# Self Service Adtech Integration.

Problem Statement:  
Nexleus is integrated with multiple adtech platforms i.e. Google Ads, Google DV360, MS ads, Yahoo, TTD etc. Currently, enabling and integrating all these integrations are manual process and require extensive back and forth communication with client and development team which causes confusion and dissatisfaction with the client.

Proposed Solution:  
Nexelus, will provide self-service features for all integrations which will allow users to configure all integrations themselves without requiring extensive involvement of Nexelus resource.  
To achieve this goal, Nexelus will develop extensive set of UI(s) along with explanatory documentation which will allow users to follow simple steps to enable integrations.  
furthermore, these UI(s) will work as a health checker services for integrations by providing real time information about health of integrations (integrations includes both, Pushing of Campaign / IO/ BO and pulling o delivery)

# Enable Integrations in IPC

The following is the required information in IPC to enable integration(s).

IPC:  
Following steps are required to enable an AdTech tool integration for company in Nexelus.

Create Consumer  
every client is assigned a consumer id in IPC. This consumer ID is unique for individual Nexelus client. Multiple companies under same Nexelus instance shares the same consumer ID.  
Following is the table for consumer. 1st step of IPC integration is to add an entry in Consumer table to assign consumer id to Nexelus client.  
 Please remember this is only 1 time setup for one company. If consumer ID is already assigned to Nexelus Client, we should reuse the consumer ID already assigned instead of creating new consumer ID.

#### Table Name: PDM\_CONSUMER

Following is the structure of this table only with required columns.

|  |  |  |
| --- | --- | --- |
| Column Name | Column Type | Description |
| Consumer\_id | Int | Unique id for consumer, always use max(consumer)\_id +1 to assign consumer id to new customer |
| Name | Nvarchar(256) | Name of consumer (company name) |
| Type | Tinyint | Type of consumer indicates if it is internal client or external, currently we have only internal consumer whose value is 3. use 3 for Nexelus clients. |
| Create\_date | Datetime |  |
| Create\_id | Nvarchar(100) |  |

### Account

Every IPC consumer is assigned an account id in IPC. Once account id is assigned to a consumer, this account ID will be used for further integration setup.

This account ID setup is one time setup for an individual Nexelus Client. Once Account ID is assigned to a client.

The following is the table for consumers.

Please remember this is only 1 time setup for one company. If the account ID is already assigned to Nexelus Client, we should reuse the account ID already assigned instead of creating new.

Table Name: PDM\_CONSUMER\_ACCOUNT  
Following is the structure of this table only with required columns.

|  |  |  |
| --- | --- | --- |
| Column Name | Type | Description |
| account\_id | Varchar(128) | Account ID for new client |
| Consumer\_id | Int | Foreign key |
| Eff\_date | Datetime | Current date |
| Exp\_date | Datetime | Future date when account will expire. |

Ad Tool Account  
Ad tool setup is required for each ad tool integration. This step requires adding an adtool account and assigning a access\_tag to the adtool account.  
Once the client is ready for ad tool integration, the system should create ad tool account in IPC for integration. This setup is required for individual ad-tool for each client.in this process, system will add entries in following 2 tables.

Table Name: PDM\_ACCOUNT\_USER  
Following is the structure of this table only with required columns.

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Description** |
| account\_id | Varchar(128) | Foreign key |
| user\_id | nVarchar | Unique id |
| user\_name | nVarchar | Current date |
| Password | nVarchar | Future date when account will expire. |
| is\_active | Tinyint |  |
| dev\_token | Varchar(256) | 333 for now for all entries. |

**AdTool IDs**

|  |  |
| --- | --- |
| **ad\_tool\_id** | **AdTool name** |
| 1 | AdWords |
| 2 | AdCenter |
| 6 | GCM |
| 13 | Sizmek |
| 14 | TradeDesk |
| 15 | Facebook Ad |
| 16 | MS Ads |
| 17 | Yahoo Gemini |
| 19 | Google Ads |
| 21 | Display & Video 360 |
| 24 | Twitter |
| 25 | LinkedIn |
|  |  |

Table Name: PDM\_ACCOUNT\_USER\_AD\_TOOL  
Following are required columns for this table

