**Accounts Payable Automation**

**Version: 2**

**Business Requirements v.1**



Nexelus USA  
New York, NEW YORK (NY)  
646-558-1950

December 14, 2020

|  |  |  |  |
| --- | --- | --- | --- |
| **Revisions** | | | |
| **Date** | **Version Number** | **Revised By** | **Reason for Changes** |
| 2020-12-14 | v.1 | Imran Rahman | Initial Draft |

|  |  |
| --- | --- |
| **Open Items** | |
| **Location** | **Description** |
|  |  |

1. Introduction 5

Project Based/Production, Non-Billable and Overhead Vendor Invoices 5

Enhancements for media invoices and general architecture 5

2. System Requirements, Scope, Assumptions 6

3. System Data Flow Design 6

4. Database Design 7

5. Login Process 11

6. Workflow Step 1 - System Initiation 11

7. Workflow Step 2 - AP Invoice Mapping 11

8. Workflow Step 3 – Scanner 11

9. Workflow Step 4 – Invoice Requests 11

10. Workflow Step 5 – Create Manual Invoice 11

11. Workflow Step 6 – Invoice Approval and Posting 11

12. User Menu 11

13. Company Configuration 11

14. Routing Setup 11

15. Multi-Currency Support and Mapping 12

16. Manage Non-Mapped Invoices 12

17. Manage Invoice Models 12

18. New Invoice Model Setup 12

19. New Invoice Model Setup (Test) 12

20. Invoice Model Setup - Request\_<N> 12

21. Manage Invoice Documents 12

22. Review Invoice Document 12

23. Additional Documents 12

# Introduction

This document specifically focuses on Version 2 of AP Automation system (AKA AP workflow). While version 1, released in Dec 2020, focused on addressing basic requirements around AP automation of workflow for media business, version 2 is to handle “non-media” accounts payable invoices in addition to developing Nexelus proprietary AI and machine learning algorithm and patterns to predict the “best” solution in resolving discrepancies for all types of invoices. Non-media AP invoices include project based, referred to as “production”, invoices along with non-billable and overhead (rent, utilities, etc.) vendor invoices.

## **Project Based/Production, Non-Billable and Overhead Vendor Invoices**

In media advertising business, insertion orders (IO) are required to process any media vendor invoice. However, for this category of vendor invoices a purchase order (PO), equivalent to an IO, is not always required or the client may not be using purchase orders through their project based/ERP systems. For Nexelus system following three fields are required to process this category of invoices:

* Project/Job code (level2)
* Activity code (level3 – however this level will not be available in non-Nexelus/ERP environments)
* Expense Type (printing, editing, retouching, meals, etc. In Nexelus, there is a set of valid expense types that are available for a specific level2/level3 combination)

If the PO is stamped then it maps to level2/level3/expense type combination, like media invoices and IO numbers. Please note that companies like 360i and HY offer both Media and Non-Media (production) services to their respective clients. Hence, the design needs to cater to both types of processing, which will potentially carry different data elements and can have different workflow in-terms of discrepancy resolution and approval levels, etc. However, the basic flow of data processing will be the same as it is implemented in the media AP workflow system, i.e., creation of models, routing for discrepancy resolution and approvals, and creating AP vendor invoice records in the ERP system along with pushing the attachment.

## **Enhancements for media invoices and general architecture**

For version 1 of the software, Nexelus relied on AWS Textract to scan and read data from a “picture” or a PDF, Version 2 will include certain predictability models to predict and assign data elements, so the workflow moves to the next action level. Following are a couple of examples/scenarios:

* **Discrepant Invoices**: Version 1 requires “manual” routing by an “admin” to route the invoice in question to someone who the admin thinks should be able to resolve the issue by typically assigning an IO number for further workflow processing. This is typically done by an admin personnel by reading the Advertiser name and/or campaign description from the scanned invoice. This step must be handled by the system using AI and machine learning. We have several data elements in the “ERP” provided data set and we need to use “already” processed data, advertisers name, campaign and media plan description, month of service on the invoice, along with IO data that provides IO dates of placements, campaign name, etc. System should be able to predict and route this type of discrepant invoices to relevant team member (employee).
  + AI to use prediction and subsequent “action of success” and/or rerouting from saved dataset, in case machine did not route to correct individual originally, and learn the pattern for future handling of the similar scenario. This type of learning must be shared across all clients.
* **System to pick IO number** even if the IO/PO number is not on the vendor invoice. Building on the point above, system should be able to locate “most suited” IO number. Visual indication of level of “predictability” must be preserved and displayed

# System Requirements, Scope, Assumptions

During phase I (version 1) we did not handle Registration, Trial Accounts, Licensing, Subscription and Billing setup through the user interface. This will still be the case with version 2 as all these elements will be maintained by Nexelus personnel on database level.

