

Tasks and Screens

- [Technician Leave Approvals](#)
- [Service Mobile Alerts](#)
- [Technician Workspace](#)

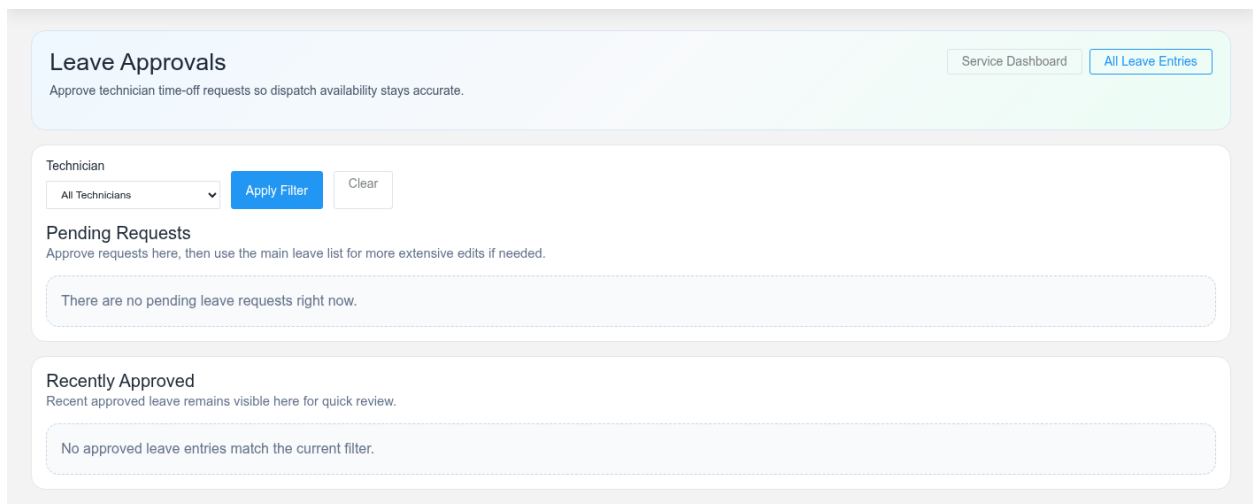
Technician Leave Approvals

Technician Leave Approvals

Purpose and when to use this screen

Technician Leave Approvals lets authorized supervisors review pending service leave before it blocks dispatch capacity. The screen requires permission to change service leave and shows up to 50 pending requests plus 20 recently approved requests, optionally filtered to one active technician.

Process leave requests



The screenshot shows the 'Leave Approvals' interface. At the top, there's a header with the title 'Leave Approvals' and a subtitle 'Approve technician time-off requests so dispatch availability stays accurate.' To the right of the header are two buttons: 'Service Dashboard' and 'All Leave Entries'. Below the header is a filter section for 'Technician' with a dropdown menu set to 'All Technicians', an 'Apply Filter' button, and a 'Clear' button. Underneath the filter is a section titled 'Pending Requests' with the subtitle 'Approve requests here, then use the main leave list for more extensive edits if needed.' This section contains a message: 'There are no pending leave requests right now.' Below that is a section titled 'Recently Approved' with the subtitle 'Recent approved leave remains visible here for quick review.' This section contains a message: 'No approved leave entries match the current filter.'

The Technician Leave Approvals approvals screen in the Brisk documentation demo.

1. Filter by **Technician** when reviewing one person, or leave the filter open to work the shared pending queue.
2. Confirm technician, leave type, start, end, notes, and whether the request overlaps another absence.
3. Review the dispatch board, today/upcoming bookings, and working-hours coverage for the requested interval.
4. If details are wrong or incomplete, use **Edit** and resolve the request with the technician. Do not approve it merely to remove it from the pending list.
5. Select **Approve** once the operational decision is made. Brisk marks the request approved, records its modification time, and notifies the technician.

Understand scheduling impact

Approved leave communicates that the technician should not be treated as available during that interval. Existing bookings do not automatically become somebody else's responsibility. Reassign or reschedule affected work deliberately and communicate customer-facing changes through the normal dispatch process.

Correcting an approval

The approval action is idempotent: selecting it again reports that the request is already approved. If approved dates or status need correction, use the supervisor-authorized leave record workflow rather than asking the technician to cancel it; technicians can cancel their own pending requests but cannot rewrite approved leave.