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Description** |
| account\_id | Varchar(128) | Foreign key |
| ad\_tool\_id | int | Id of adtool, this is predefined id for each adtool. Please ceck adtool id lists for this Entery |
| seq\_id | int | Incremental Seq id assigned to this record, for each tool , it starts with 1 |
| user\_id | Varchar | Foreign key from pdm\_account\_user table. |

**Table Name: pdm\_account\_ad\_tool**Followings are required columns.

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Description** |
| account\_id | Varchar(128) | Foreign key |
| ad\_tool\_id | Int | Id of adtool, this is predefined id for each adtool. Please see section “c” above for ids. |
| seq\_id | Int | Incremental Seq id assigned to this record, for each tool , it starts with 1 |
| access\_tag | narchar(256) | Foreign key from pdm\_account\_user table. |
| ipc\_name | Nvarchar(256) |  |
| Eff\_date | Datetime |  |
| Exp\_date | Datetime |  |

## Ad Tool Access / Configuration

Once IPC account is created for Adtool, net step requires providing configuration data required for integration. This data varies from adtool to adtool. Following table will be populated for configurational data.  
Table Name: PDM\_ACCOUNT\_AD\_TOOL\_API\_ACCESS  
Followings are required columns.

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Type** | **Description** |
| account\_id | Varchar(128) | Foreign key |
| ad\_tool\_id | Int | Id of adtool, this is predefined id for each adtool. Please ceck adtool id lists for this Entery |
| seq\_id | Int | Incremental Seq id assigned to this record, for each tool , it starts with 1 |
| Api\_access\_property\_id | Int | Property\_id.id  Every Adtool has predefined set of properties which needs to be added for each tool. Please review AdTool required information section for api\_acce\_properties |
| api\_access\_property\_id | Nvarhar(1600) | Property value |
| Create\_id | Varchar(100) |  |
| Create\_date | Datetime |  |
| Modify\_id | Varchar(100) |  |
| Modify\_date | Datetime |  |

**Required Information for Each AdTool:**

1. **Google Ads**Following information are required in IPC to integration Google Ads
   1. **Budget Order Creation**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| **Developer token** |  | 2 |
| **Mcc** |  | 3 |
| **Email Address** |  | 4 |
| **Client Secret** |  | 5 |
| **Client ID** |  | 6 |
| **Access Token** |  | 7 |
| **Refresh Token** |  | 8 |
|  |  | 9 |
|  |  | 10 |
|  |  | 11 |
|  |  | 12 |
|  |  |  |

* 2. **Delivery**

1. **GCM**
   1. **UI integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| Access\_token |  | 10 |
| Refresh\_token |  | 11 |
| File store path |  | 13 |
| Profile id |  | 14 |

* 1. **Delivery Pull**
     1. **Report ID**

1. **DV360**
   1. **UI Integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| Access\_token |  | 10 |
| Refresh\_token |  | 11 |
| File store path |  | 13 |
| Profile id |  | 14 |

* 2. **Delivery**
     1. **Report ID**

1. **Microsoft Advertising**
   1. **UI Integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| Dev Token |  | 2 |
| Master Account |  | 3 |
| Refresh\_token |  | 10 |
| Access token |  | 12 |
| Store path |  | 13 |

* 1. **Delivery**
     1. **Report ID**

1. **TheTradeDesk**
   1. **UI Integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| Access\_token |  | 10 |
| Refresh\_token |  | 11 |
| File store path |  | 13 |
| Profile id |  | 14 |

* 1. **Delivery** 
     1. **Schedule ID**

1. **Yahoo**
   1. **UI Integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| Access\_token |  | 10 |
| Refresh\_token |  | 11 |
| File store path |  | 13 |
| Profile id |  | 14 |

* 1. **Delivery**
     1. **Report ID**

1. **LinkedIn**
   1. **UI Integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| Access\_token |  | 10 |
| Refresh\_token |  | 11 |
| File store path |  | 13 |
| Profile id |  | 14 |

* 1. **Delivery**
     1. **Report ID**

1. **Twitter**
   1. **UI Integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| Access\_token |  | 10 |
| Refresh\_token |  | 11 |
| File store path |  | 13 |
| Profile id |  | 14 |

* 1. **Delivery**
     1. **Report ID**

1. **FaceBook**
   1. **UI Integration**

|  |  |  |
| --- | --- | --- |
| **Entity** | **Description** | **API\_access\_property\_id** |
| App Name |  | 1 |
| Access\_token |  | 10 |
| Refresh\_token |  | 11 |
| File store path |  | 13 |
| Profile id |  | 14 |

* 1. **Delivery**
     1. **Report ID**

# IPC Changes - API(s).

Following are the APIs which will be created in IPC for communication between Nexelus and IC.

