

Shelfbot IT Integration

SB-SPEC-IT · Version 3.3 · 10 September 2026

1. Introduction

1.1 Purpose

This document specifies what the Customer's IT team provides for a Shelfbot deployment and how the Shelfbot system connects to the Customer's site and to the Customer's WMS: the site network, the connectivity for cloud replication and remote support, and the requirements the WMS API integration must meet. It is written for the Customer's IT team.

1.2 Scope

This document **covers**:

- The network the system needs at the site, and what the Customer provides
- The data flows, their direction and the firewall rules they need
- How the system behaves when connectivity is lost
- The requirements on the WMS API integration: connectivity, environments, integration profiles, testing, go-live and ongoing change control
- Who provides what

It **does not cover**:

- The API itself, which is specified in the Shelfbot API document
- When each item is due, the build steps and the readiness evidence, which are set out in the Shelfbot Project Plan and recorded in the IT Integration workbook
- Acceptance of the live integration, which is tested in the Shelfbot Site Acceptance Test
- Per-site values such as the WMS software and version, addressing and hostnames, which are recorded in the IT Integration workbook at onboarding, and contacts, which are in the project's Contact Register

1.3 Related documents

Document	Used for
Shelfbot API	The API itself: transport, authentication, webhooks, idempotency, message shapes
Shelfbot Project Plan	Programme: the milestones and readiness requirements that time the network and integration work, and the build and test steps
Shelfbot Site Acceptance Test	Acceptance of the live integration: the network and connectivity confirmation, the product sync, putaway and pick cycles
Shelfbot Grid Design	Physical positions: server cabinet location, network point and cable route
Shelfbot Information Security Specification	Remote support access, audit logging, encryption and patch management

Document	Used for
Shelfbot Service Level Specification	The API availability commitment and the reported integration KPIs

1.4 Definitions

Term	Meaning
Server cabinet	The Shelfbot-supplied cabinet holding the Shelfbot Server, an Ethernet switch, a UPS and a 4G modem/WiFi router, installed at the position fixed in the Grid Design.
Uplink	The single wired connection from the server cabinet to the Customer's network point.
Dedicated VLAN	A Customer VLAN that carries the uplink only: no other devices on it and no routing between it and the rest of the Customer's network.
Shelfbot cloud	Shelfbot's hosted service, which receives the replicated site database and delivers the customer portal, Shelfbot's monitoring and the WMS event stream.
Sandbox	A Shelfbot API environment with its own hostname and credentials, used to build and test the integration.
Webhook	An HTTPS delivery from the Shelfbot cloud to the Customer's WMS endpoint carrying order, inventory and exception events.
Integration profile	A recommended starting set of API endpoints and event types for a WMS to implement: Shelfbot Simple or Shelfbot Full.
Operator mobile computers	The customer supplied handheld mobile computers with integrated barcode scanners (Zebra TC53 or equivalent) that the operators use at the stations and the put-to-light trolleys.
IT Integration workbook	The per-site record, kept with the project, of the IT team's values and completion evidence; contacts are in the project's Contact Register.

2. Network

2.1 How the system connects

The server cabinet contains the Shelfbot Server, an Ethernet switch, a UPS, and a 4G modem/WiFi router. It is installed at the position fixed in the Grid Design, where the Customer provides a network point.

The cabinet is self-contained:

- The cabinet router runs its own private network for all Shelfbot equipment, behind the single uplink the Customer provides. Nothing inside the cabinet network is addressable from the Customer's network, and the system never initiates connections into the Customer's network.
- The robots are wired: they connect over cabling installed with the racking, never over wireless.
- The cabinet router's WiFi serves only the operator mobile computers used at the stations and the put-to-light trolleys, and the fixed barcode scanners, all within the grid and trolley area. The robots do not use WiFi. The Customer does not provide wireless coverage for the system.

2.2 What the Customer provides

- A **dedicated VLAN** terminating at one wired network point at the server cabinet position. The VLAN carries the cabinet uplink only: no other devices on it, and no routing between it and the rest of the Customer's network. The uplink address may be DHCP or static, agreed with Shelfbot and recorded in the IT Integration workbook.
- **Outbound internet connectivity** from that point, per *Data flows and firewall rules* below. **No inbound connections are required**: every connection is initiated outward by Shelfbot equipment.
- **Operator mobile computers**: handheld mobile computers with integrated barcode scanners, Zebra TC53 or equivalent, to the minimum specification in the Shelfbot Technical Data Sheet, in the quantities agreed in the Project Plan. The Customer supplies, manages and patches them under its own mobile device management policy. They connect to the cabinet router's WiFi (*How the system connects*).