Approval controls

Apply the organization's leave policy consistently, keep sensitive medical details out of general notes, and separate policy approval from dispatch coverage work. Never share supervisor credentials to make the Approve button appear.

Service Mobile Alerts

Service Mobile Alerts

Purpose and when to use this screen

Mobile Alerts registers the current browser for service dispatch and technician push notifications. Registration is per browser/device, not per phone number, and depends on browser push support, notification permission, a service worker, and server-side VAPID configuration.

Review mobile alerts

The screenshot shows the 'Mobile Alerts' interface. At the top, there's a navigation bar with a menu icon, the date 'Mon, Aug 10, 2026', a search bar, and a 'My site' link. The main heading is 'Mobile Alerts' with a subtitle 'Register mobile browsers for service dispatch updates and technician notifications.' and a 'Back to Dispatch' link.

The interface is divided into two main sections:

- Current Device:** This section displays the status of the current browser. It shows 'Browser support: Supported', 'Permission: denied', and 'Server status: VAPID keys or push dependency still need configuration.' Below this, there's a 'Device Label' field with the example 'Jason iPhone'. There are two buttons: 'Enable Push on This Device' (blue) and 'Disable This Device' (red). At the bottom of this section are 'Send Test Notification' and 'Run Push Debug' buttons.
- Registered Devices:** This section shows a table of registered devices. The table has columns for 'Label', 'Last Seen', and 'Status'. The message 'No mobile devices have been registered yet.' is displayed below the table header.

Below the 'Current Device' section, there's a 'Temporary Push Debug' section with a description: 'Runs the browser capability checks directly on this device and prints the results below.' It contains a JSON object representing the browser's capabilities:

```
{
  "timestamp": "2026-08-10T18:13:50.304Z",
  "href": "http://127.0.0.1:8010/service/mobile-alerts/",
  "origin": "http://127.0.0.1:8010",
  "secureContext": true,
  "notificationAvailable": true,
  "notificationPermission": "denied",
  "serviceWorkerAvailable": true,
  "pushManagerAvailable": true,
  "serverReadyFlag": false,
  "publicKeyPresent": false,
  "userAgent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) HeadlessChrome/149.0.7827.55 Safari/537.36",
  "serviceWorkerRegistrations": [],
  "currentRegistrationScope": null
}
```

The Service Mobile Alerts alerts screen in the Brisk documentation demo.

1. Open the page on the device and browser that should receive alerts.
2. Review **Browser support**, **Permission**, and **Server status**. If the server is not ready, an administrator must configure the push dependency and VAPID keys.

3. Enter a meaningful **Device Label**, such as `Jason iPhone`, so the subscription can be recognized later.
4. Select **Enable Push on This Device** and respond to the browser permission prompt.
5. Select **Send Test Notification** and confirm the notification appears while the browser is both open and in the background.

Manage registered devices

The Registered Devices table shows label, last-seen time, and active/disabled state. Remove an old subscription after a phone is replaced, a browser profile is reset, or a device should no longer receive work information. Use **Disable This Device** when working on the currently registered browser.

Act on an alert

Open the linked [Service Order](#) or booking and verify the customer, equipment, priority, schedule, technician, and current status. A push alert may be stale after reassignment or completion; the service record is the source of truth.

Troubleshoot delivery

Use **Run Push Debug** to capture browser capability results. Check operating-system notification settings, browser permission, network access, active subscription status, and server readiness. Record the device label and debug output when escalating, but do not include customer or job details in an external support ticket unless authorized.

