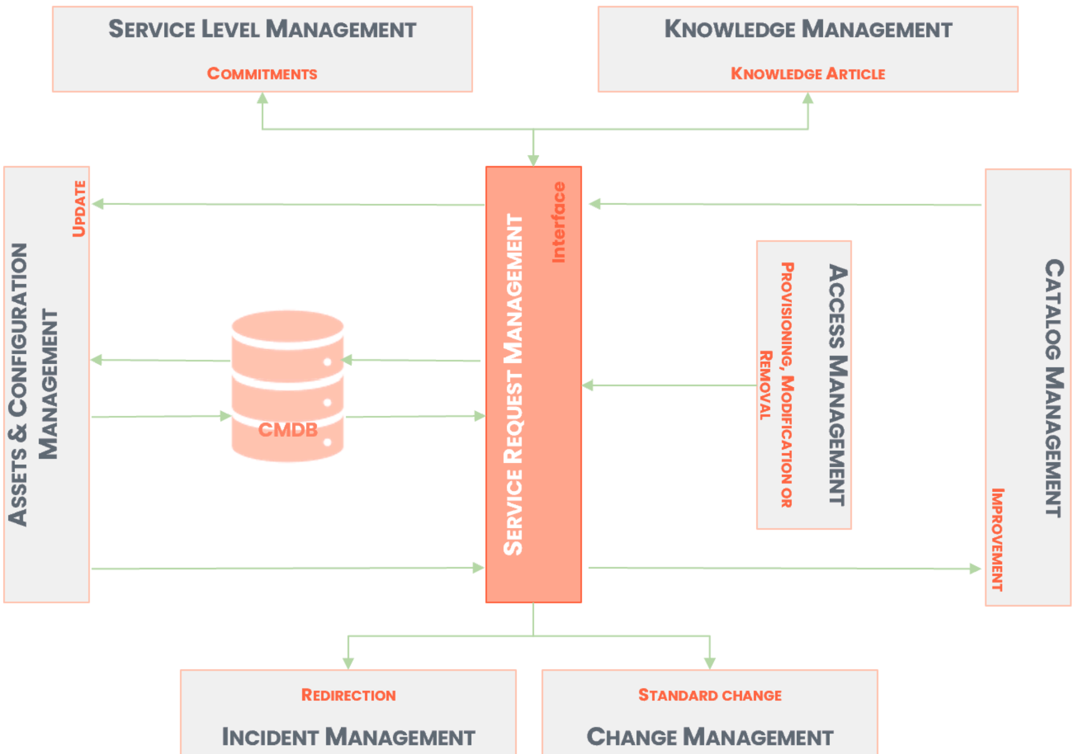
The process is structured around four main stages:

- **Service Request Opening:** the request is logged and prioritized. This step ensures the request is captured correctly and assigned an appropriate level of urgency.
- **Service Request Approval:** when required, the request goes through an approval workflow to ensure its legitimacy and alignment with policies or budget constraints.
- **Service Request Fulfillment:** approved requests are executed by the relevant fulfilment team(s). Actions are tracked, and service delivery is verified to ensure expected outcomes are achieved.
- **Service Request Closure:** the process concludes with the collection of user satisfaction feedback (when applicable) and formal closure of the request, confirming that the user's need has been addressed.




## 2.3. Link with other ITSM process

The Service Request Management process is closely connected with the following ITSM practices:




- **Incident Management**: to redirect or reclassify requests that are actually incidents (e.g., unplanned service interruptions).

- **Access Management**: for requests involving access provisioning, modification, or removal.

- **Change Management**: when fulfilling requests requires standard, low-risk, and pre-authorized changes to services or infrastructure.
  In a future version of the Service Request Management process, a stronger integration with the Change Enablement process will be necessary and particularly for IT-for-IT requests that involve predefined technical modifications. These types of requests often require the execution of low-risk, repeatable changes governed by the change process. It will be important to implement mechanisms that allow the identification and tracking of such requests. This includes monitoring the number of related changes executed per month as well as measuring the success rate of these changes, in order to ensure reliability and continuous improvement in service delivery.
- **Asset & Configuration Management**: to ensure that all assets and configuration items (CIs) involved in the request are correctly identified, tracked, and up to date, enabling accurate and efficient fulfilment.

- **Service Catalog Management**:
  - Provides the interface and structure through which users can submit and categorize service requests.
  - Is continuously improved by identifying recurring or high-volume requests that should be **standardized and formally included** in the service catalog.

- **Knowledge Management**: to support the request process with relevant documentation, self-service content, and resolution guidance for users and support teams.

- **Service Level Management**: defines the expected response and resolution times for each type of service request, ensuring measurable and transparent delivery aligned with user expectations.

- **Continual Improvement**: provides insights and analysis to optimize the performance of the service request process and the quality and relevance of available services.