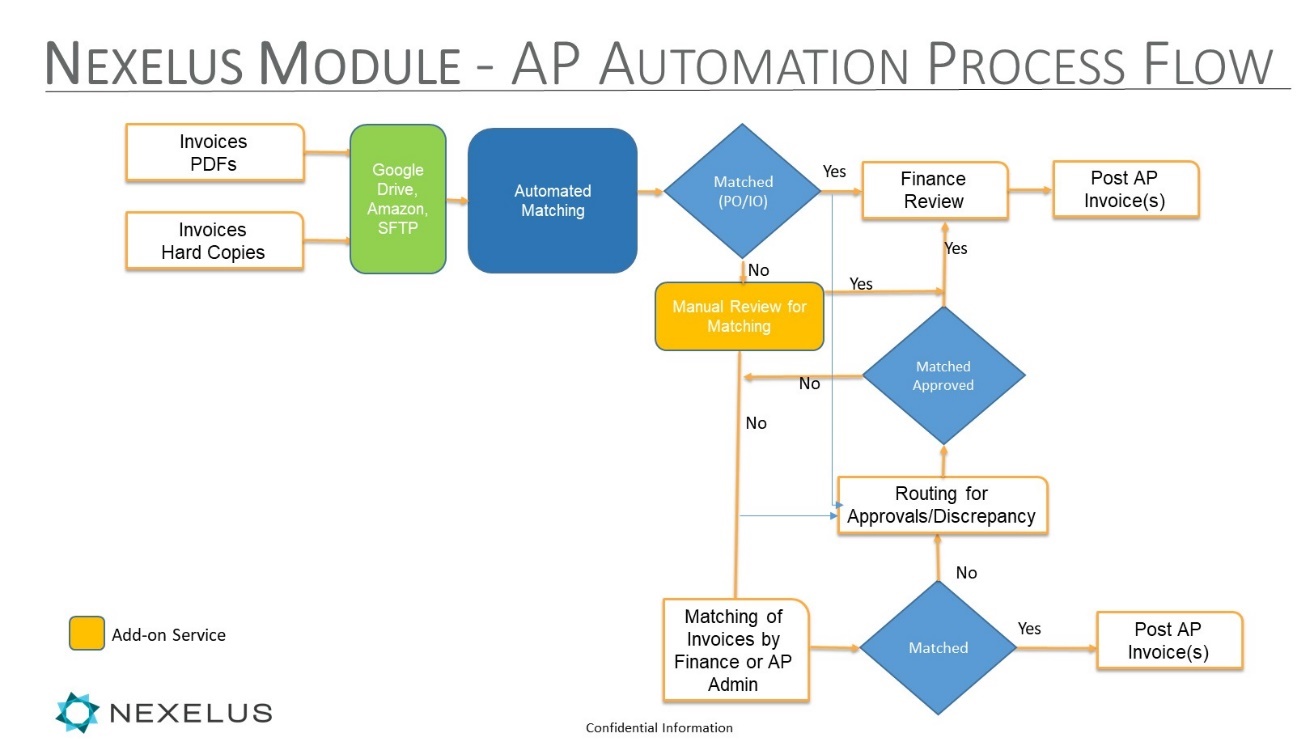
Invoice input documents (scanned images or generated) are supported only in PDF, JPEG or PNG formats. AP Automation system will not convert input files from other formats.

AP Automation System is designed as totally disconnected, self-sufficient application but it does require data feed from the other system – client’s ERP system, not in a real-time though. Please note that certain fields, especially for non-media, must be configurable as the “activity/level3” is unique to Nexelus. Most systems have project/job and expense type as the hierarchy. In some legacy systems overhead can be coded to “client code” and expense types without any tagging of a project code.

No change to AP Automation database design with reference to multi-tenant database approach using Amazon Aurora MySQL database engine initially deployed on a single database instance. Common schema to maintain sharable set of model definitions and corresponding reference data will be reused across client’s dedicated schemas. Each client schema will maintain client setup information, routing setup data, processed invoices, etc.

# System Data Flow Design

Below is high level diagram that shows overall data flow within an application (no change from version 1).



# Database Design

New tables, along with changes to existing tables will be discussed and documented.

Section below list data elements that will read into the AP workflow to handle non-media invoices







# Login Process

No change, same as version 1

# Workflow Step 1 - System Initiation

No change, same as version 1

# Workflow Step 2 - AP Invoice Mapping

* TBD

# Workflow Step 3 – Scanner

No change, same as version 1

# Workflow Step 4 – Invoice Requests

* TBD

# Workflow Step 5 – Create Manual Invoice

TBD

# Workflow Step 6 – Invoice Approval and Posting

* For non-media/production invoices. Separate approval process is to be established

# User Menu

TBD

# Company Configuration

TBD. There will be necessary changes

# Routing Setup

TBD. There will be necessary changes

# Multi-Currency Support and Mapping

We may need to handle multi-currency processing and we should also discuss multi-language support as well

# Manage Non-Mapped Invoices

# Manage Invoice Models

# New Invoice Model Setup

# New Invoice Model Setup (Test)

# Invoice Model Setup - Request\_<N>

# Manage Invoice Documents

# Review Invoice Document

# Additional Documents