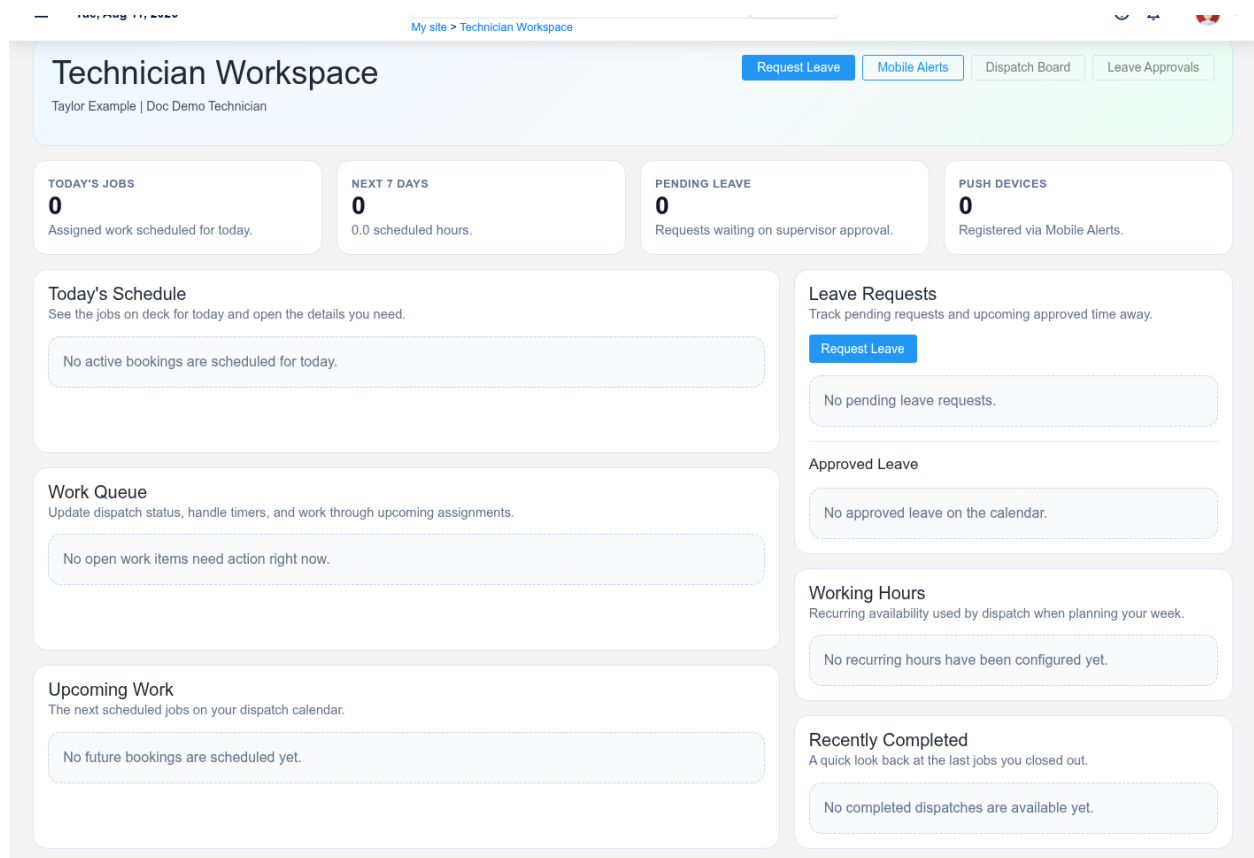
Technician Workspace

Technician Workspace

Purpose and when to use this screen

Technician Workspace is the signed-in technician's daily service landing page. It combines today's schedule, the actionable work queue, upcoming bookings, timer state, leave requests, and notification-device status. It is a working view of assigned records; the service order and booking remain the source of truth.

Use the technician workspace



The technician's Brisk workspace for assigned service work and next actions.

Start with **Today's Jobs** and the **Work Queue**. For each card, confirm customer, job reference, priority, status, scheduled time, and location. Read dispatch notes before traveling. Use **Directions** for navigation, then **Open Job** to review equipment, approved scope, parts, labor, notes, and

customer context on the [Service Order](#).

Update status and time as work happens

Use only the status actions offered on the booking, such as marking the next dispatch state. Do not advance status to remove a job from the queue when the underlying event has not occurred. Start the timer for the service order when billable or tracked work begins and stop it when that work pauses or ends. Only one active timer should represent the technician's current job; resolve an unexpected running timer before starting another.

Review upcoming work and capacity

The next-seven-days metric and Upcoming Work table help the technician anticipate appointments, locations, and schedule conflicts. They are not a substitute for dispatch approval. If a job is missing, duplicated, assigned to the wrong technician, or scheduled at an impossible time, contact dispatch rather than editing unrelated records.

Manage leave and alerts

Use **Request Leave** to submit planned absence and track it as Pending until a supervisor acts. An approved request affects scheduling capacity; technicians cannot simply rewrite approved leave. Use **Mobile Alerts** to register or test the current device. Open the linked job from an alert and verify it is still current before acting.

If the workspace is empty or unavailable

Confirm the user is linked to an active Technician and Employee, the bookings are assigned to that technician, and the schedule/status is within the workspace scope. Dispatch-board and approval links appear only when the user has the corresponding permissions.

Navigation

Common tasks

- [Follow service work through billing](#)