# 3. Process Workflow

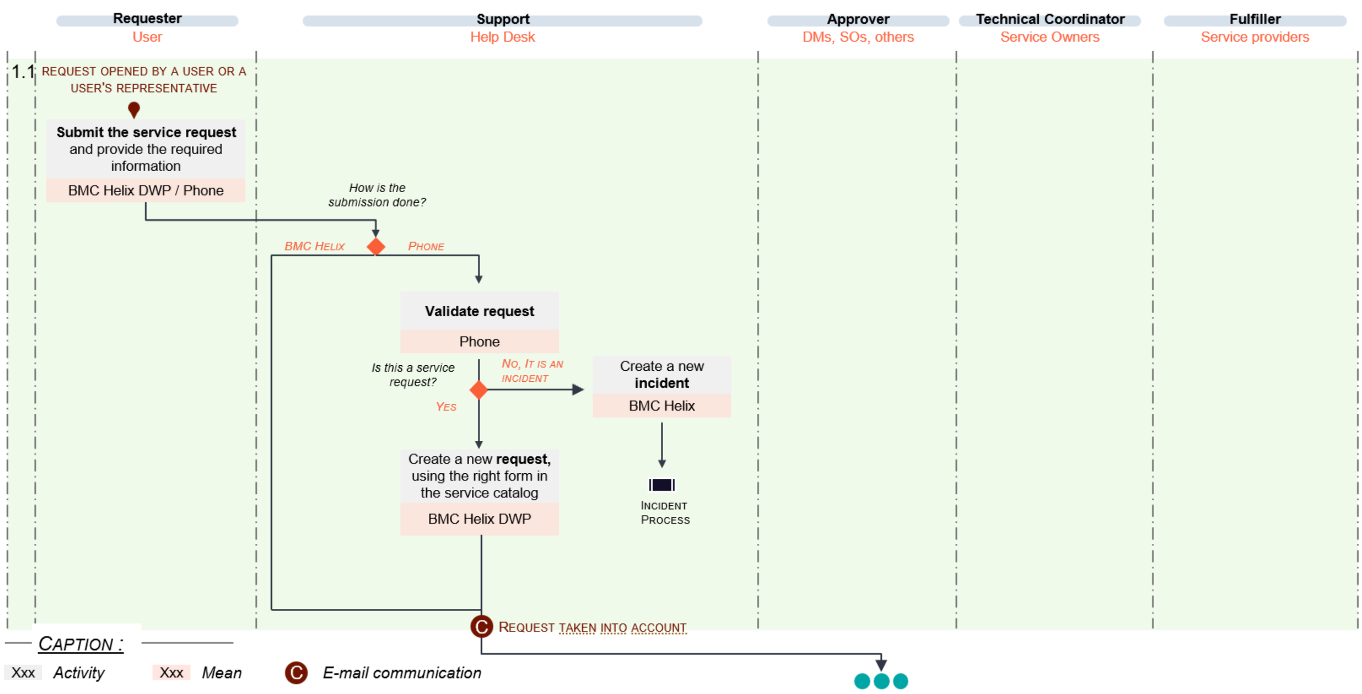
## 3.1. Service Request Opening

The **Service Request Opening** activity is composed of two key sub-steps that ensure the request is properly initiated and managed from the outset:

- **Logging** the request in the ITSM tool
- **Prioritizing** the request based on predefined criteria

These sub-activities lay the foundation for accurate routing, timely processing, and appropriate handling throughout the lifecycle of the request.

### 3.1.1. Logging




This step corresponds to the **initial capture and registration of the service request** in the ITSM system. It can occur through two channels:

- **Via Syra (the BMC Helix Digital Workplace portal)**
  The user independently submits a service request by selecting the appropriate form from the service catalog. **<u>The request is automatically logged and categorized into the ITSM tool</u>** with the relevant details provided by the user.

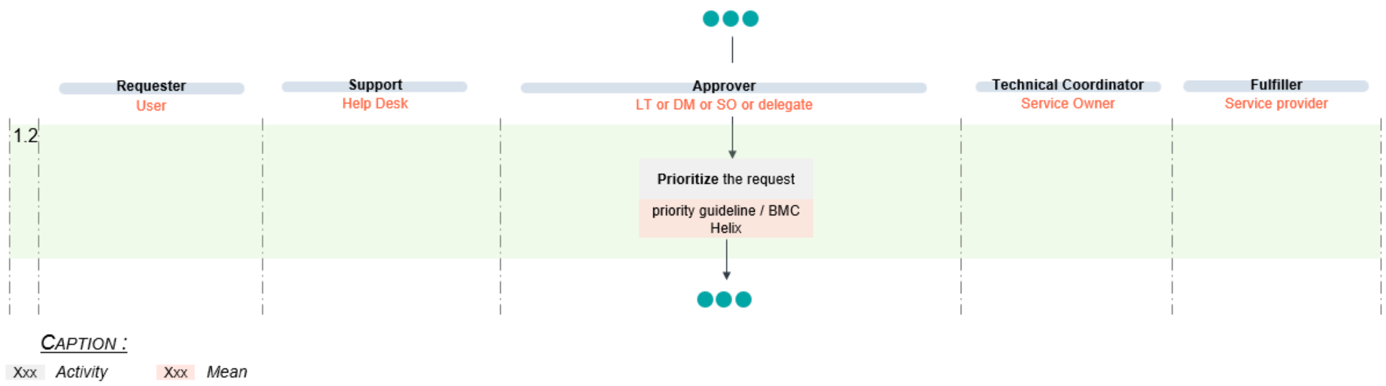
- **Via phone through the help desk**
  If the user contacts the help desk by phone, the support agent first **validates the nature of the request**.
  - If it qualifies as a service request, the agent must create the request **using the correct form in Syra (DWP)**, and **not via the back-end** or ticketing module directly. **<u>This ensures proper categorization and alignment with catalog-driven workflows</u>**.
  - If the request is actually an incident (e.g., a service malfunction), it is redirected to the **Incident Management process**, and an incident is logged instead.

This logging activity ensures that all service requests are accurately captured, traceable, and aligned with predefined catalog entries, allowing for standardized processing and prioritization.

Once the service request is created, whether submitted directly by the user or logged by the help desk, **a confirmation email is sent to the user**, acknowledging that the request has been successfully submitted and taken into account.

Following this, the request is **assigned to the person or team identified to validate it**, if an approval step is required.

### 3.1.2. Prioritizing




Once the service request has been logged, it is **automatically or manually prioritized** to ensure timely and appropriate **execution of the request** and **delivery of the requested service**.

Each team is free to organize the processing of these requests as it sees fit. However, below are **a few guidelines that may help** the team optimize the order in which requests are processed **in their respective backlogs**.