2.3 Data flows and firewall rules

Every flow is initiated outbound. The destination hostnames and ports for the deployment are recorded in the IT Integration workbook; they are stable for the life of the deployment and Shelfbot gives notice before any change.

Flow	From → To	Protocol	Purpose
Replication and support	Shelfbot Server → Cloudflare's network (Cloudflare Zero Trust)	Outbound HTTPS/QUIC; ports per the IT Integration workbook	One outbound connection carrying both live replication of the Shelfbot Server's database to the Shelfbot cloud (shelfbot.io), which feeds the customer portal and Shelfbot's monitoring, and remote support access. Outbound-initiated, identity-verified, and it reaches only the equipment inside the Shelfbot cabinet
Time sync	Shelfbot Server → public time servers	NTP (UDP 123)	Trustworthy timestamps on the audit trail
WMS → Shelfbot API	Customer WMS → the deployment's Shelfbot API hostname	HTTPS (TCP 443)	SKU and order sync (<i>WMS API integration</i>)
Shelfbot → WMS events	Shelfbot cloud → the Customer's WMS endpoint	HTTPS (TCP 443)	Order, inventory and exception events (<i>WMS API integration</i>)
Portal	Users' browsers → shelfbot.com	HTTPS (TCP 443)	Live dashboard: ordinary web traffic, no site firewall involvement

2.4 If the connection fails

The system is designed to keep operating locally:

- **Site uplink lost**: the cabinet fails over to its 4G path automatically. Replication, support access and event delivery continue; warehouse operation is unaffected.
- **All connectivity lost** (site network and 4G together): the system keeps operating locally. Picking and putaway continue on orders already received. Order and inventory events queue on the Shelfbot Server and are delivered, in order, when connectivity returns, under the retry rules in the

Shelfbot API document. New orders from the WMS and portal freshness wait for the connection; remote support is unavailable until a path returns.

2.5 The 4G path

The cabinet's 4G path needs adequate signal at the server cabinet position. Where the signal is inadequate, an external antenna is fitted at a position agreed with the Customer. The 4G modem, SIM and data plan are part of the Shelfbot subscription (*Responsibility Matrix*).

3. WMS API integration

The Customer's WMS integrates with the Shelfbot ASRS through the Shelfbot API. The Customer builds and maintains the WMS side of the integration against the Shelfbot API document; Shelfbot provides the API, its documentation, a sandbox environment and credentials, and supports the integration during the build and on an ongoing basis under the Maintenance and Support Services.

3.1 Connectivity

Both legs of the integration travel outbound over HTTPS with signed requests, per the Shelfbot API document's *Authentication* section:

- **WMS to Shelfbot:** the WMS calls the deployment's Shelfbot API hostname from wherever the WMS runs. This is outbound HTTPS only; the Customer exposes nothing for this leg.
- **Shelfbot to WMS:** the Customer provides an HTTPS endpoint where the WMS receives order, inventory and exception events. Shelfbot webhooks are authenticated by their HMAC signature, per the Shelfbot API document's *Authentication* section. Delivery originates from Cloudflare's published IP ranges, which are shared infrastructure, so source-IP filtering is not a supported authentication method: verify the signature on every delivery. The endpoint must be reachable from the internet, and the WMS should acknowledge each delivery quickly with any HTTP 2xx and process the events asynchronously, so that processing never holds up the acknowledgement. A cloud-hosted WMS usually satisfies this as it stands; an on-premise WMS needs its receiving endpoint published through the Customer's usual public-facing arrangements.
- TLS at both ends per the API document's *Transport* section, with valid publicly-trusted certificates.

3.2 Environments

There are two environments with the same API surface. The **sandbox** has its own hostname and credentials and is used to build and test the integration. **Production** has its own hostname and credentials, issued for go-live.

3.3 Integration profiles

Shelfbot sends every WMS the same event stream; there are no tiers of service and no subscription options. What differs between integrations is how much of that stream the WMS chooses to process. Two profiles are documented as recommended starting points:

- **Shelfbot Simple:** implement the SKU update, SKU delete and order update endpoints, and close every order with fulfilled or cancelled once the WMS has finished with it; acknowledge every webhook delivery with any HTTP 2xx; process order.completed and shipper.closed (the manifest of

each shipper an order is picked into) and ignore all other event types; and use the read endpoints, such as fetching an order's current state or resolving a shipper label, whenever current truth is needed.

