

Verso

Software Lifecycle, Maintenance and Support

Document ID	IM.VERSO.INT.04
Document version	1.0
Date of issue	2026-10-06
Applies to	Verso 2026-09-25 (4bac47a)
Language	English
Publisher	iMobile

Contents

- 1. General 3
- 2. Development and release 3
 - 2.1. Development tools 3
 - 2.2. Release build 3
 - 2.3. Pre-release verification 3
 - 2.4. Version numbering 4
 - 2.5. Delivery of updates 4
- 3. Defect correction 4
 - 3.1. Reporting a defect 4
 - 3.2. Handling 4
 - 3.3. Timescales 4
- 4. Product improvement 5
- 5. Technical support 5
- 6. Personnel 5
 - 6.1. Customer personnel 5
 - 6.2. Vendor personnel 5

About this document

This document describes how Verso is developed, released, corrected and supported, and which skills are needed to operate it. It is part of the documentation delivered with the product.

Response and resolution times are not set by this document; they are agreed in the contract.

1. General

Item	Value
Product	Verso
Version covered by this document	2026-09-25 (4bac47a)
Release date	2026-09-25
Publisher	iMobile
Website	https://new-imobile.com

Verso builds a presentation from a text prompt or uploaded documents: it drafts an outline, picks slide layouts from a template, writes the text, adds charts, images and icons, lets the user edit the result in the browser and exports it to PPTX or PDF. It is offered as a vendor cloud service paid with credits and as a Docker package installed on the customer's servers under an annual licence.

The description follows the process groups of ISO/IEC/IEEE 12207 (Software life cycle processes): development, release, maintenance and operational support.

2. Development and release

2.1. Development tools

Server side: Python 3.11 (FastAPI, SQLAlchemy, SQLModel, Alembic); web interface: TypeScript, Next.js 16, React 19; runtime for the web interface and document parsing: Node.js 20; front proxy: nginx; export runs in Chromium; licence verification module: Python compiled to a native extension with Cython. The product is shipped as a Docker image.

2.2. Release build

The image is built with a multi-stage Dockerfile on `python:3.11-slim` (Debian 13) and `node:20`; dependencies are pinned by `uv.lock` and `package-lock.json`, and the Chromium version is pinned. The `presenton-export v0.4.2` export component is downloaded pre-built during the build. The licence verification module is compiled with Cython and added on top of the base image by `Dockerfile.onprem`. The build scripts do not sign images or archives.

2.3. Pre-release verification

The source tree contains automated tests: the server side uses pytest (87 test files: unit, integration, regression), the web interface has Node.js tests and a Cypress component test, and the licence verification module has 14 pytest tests, including verification of a signed sample licence and rejection when fields are altered. No release acceptance test programme is part of the delivery.

2.4. Version numbering

Verso has no release numbering of its own. A release is identified by the date and the short identifier of the source revision it was built from; this set describes the release of 2026-09-25, revision 4bac47a. The `package.json` file of the source tree carries the version of the Presenton code base the product is built on, 0.9.3-beta. The web interface does not display a version number. The licence verification module has its own version, 1.0.0.

2.5. Delivery of updates

On-premises: the vendor supplies a new image (or source) and package; the administrator rebuilds and restarts the container, and the database schema is upgraded at start-up by Alembic when `MIGRATE_DATABASE_ON_STARTUP=true` is set. The program has no automatic update and no update check. The cloud service is updated by the vendor.

3. Defect correction

3.1. Reporting a defect

Defect reports are accepted through:

- contact form: <https://new-imobile.com/contact>
- e-mail: admin@new-imobile.com

A report should state:

- the product name and version;
- the operating system version and a description of the environment;
- the steps that led to the defect;
- the text of the error message;
- the product's log files, if any (their location is given in the Installation and Administration Guide).

3.2. Handling

1. The report is registered and its receipt is confirmed to the reporter.
2. The defect is reproduced on the version named by the reporter and on the current version.
3. The report is classified as a product defect, a configuration or environment problem, or a change request.
4. A product defect is corrected in the source code; the correction is verified and included in the next release. If the defect prevents the product from being used for its purpose, the correction is released separately, without waiting for the next scheduled release.
5. The reporter is told the outcome and the version that contains the correction.

3.3. Timescales

Response and resolution times depend on the contract (licence agreement or support agreement) and are not given in this document.

4. Product improvement

Changes originate from user reports, pre-release verification, changes in the operating systems and third-party components the product depends on and, for functions tied to legal requirements, changes in those requirements.

Changes are delivered in subsequent releases. The version number changes according to the rules given under "Version numbering". The content of each release is communicated to users when it is published.

5. Technical support

Questions about installation, configuration and use are accepted through the same channels as defect reports. Support languages are English and Russian.

Technical support covers:

- advice on installing, configuring and using the product;
- receiving and handling defect reports;
- providing updates to the extent covered by the licence.

Support hours and any services beyond the above are defined in the contract.

6. Personnel

6.1. Customer personnel

A user needs basic skills with a web browser and presentation software. An on-premises administrator needs Linux and Docker (Compose) administration skills, experience setting up a TLS reverse proxy, and familiarity with model provider API keys or a local model server.

6.2. Vendor personnel

Development, release and technical support are carried out by the vendor's own specialists, who are proficient in the tools listed under "Development tools" and know the product's internals.

The location of the development team, the development infrastructure and the support desk is stated in the contract.

Document history

Document version	Date	Product version	Change
1.0	2026-10-06	2026-09-25 (4bac47a)	First issue