Prioritization could be based on two main criteria:

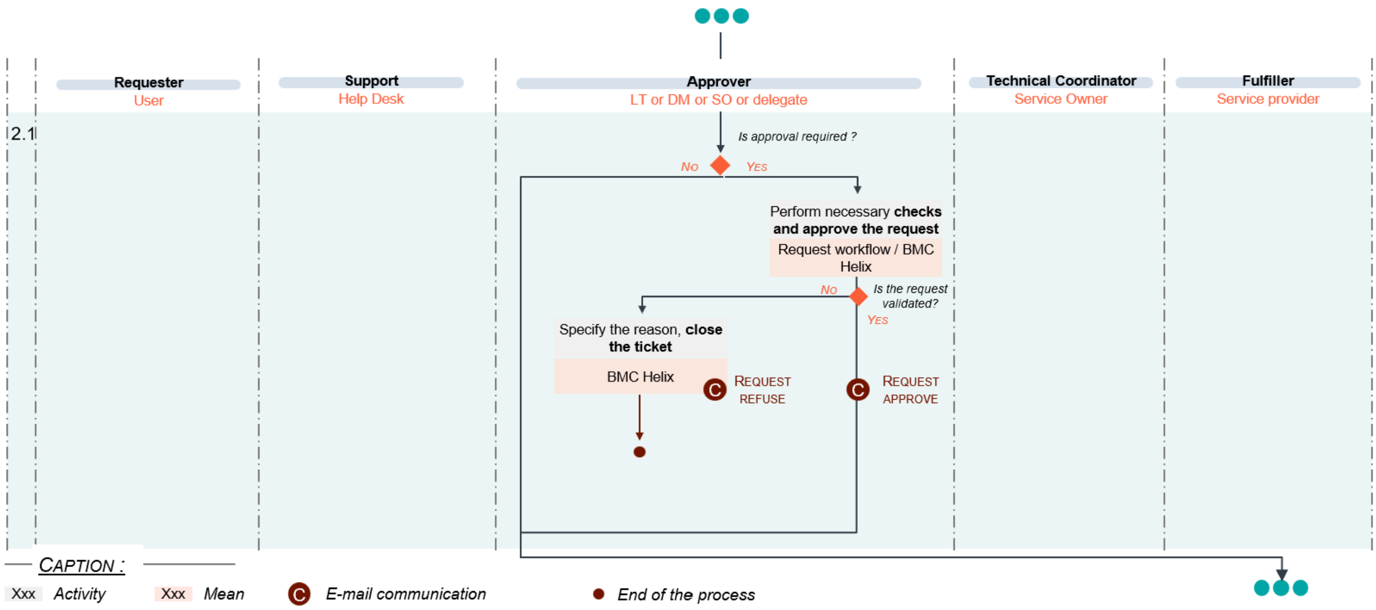
1. The **SLA defined in the service catalog**, which sets the standard timeframes for processing and fulfilment of each request type
2. The **business criticality** of the request, which is evaluated using the following indicators:
   - The SLA itself, which already reflects a baseline level of importance
   - The **VIP status** of the requester, if applicable
   - The **target delivery date** specified by the user. When feasible, execution is aligned with the urgency expressed by the user.

     *For example, if two users request a laptop on the same day but provide different desired delivery dates, the request with the earlier target date will be prioritized, assuming no constraints prevent this.*

This approach ensures that services are delivered in line with operational priorities and user expectations.

## 3.2. Service Request Approval

### 3.2.1. Authorization




Once the request has been logged and prioritized, the process checks whether **approval is required**, depending on the nature of the request.

There are several **types of approval** that can be configured:

- **Auto-approved / No approval required**
- **Single user approval**, such as a direct manager or line manager
- **Group approvals** (**recommended**) for stronger governance

If **multiple approvers or groups** are involved, approvals can follow two possible modes:

- **Sequential**: approval from the second group is triggered only after successful approval from the first group
- **Parallel**: approval requests are sent to multiple groups simultaneously

If **no approval is required**, the request moves directly to the fulfilment phase.

If **approval is required**, the identified approver (e.g., line manager, delivery manager, service owner, or delegate) receives a notification and is responsible for:

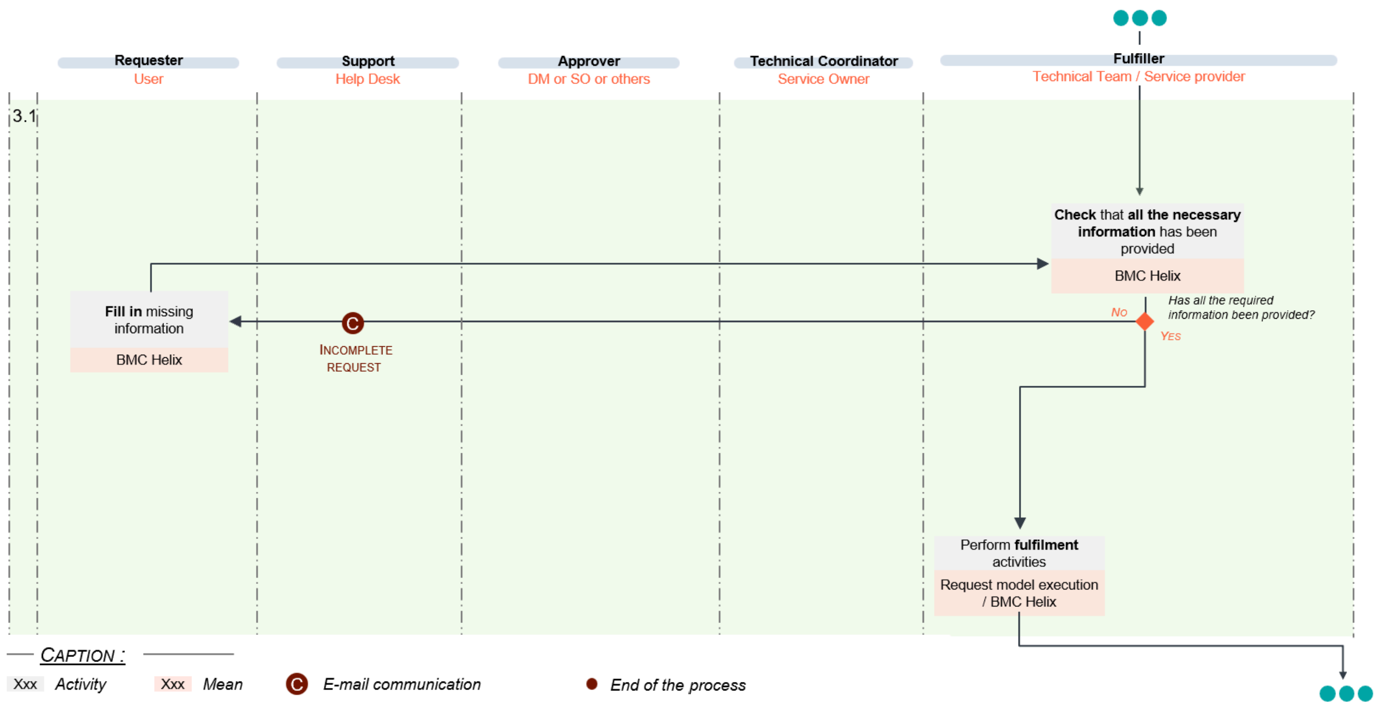
- **Reviewing the request** and performing necessary checks
- **Approving or rejecting** the request within the BMC Helix workflow

If the request is **approved**, it moves forward to the execution phase. If the request is **rejected**, the approver must specify the reason, and the request is closed in the system. In **both cases**, an **email notification is sent to the user** to inform them of the status change (approval or rejection).

