ROSSUM

TECHNICAL WHITEPAPER

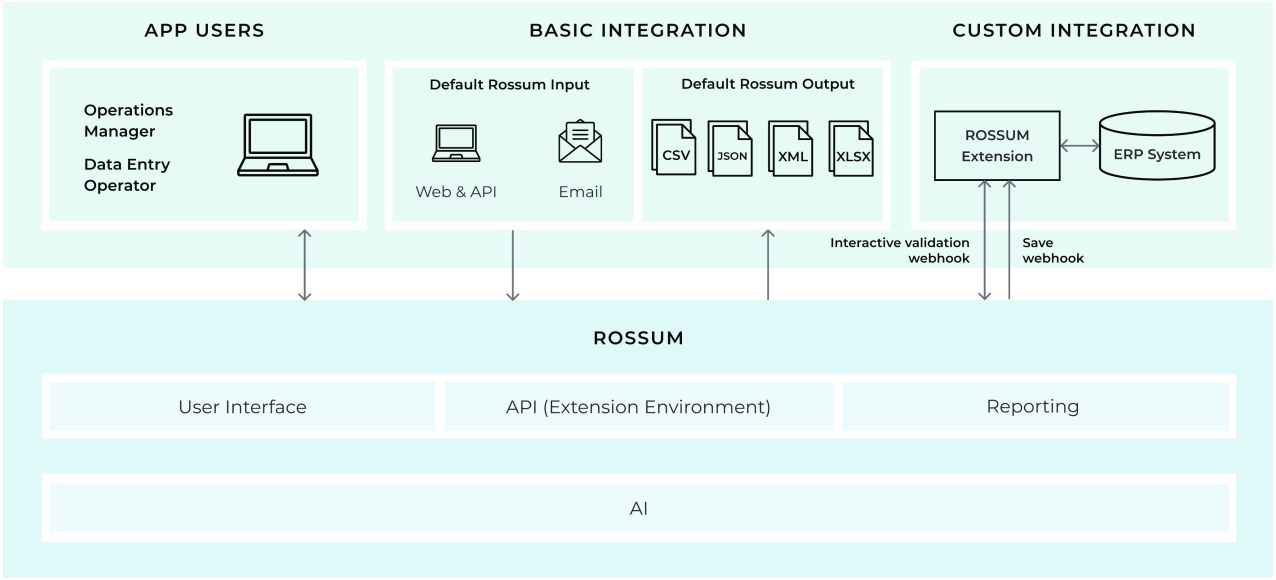
ROSSUM INTEGRATION OVERVIEW

LAST UPDATED: 2020-04-15

ROSSUM PLATFORM ARCHITECTURE

Rossum, as a platform for AI-based cognitive data capture from documents, consists of several modules spanning from the AI engine through the validation interface and workflow components to the extension API and default input/output interfaces.

At a high level, Rossum platform consists of four main components: **the AI Core Engine** that can automatically detect data fields, **the Validation User Interface** for data verification, correction and training, **the Extension Environment** that allows custom business rules to be plugged in and applications built around Rossum, and **the Reporting Engine** for process analysis and efficiency improvements.



ROSSUM DOCUMENT WORKFLOW

When Rossum captures data from documents, they undergo a series of states as they get processed by the system. Rossum is optimized for batch processing of documents, applying the AI-based capture asynchronously and then encouraging a periodic user batch review of all documents received so far. The typical workflow is as follows:

1. Documents are received by Rossum over time as they arrive. Each document is associated with a particular queue, while multiple queues may exist.

2. Documents stay in the “processing” state while data is automatically captured by the AI engine on background - then, they move to the “to review” state.
3. Custom business logic provided by a Rossum Extension may automatically export some of the documents queued for review without human intervention.
4. Users periodically log in to the Rossum web app, review the queued documents within the validation interface and “export” validated documents.
5. The validation interface allows the user to also capture extra data that is not automated by the AI engine, and its behavior may be customized by a Rossum Extension e.g. to provide on-the-fly consistency checks or interactive Vendor or PO Matching.
6. The user may also “postpone” or delete documents during validation.
7. Exported documents are then available for further processing on a queue basis - e.g. bulk download of the captured data.

While this is the most proven workflow for most applications, other workflow models are possible and covered later in the document. Most notably, it is possible to control when and which documents are reviewed by the user externally (see “embedded Rossum app”).

INTERFACING WITH ROSSUM

Rossum provides default input and output methods out of the box.

At **input**, Rossum can accept documents either through an **email gateway** (each queue comes with a unique inbox email address on Rossum servers which can receive documents) or by **browser upload**.

At **output**, Rossum allows users to **export** captured document data from a given queue. Documents can be selected using a flexible filter system (in particular based on their status and time range). Multiple formats are supported - JSON, XML, CSV and XLSX.