- **Shelfbot Full:** everything in Simple, plus processing line.picked progress events, shipper.opened, inventory events, exceptions, and serial and batch detail.

These are starting points, not tiers. A WMS moves between them one event type at a time, taking on an additional event type whenever it becomes worth processing, and both are fully compatible integrations. Per the Shelfbot API document's conformance rule, a compatible integration means the inbound endpoints implemented correctly plus the event stream acknowledged; which event types a WMS processes is its own choice.

3.4 Integration testing

The integration is tested end to end against the sandbox with dummy data before it is used on the live system. The test covers normal operation (product sync, the putaway cycle and the pick cycle) and the exception scenarios: a cancelled order, a short pick and a rejected item. What each scenario must demonstrate follows the chosen integration profile: a Simple integration shows the correct final state arriving through order.completed and the read endpoints; a Full integration also shows the individual exception events being processed. The product sync, putaway and pick cycles are exercised again on the live integration in the Shelfbot Site Acceptance Test (its *Operational Acceptance* section). The steps, owners and timing of the build and test are set out in the Shelfbot Project Plan (its *Integration* section).

3.5 Go-live

The integration's first production act is the catalogue load: the WMS pushes the full SKU catalogue through the API's SKU update endpoint in batches, the same path that ongoing sync uses from then on. The initial load is not a special operation: SKU updates are idempotent upserts, so re-pushing the whole catalogue is always safe, at go-live or at any time afterwards. The full load doubles as the production shakedown: once the catalogue has loaded cleanly, the integration is live and only changes flow thereafter.

3.6 Ongoing requirements

- **Change notice runs both ways.** The Customer tells Shelfbot before WMS upgrades, endpoint moves or certificate changes that affect the integration; Shelfbot gives notice before any change to API hostnames or credentials.
- **Credentials** are stored in the Customer's secrets management, never in source code or repositories. Shelfbot rotates credentials on request.
- **Integration support.** Shelfbot supports the integration on an ongoing basis under the Maintenance and Support Services.
- **Network changes.** The Customer's IT contact, recorded in the project's Contact Register, is the notification point in both directions for anything network-side: planned outages, ISP changes, renumbering, firewall changes. A planned change to the uplink is handled like any planned outage: tell Shelfbot in advance, and the system rides through on the 4G path.
- **Cabinet equipment.** Shelfbot monitors the deployed system remotely and maintains the cabinet equipment, including patching, per the Shelfbot Information Security Specification (its *Patch & Lifecycle Management* section).

4. Responsibility Matrix

Item	Shelfbot	Customer
Server cabinet and its equipment, including the 4G modem, SIM and data plan	Provides and maintains as part of the subscription	Provides the position and the network point
External 4G antenna where the signal at the cabinet position is inadequate	Supplies and fits	Agrees the position
Operator mobile computers: handheld devices with integrated barcode scanners to the minimum specification in the Shelfbot Technical Data Sheet	Confirms Shelfbot App compatibility with the device model before deployment	Supplies, manages and patches
Dedicated VLAN and the wired network point at the server cabinet position	Specifies	Provides and maintains
Outbound internet connectivity and the firewall rules for the flows in <i>Data flows and firewall rules</i>	Specifies	Provides and maintains
Replication, monitoring and remote support connection	Operates	Permits, outbound only
Shelfbot API, API documentation, sandbox and production credentials	Provides	
WMS side of the integration: inbound endpoints, the event endpoint, TLS certificates and secrets handling	Supports the build and the live integration under the Maintenance and Support Services	Builds, tests and maintains
End-to-end integration test against the sandbox	Performs jointly	Performs jointly
Catalogue load at go-live and the ongoing SKU and order sync		Performs
Change notice for API hostnames and credentials	Gives	
Change notice for WMS upgrades, endpoint moves, certificate changes and network changes		Gives
Cabinet equipment patching and lifecycle	Performs	
Per-site values: WMS software and version, uplink addressing, hostnames and ports, antenna position	Records in the IT Integration workbook at onboarding	Supplies

5. Revision History

Version	Date	Change
3.1	19/08/2026	Published to Web Site
3.2	06/09/2026	Restructured to the standard specification format: scope, definitions and responsibility matrix added; programme timing, build steps and readiness evidence moved to the Project Plan and the IT Integration workbook; integration profiles added; webhook authentication, acknowledgement and catalogue load clarified

Version	Date	Change
3.3	10/09/2026	Integration profiles updated for shipper events: Simple processes order.completed and shipper.closed and closes every order; Full adds shipper.opened; shipper label lookup named among the reads