## 3.3. Service Request Fulfillment

### 3.3.1. Actions execution

Once the request has been approved (or if no approval is required), it enters the **execution phase**, where the requested service or action is delivered following the predefined request model.




This phase is structured around three operational layers: **Service Request**, **Work Orders**, and **Tasks**, which provide a scalable and traceable framework for fulfilment.

1. **Service Request**

The **Service Request** tracks the entire lifecycle of the user's demand. It may result in the creation of one or more **Work Orders**, depending on the components involved. The request is only considered complete when **all associated Work Orders have been successfully closed**.

## 2. Work Orders

**Work Orders** manage specific fulfilment activities linked to the Service Request. They are:

- **Automatically created** based on the request type
- Or **manually triggered** from other ITSM processes (e.g., Incident, Problem) when additional actions are required

Work Orders may be executed **in parallel or in sequence**, depending on the process design. Each one includes a **description field** for the fulfilment team to document progress, inputs received from the user, and any clarifications.

Work Orders can be **linked to other Work Orders or ITSM records** to reflect dependencies, although these relationships do not trigger automated actions.

Work Order priority is **set to "low" by default**, but can be redefined by the **Service Owner** depending on business criticality.

## 3. Tasks

Each Work Order can include **multiple Tasks**, which represent the detailed execution steps.

- Tasks may be assigned to **different support groups**, enabling distributed or parallel fulfilment.
- A Work Order is automatically marked as completed when **all its associated tasks are closed**.

## 4. Execution Oversight and Coordination

The **Service Owner** acts as a **conductor (orchestrator)** of the execution phase. They **monitor the overall progress** of the request, ensuring that tasks and work orders are progressing according to plan, and that no blockers are preventing delivery.

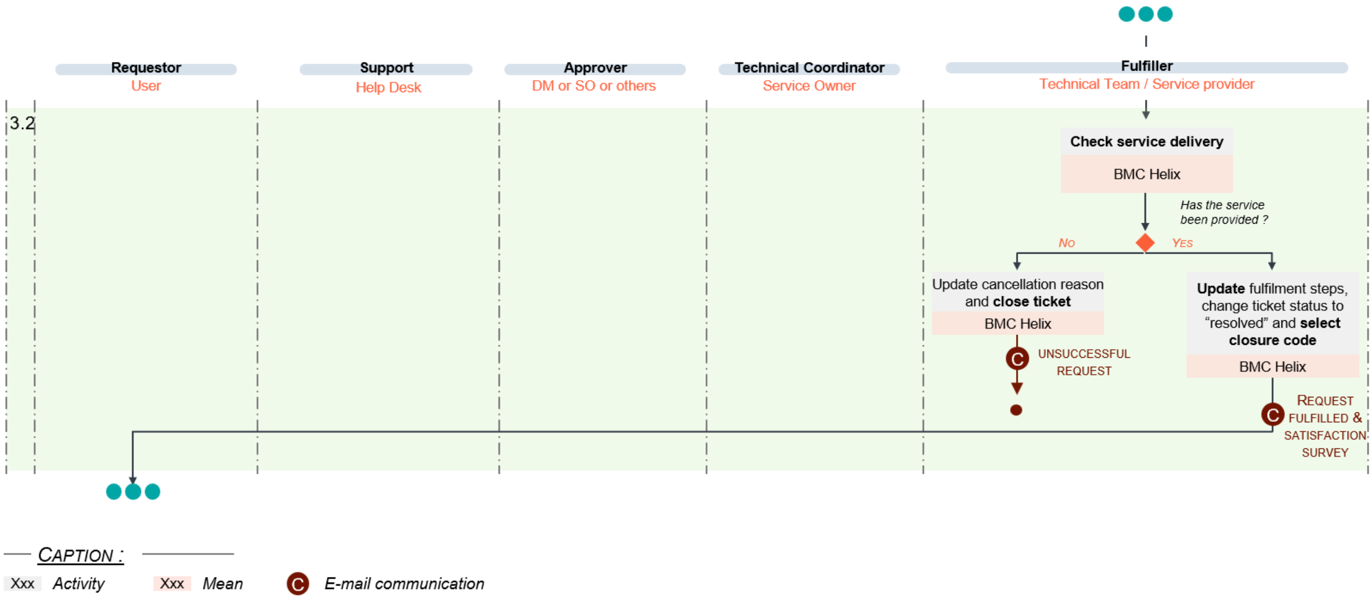
Before execution begins, the fulfilment team verifies that **all required information is present** to proceed.

- If **information is complete**, execution starts immediately.
- If **information is missing**, an **automatic notification** is sent to the requester via BMC Helix, asking them to provide the missing details. Execution is **put on hold** until the data is completed.

This layered and coordinated execution model ensures that services are delivered efficiently, with a high degree of visibility, process integrity, and alignment with user expectations.

### 3.3.2. Service delivery check

After the fulfillment actions have been completed, the process enters the **service delivery verification phase**. This step ensures that the requested service has been delivered as expected and in accordance with the defined requirements.




The **fulfilment team** (internal or external) **ensures that the service is actually delivered** and whether all fulfilment steps are complete in BMC Helix.

