

# ITIL® SERVICE REQUEST FULFILLMENT



[ITIL 4 Service Request Management >](#)

## ITIL service request Fulfillment

When a user submits a formal request for something — a password change, new hardware or software they would like, or pretty much anything they want or need, it's called a **service request**.

ITIL's formal definition of service request is "*a request from a user for information, advice, a standard change, or access to a service.*"

So what's a standard change? A **standard change** is simply a pre-approved change that is low risk and that follows a standard procedure.

## Download Now: ITIL 4 Best Practice e-Books

These all-new for 2020 ITIL e-books highlight important elements of ITIL 4 best practices. Quickly understand key changes and actionable concepts, written by ITIL 4 contributors.

[Free Download >](#)



### [Free Download >](#)

It's important right out of the gate to point out the difference between service requests and incidents, though, since they're easily confused but follow much different ITIL processes.

Incidents are unplanned interruptions to your IT services, or reductions in the quality of your IT services. So when a user reports an incident, they are notifying you of the unavailability or decreased performance of an IT service they normally have access to.

Service requests, on the other hand, deal with requests for something new to be provided to the user that they don't already have, whether that's a new version of a software program, or access to an online portal.

In general, service requests are often lower risk, requiring less approval protocol. In fact, many service requests may be things that are already preapproved — like being granted access to a printer in a common area, or upgrading a laptop to the latest version of a company's preapproved productivity software suite.

Since these types of requests are frequent (but also less risky than incidents), ITIL draws a distinction between them, and suggests handling them with their own set of processes.

To that end, request fulfillment is exactly that: the process to handle service requests.

## **The objectives of request fulfillment**

Simply put, request fulfillment is all about making sure customers have easy access to the IT services they need to get their jobs done. Top objectives are:

- To help users clearly understand what services are available, how to request them, and how long it will take for them to be fulfilled
- To create a distinct process for handling service requests that is different from your change management and incident processes
- To properly deliver all of the components of the standard services requested (like software AND licenses, for example, in the instance of a software request)
- To assist with general information, comments, and complaints

## **Scope**

The types of service you offer, and the requests you receive, will vary dramatically from company to company. Unlike incidents, which are unplanned, service requests can be planned for. That means the process for how you handle each type of service request can be broken down into a thoughtful, methodical set of steps or actions and documented in a process flow.

These request models should be built for each type of request you will receive, and address each step or phase of request fulfillment. Be sure to consider:

- **Who will handle the request?** What individuals or teams are needed?
- **How is the service delivered?** What's the process?
- **How quickly will you fulfill the service request?** Is there an SLA or time window?
- **If the service can't be completely fulfilled, what happens?** Plan for how you will escalate.

Ultimately, it's up to each organization that provides standard services to properly document their offerings.

## Why do it, anyway?

Simple. Request fulfillment is all about efficiency and cost reduction. By making it super easy for users to get what they need, you can cut down on the confusion and bureaucracy often associated with asking for what you need. Simple, repeatable processes cost the business far less to fulfill, and in general, result in far happier users.

## The service request process

The process begins when a user places a service request. This is often handled through the service desk, using self-help tools where they can easily choose the service they require from a standard menu of selections you have pre-defined.

Not every service request has to come through a web-based self-help interface, however. It's not uncommon for a requestor to call the service desk directly, for instance.

Once a service is requested, it may be automatically approved or alternately routed for approval, since occasionally services are offered that still require management or financial approval, or even approval from a compliance perspective.

In those cases, your request model should include all appropriate approval steps, along with plans for how the request will be handled once approved and declined.

After a requested service is approved, or when no approval is required, the request must be assigned to the appropriate individual or team for review, and ultimately, fulfillment. Simple requests are often handled by the service desk team, whereas requests that require third party products or services may be forwarded on to vendors or other internal / external resources.

At each step of the way, it's important that the service desk keep track of the status of the request. Typical request statuses are:

1. Open
2. Assigned
3. In Progress
4. Pending (approval, additional information, availability, etc.)
5. Complete
6. Closed

Once the service has been delivered (and completion is verified), it should be referred back to the service desk for closure, prior to which the service desk should confirm that the requestor is satisfied

with the services delivered.

## **The 5 sub-processes of ITIL request fulfillment**

ITIL 2011 completely revised the request fulfillment process, breaking it down into 5 sub-processes:

### **Request Fulfillment Support**

*Objective:* To provide and maintain the tools, processes, skills and rules for an effective and efficient handling of Service requests.

### **Request Logging and Categorization**

*Objective:* To record and categorize the Service Request with appropriate diligence and check the requester's authorization to submit the request, in order to facilitate a swift and effective processing.

### **Request Model Execution**

*Objective:* To process a Service Request within the agreed time schedule.

### **Request Monitoring and Escalation**

*Objective:* To continuously monitor the processing status of outstanding Service Requests, so that counter-measures may be introduced as soon as possible if service levels are likely to be breached.

### **Request Closure and Evaluation**

*Objective:* To submit the Request Record to a final quality control before closed. The aim is to make sure that the Service Request is actually processed and that all information required to describe the request's lifecycle is supplied in sufficient detail. In addition to this, findings from the processing of the request are to be recorded for future use.

## **Relationships to other ITIL processes**

It's worth noting the linkages between Request Fulfillment and other ITIL processes.

While request fulfillment is pretty simple, it still connects to your Financial Management process (to understand the cost of services, and ensure that the resources and workload involved in fulfillment them is accounted for) and your Change Management process (whenever a request relates to a Standard Change).

## **Measuring your effectiveness**

ITIL also defines a few key metrics you can use to judge both the efficiency and effectiveness of your request fulfillment process, including:

1. How satisfied your users are with how their service requests are handled
2. The amount of outstanding service requests currently in your backlog
3. How long it takes you to handle each type of service request
4. Cost (per type of service request, on average)

5. The percentage of service requests that are handled within agreed SLAs
6. A breakdown of service requests by stage (i.e., in progress, closed, etc.)

## Key recommendations

- **Make it easy to request your services.** Start with the most popular, easy-to-fulfill services, and create a web-based request catalog (using a shopping basket approach) to make it easy for users to place their requests. Ask for the information you need to fulfill the request, but be careful not to bog users down with too many form fields or unnecessary information.
- **Be sure to clearly communicate your SLA's.** It's important that both your service desk, any other parties involved in fulfillment, and your users / customers are all in agreement around how long fulfillment will take.
- **Document the details.** For each service you offer, make sure you think through and document who is allowed to request them, what approvals are required, who pays for them, etc.
- **Stay on top of user satisfaction.** Clear communication at every stage about what to expect and when can keep your customer satisfaction ratings high and reinforce the value of your service desk operation.