- If the service has been **successfully delivered**, the fulfiller:
  - Updates the fulfilment details in the system
  - Selects the appropriate **closure code**
  - The request is marked as **fulfilled**, and the process continues toward final closure
  - **An automatic e-mail notification is sent to the requester to inform them that the service has been delivered and the request is closed**

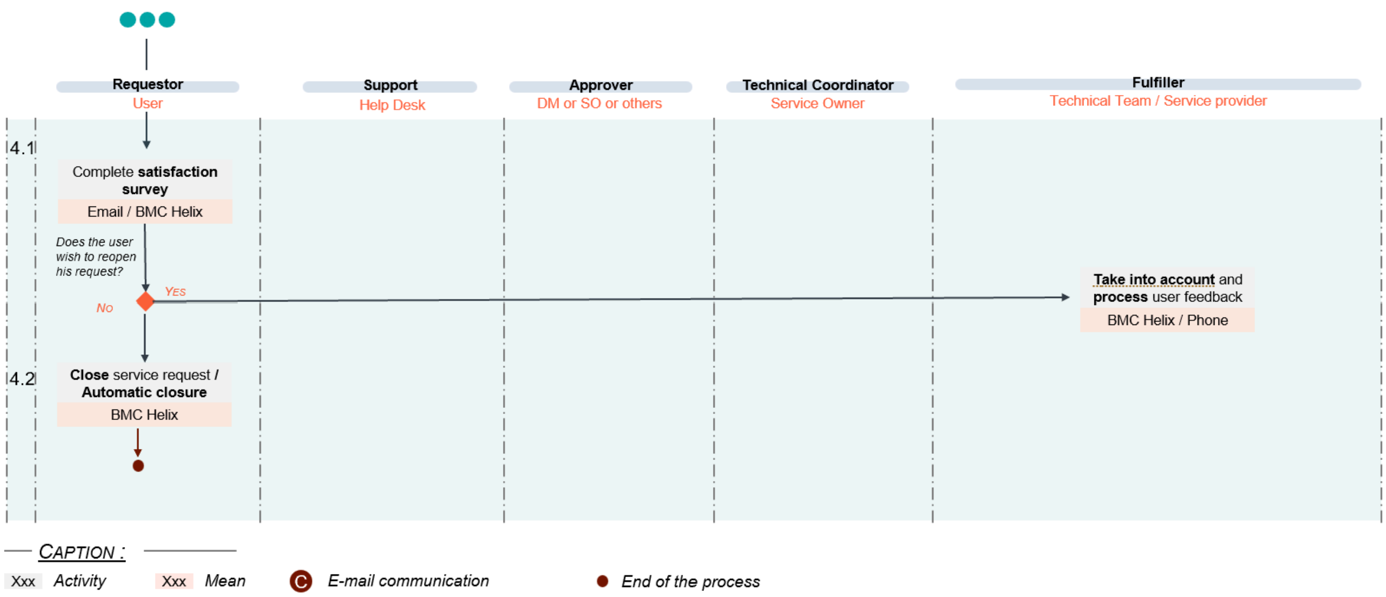
- If the service **could not be delivered** (e.g., technical issue, user no longer needs it), the fulfiller:
  - Documents the **reason for cancellation** in BMC Helix
  - **Closes the ticket** accordingly, marking it as an **unsuccessful request**
  - **An automatic e-mail notification is sent to the requester to inform them of the cancellation and the reason for it**

*Concrete ways to check service delivery may include:*

- *Verifying that the expected system access has been granted (e.g., by testing login or validating permissions)*
- *Confirming hardware delivery and installation with proof of delivery or a signed handover document*
- *Ensuring that software has been deployed and is operational (via test or user feedback)*
- *Requesting user confirmation (by email, ticket comment, or phone) that the service meets expectations*
- *Checking that all related tasks and work orders are closed, and no blocking items remain*

This step is essential for ensuring data quality, tracking completion accuracy, and supporting clear communication with the user regarding the outcome of their request.

## 3.4. Service Request Closure




### *3.4.1. Collect user satisfaction*

Once the service request has been fulfilled, the requester receives an **automatic notification inviting them to complete a satisfaction survey**, either by email or via BMC Helix.

This survey allows the requester to rate their overall experience and provide qualitative feedback. It typically includes:

- A 5-smiley scale to rate the experience
- A free text field for additional comments
- If the rating is **3 to 5**, the user is encouraged with a positive prompt (e.g., "Great! Tell us what you liked.")
- If the rating is **1 or 2**, a comment is **mandatory**, with a prompt such as "Please tell us what went wrong and how we can improve."

**By default, 100% of completed service requests trigger a survey**, unless specific conditions are configured to deactivate it (for instance, in the case of **generic requests**, e.g., automated or low-value actions such as password resets, or **redirected requests**, e.g., requests that are handled outside the main ITSM tool or transferred to another department like HR or Security).

The survey is a key feedback mechanism shared across ITSM and Business Workflow processes (e.g., Incidents, Service Requests) to support **continuous service improvement** and measure **user satisfaction**.

### *3.4.2. Closure*

After the request enters the **"Completed" state**, a **5-working-day grace period** begins during which the requester can **reopen the ticket** if they consider the request unsatisfactorily fulfilled.

- If the requester is satisfied, no action is needed, and the ticket will be **automatically closed** at the end of the grace period.
- If the requester is not satisfied, they may **reopen the ticket within the 5-day window**. A blank work order titled **"Reopen – root cause analysis"** is generated and assigned to the **last support group that handled the request**, which will determine the next steps.
- If no action is taken by the requester, the request is **transitioned to "Closed"** automatically.

Once in the **"Closed" state**, the request **cannot be reopened**. Any new issues or needs must be addressed through the creation of a **new Service Request**.

This controlled closure mechanism ensures clarity, avoids ambiguous re-openings, and gives users time to confirm their satisfaction before finalizing the request.

# 4. Roles & responsibilities

| | **Requester** (End User) | **Support** (Help-Desk) | **Approver** (LT, DM, SO or other) | **Coordinator** (Service Owner) | **Fulfiller** (Technical Team) |
|---|---|---|---|---|---|
| ***1. Service Request Opening*** | | | | | |
| ***Logging, if the requester submits his request directly in BMC Helix*** | | | | | |
| Create the new request via BMC Helix DWP | R,A | I | I | I | I |
| ***Logging, if the requester calls the Help Desk to submit his request*** | | | | | |
| Validate request / Redirect to incident process | C | R,A | - | - | - |
| Create the new request via BMC Helix DWP | C | R,A | I | I | I |
| ***In both cases,*** | | | | | |
| Inform the user that request has been taken into account | I | R,A | - | - | - |
| ***Prioritizing,*** | | | | | |
| Prioritize the request | C | - | R,A | I | I |
| ***2. Service Request Approval*** | | | | | |
| Perform necessary checks (if necessary) and approve the request | I | - | R,A | C | I |
| Update request ticket information | - | - | R,A | - | - |
| Inform the user that his | I | - | R,A | - | - |

| | | | | | |
|---|---|---|---|---|---|
| request has been refused or approved | | | | | |
| ***3. Service Request fulfilment*** | | | | | |
| ***Actions execution*** | | | | | |
| Check that all the necessary information has been provided | I (if incomplete) | - | - | I | R,A |
| Perform fulfilment activities | I | - | - | I | R,A |
| ***Service delivery check*** | | | | | |
| Check service delivery | - | - | - | C | R,A |
| Update request ticket informations | - | - | - | I | R,A |
| Inform the user that his request is fulfilled or unsuccessful | I | - | - | I | R,A |
| ***4. Service Request Closure*** | | | | | |
| ***Collect user satisfaction*** | | | | | |
| Complete satisfaction survey | R,A | I | - | I | I |
| ***Closure*** | | | | | |
| Close service request | C | - | - | I | R,A |

*Responsible (**R**), Accountable (**A**), Consulted (**C**), Informed (**I**)*

# 5. Metrics & KPIs

To monitor the effectiveness of the Service Request Management process and ensure it delivers both operational performance and user satisfaction, **three practice success factors (PSFs)** have been identified.

- **Efficiency & SLA Compliance** – to ensure timely and accurate fulfilment of service requests in line with defined targets
- **Authorization & Security** – to guarantee that only approved and compliant requests are fulfilled
- **User Experience & Satisfaction** – to maintain high levels of service quality, responsiveness, and communication

Each PSF is supported by a set of specific **Key Performance Indicators (KPIs)** that enable regular measurement, process control, and continuous improvement.

## 5.1. Efficiency & SLA Compliance

This success factor focuses on ensuring that service requests are processed swiftly, reliably, and in full compliance with defined service level agreements (SLAs). **Requests must be fulfilled efficiently and on time, in line with agreed service level targets**

| KPI | Description | Calculation mode |
|---|---|---|
| **Total number of service requests processed** | Total number of service requests processed over the previous period, by service type | Total number of service requests processed during the selected period |
| **Service request breakdown** | Breakdown of service requests by status (logged, in progress, closed, etc.) and by service type | Count of requests by status (Logged, In Progress, Closed, etc.) and by service type |
| **Average processing time** | Average processing time per service request, by request type | Σ (Processing time for requests of a given type) / Number of requests of that type |

| | | |
|---|---|---|
| **% of service requests completed within SLA** | Number and percentage of service requests completed within SLA target times | (Number of requests completed within SLA / Total number of requests) × 100 |
| **First contact resolution rate** | Percentage of requests resolved by the service desk at first point of contact | (Number of requests resolved at first point of contact / Total number of requests) × 100 |
| **% of requests fulfilled remotely or via automation** | Number and percentage of requests fulfilled remotely or through automation | (Number of remotely or automatically fulfilled requests / Total number of requests) × 100 |
| **% of misclassified or reassigned requests** | Number and percentage of misclassified or reassigned requests | (Number of misclassified or reassigned requests / Total number of requests) × 100 |
| **Standard vs. specific request ratio** | Ratio of standard/recurring vs. specific requests | Number of standard requests / Number of specific (non-standard) requests |
| **Total number of specific requests submitted** | Total number of specific (non-standard) requests submitted | Total count of non-standard service requests submitted |

## 5.2. Authorization & Security

This factor ensures that the fulfilment process remains secure and compliant by validating that only properly approved requests are executed.

| KPI | Description | Calculation mode |
|---|---|---|
| **Percentage of service requests properly authorized before fulfilment** | Measures the proportion of service requests that received the required approval before being fulfilled | (Number of requests fulfilled with required approval / Number of requests requiring approval) × 100 |

## 5.3. User Experience & Satisfaction

This factor measures how well the process meets user expectations and ensures that service quality and communication remain at a high level.

| KPI | Description | Calculation mode |
|---|---|---|
| **User satisfaction score based on closure surveys** | Measures user satisfaction with the service received, based on feedback collected at closure | Average score from user satisfaction surveys submitted after request closure |
| **Percentage of service requests reopened after closure** | Tracks the proportion of requests that were closed but had to be reopened due to incomplete or incorrect fulfilment | (Number of reopened requests / Total number of closed requests) × 100 |
| **Total number of incidents caused by request fulfilment activities** | Counts the number of incidents directly caused by service request fulfilment actions | Total count of incidents linked to fulfilment actions over the selected period |
| **Current size of the backlog of pending or delayed service requests** | Represents the number of requests that are still pending or delayed beyond expected processing times | Count of open service requests that are past their expected resolution time or marked as delayed |

# 6. Governance

## 6.1. Governance objectives

The governance of the Service Request Management process aims to ensure not only operational efficiency but also alignment with business needs and continuous improvement. Its main objectives are:

- **Ensure efficient processing in line with service level commitments**:
  Guarantee that all requests are handled according to defined procedures, within agreed timeframes and by the appropriate stakeholders.
- **Identify recurring, non-standardized requests that could be added to the service catalog**:
  Detect repetitive, unstructured requests that could benefit from formalization and inclusion in the catalog to streamline handling.
- **Analyze the evolution of request volumes and types**:
  Monitor trends to better allocate resources, adjust priorities, and identify optimization opportunities.
- **Maintain a high level of user satisfaction**:
  Track the perceived quality of service, the clarity of communication, and the relevance of provided solutions.
- **Define and implement continuous improvement actions**:
  Leverage insights from monitoring activities to initiate concrete, targeted improvements across the process and service offering.

This governance framework enables a structured and proactive approach to managing service requests, fostering operational excellence and user-centric delivery.

To support the effective governance of the Service Request Management process, two dedicated committees have been established: an **Operational Committee** and a **Continuous Improvement Committee**. Each serves a specific purpose, cadence, and set of objectives to ensure both short-term control and long-term process evolution.

## 6.2. Operational Committee

<table>
<tr><td colspan="3">Operational Committee</td><td>Frequency</td><td>Biweekly</td><td>Duration</td><td>60 min</td></tr>
<tr><td>Participants</td><td>• Process Owner<br>• Service Desk Manager<br>• Support Teams</td><td>Objectives</td><td colspan="4">• Monitor process execution<br>• Identify operational issues<br>• Define and follow up on corrective actions</td></tr>
<tr><td colspan="7">Agenda</td></tr>
<tr><td colspan="7">• Review of past activity over the last two weeks (volume of requests by status and type, compliance with SLAs, notable user feedback)<br>• Identification of blocking points (unprocessed requests, recurring problems, non-standard requests, technical or organizational obstacles, team capacity limitations, tool or workflow malfunctions, etc.)<br>• Definition of corrective actions<br>• Follow-up on previously defined actions</td></tr>
<tr><td colspan="3">Input data/Elements to be prepared</td><td colspan="4">Output data/Expected results</td></tr>
<tr><td colspan="3">• Activity report (volume, types, SLA performance)<br>• List of overdue or abnormal tickets<br>• Significant user feedback<br>• Status of ongoing corrective actions</td><td colspan="4">• List of corrective actions to implement<br>• Operational performance insights<br>• Team alignment on short-term priorities<br>• Items to escalate to the Governance Committees</td></tr>
</table>

## 6.3. Continuous Improvement Committee

<table>
<tr><td colspan="3">Continuous Improvement Committee</td><td>Frequency</td><td>Monthly</td><td>Duration</td><td>90 min</td></tr>
<tr><td>Participants</td><td><ul><li>Process Owner</li><li>Continual Improvement Manager</li><li>Help Desk</li><li>Support Teams</li><li>Service Owners</li></ul></td><td>Objectives</td><td colspan="4"><ul><li>Analyze trends and performance data to identify improvement opportunities</li><li>Improve user experience and service quality through standardization and automation</li><li>Monitor progress on improvement initiatives</li></ul></td></tr>
<tr><td colspan="7">Agenda</td></tr>
<tr><td colspan="7"><ul>
<li>Presentation of monthly activity (Volume by type, identification of trends, associated impacts, backlog size) 15 min</li>
<li>Analysis of compliance with service level agreements (SLAs) (Percentage of requests fulfilled within target timeframes, identification of discrepancies, root cause analysis) 15 min</li>
<li>Analysis of user satisfaction and associated irritants (Survey results, % of reopened requests, pain points, incident feedback from users) 15 min</li>
<li>Analysis of operational quality KPIs (Requests resolved at first contact, misclassified or reassigned requests, % of automated or remotely fulfilled requests) 10 min</li>
<li>Review of compliance and security-related KPIs (% of authorized requests, number of security-related incidents caused by fulfilment) 5 min</li>
<li>Identification of recurring requests to be standardized and/or automated (Focus on high-volume, low-complexity requests to reduce manual effort and improve speed) 15 min</li>
<li>Follow-up and update of the improvement action plan (Status of previous actions, blockers, new opportunities, and CSI register update) 15 min</li>
</ul></td></tr>
<tr><td colspan="3">Input data/Elements to be prepared</td><td colspan="4">Output data/Expected results</td></tr>
<tr><td colspan="3"><ul><li>Monthly KPI dashboards (including from operational committee)</li><li>SLA reports</li><li>User satisfaction survey results</li><li>List of recurring pain points escalated from the operational committee</li><li>Candidate requests for automation or standardization</li><li>CSI register (open and proposed actions)</li></ul></td><td colspan="4"><ul><li>Updated CSI register with prioritized actions</li><li>List of improvement initiatives with owners and deadlines</li><li>Decisions on requests to be standardized or automated</li></ul></td></tr>
</table>

# 7. Tools & Deliverables

## 7.1. Tools

To ensure consistency, traceability, and efficiency in request handling, the Service Request Management process relies on two main tool ecosystems: **BMC Helix** (ITSM platform) **and Google Workspace**. These tools support all steps from request submission and categorization to fulfilment coordination, tracking, and performance monitoring.

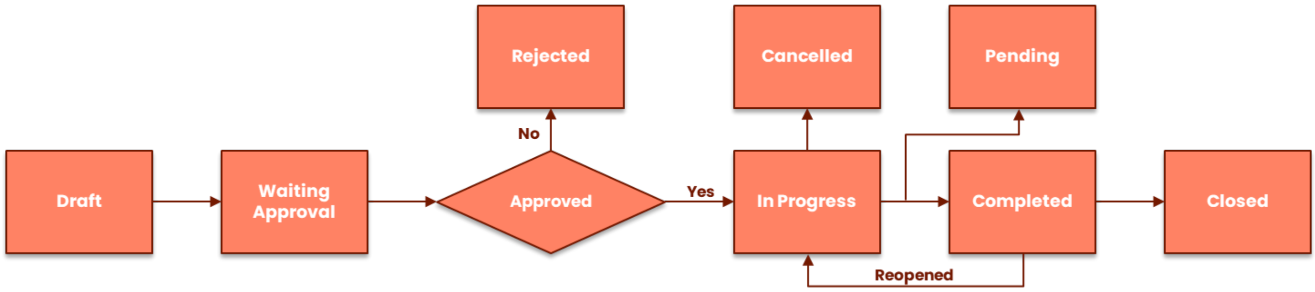
The table below outlines the tools and modules used, grouped by typology, with their respective functions throughout the SRM lifecycle.

| Typology | Tool/Module | Description of use in the process |
|---|---|---|
| **ITSM Tool** | BMC Helix – Service Request Module | Submission, categorization (standard/specific), approval workflow, fulfilment tracking, and closure |
| | BMC Helix – CMDB | Linking requests to CIs for impact awareness and reporting consistency |
| | BMC Helix – Knowledge Module | Access to support articles and procedures to enable self-service and assist fulfilment teams |
| | BMC Helix – Reporting/KPI Dashboards | Monitoring SLA compliance, fulfilment times, first contact resolution, backlog, and satisfaction rates |
| **Google Suite** | Google Sheets / Slides | Used to consolidate KPI tracking, backlog reviews, and prepare SRM process governance reviews |
| | Google Drive | Centralized storage of user guides, process documentation, and request templates |
| | Google Chat / Gmail | Coordination between fulfilment teams and process owners, and communication of critical request updates |

## 7.2. Tickets, work orders and task status

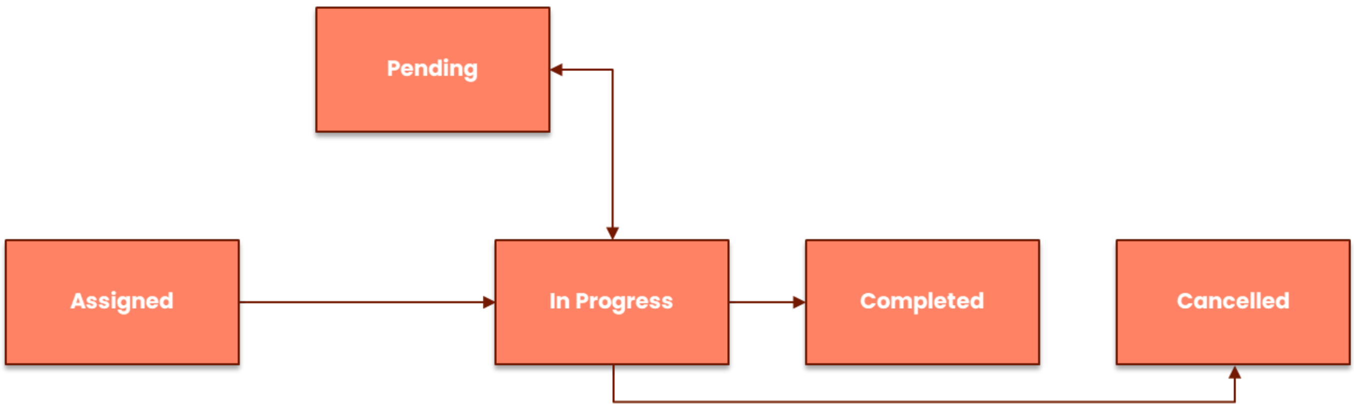
### 7.2.1. Service request ticket status

The diagram below represents the standard lifecycle of a service request ticket managed in the ITSM tool **BMC Helix**. It illustrates the different statuses a ticket may go through — from creation to approval, fulfilment, resolution, or cancellation — until final closure. This structured workflow ensures traceability, proper validation, and consistent communication throughout the entire process.



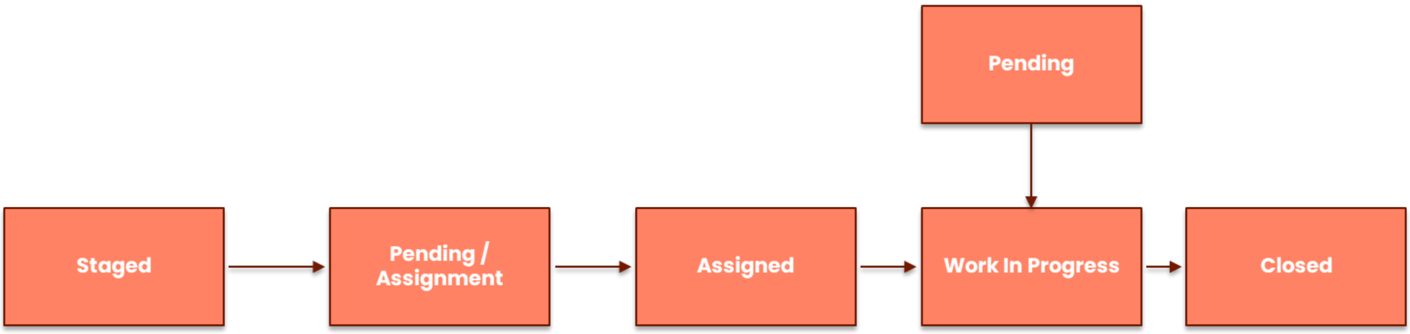

### 7.2.2. Work order status

The diagram below illustrates the lifecycle of a **work order** in Syra. It shows the different statuses a work order can go through, from assignment to completion or cancellation, including intermediate steps such as pending. This structured workflow ensures proper execution, traceability, and alignment with operational best practices.



### *7.2.3. Task status*

The diagram below outlines the lifecycle of a **task** within an ITSM process, as managed in **BMC Helix.** It illustrates the various possible states a task can move through — from staging and assignment to completion or closure — including pauses and work in progress. This workflow enables clear tracking and consistent execution of operational actions linked to a work order or broader process.




## 7.3. Deliverables

The Service Request Management practice produces several key deliverables that support effective service restoration, process control, reporting, and continuous improvement:

- **Customer satisfaction survey:** templates to collect end-user feedback on the fulfillment experience and support service improvement.

- **Knowledge Articles:** User-facing content enabling self-help or guiding users on how to submit requests properly (e.g., guides, FAQs).

- **Communication Templates and Notifications:** Standardized communication formats for notifying end-users regarding their requests (request received, in progress, fulfilled, rejected, etc.).

- **KPIs and continuous Improvement Plan:** A set of identified KPIs and improvement actions, based on metrics analysis and user feedback, to enhance process efficiency. These deliverables will be the basis of the committee dedicated to the review of the continuous improvement committee (described in the governance chapter).

- **Training and Onboarding Materials:** Guides and materials for onboarding new support staff or training existing teams on request handling procedures and tools. Example: the user guide (available from Mid June)