1 - IPC…ConfigurationService/ConfigureIPC  
Request: CompanyKey  
This call will insert Data into following 3 tables.PDM\_CONSUMER

PDM\_CONSUMER\_ACCOUNT

PDM\_ACCOUNT\_USER  
 **Response**: IPC account number, uid, password.

2 - IPC…ConfigurationService/populateAccountSummary  
 **Request**: AccountID, uid, password  
This function will return all configured integrations for this account.. **Response**: IPC account number, uid, password.

3 - IPC…ConfigurationService/AddIntegrationAccount  
 **Request**: IntegrationType, List of propertis.  
This call will insert data into Api keys tables and will create an integration for specific adtool. **Response**: Addtool ID, Seq ID, Tag

# Nexelus Changes:

AdTool Integration(s) UI:  
  
New UI “Adtool Integrations” will be add in Nexelus.  
UI Link in Menu:

AdTool Integrations (This will be variable name) and it’s security and rights can be managed from Security Rules. By default, no one will have access to this UI.

once user clicks on ”AdTool Integrations” link, system will behave as following.

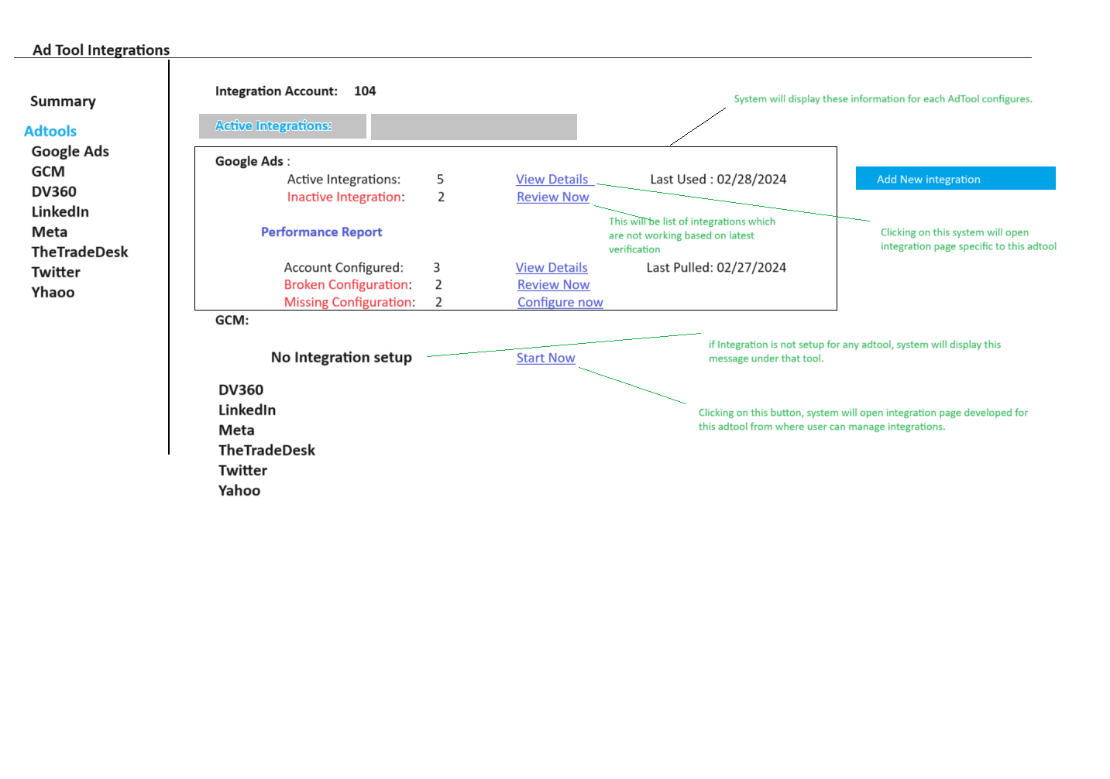
#### In the Case of Non-Media Company.

if company is not registered as Media Company (based on license to media module) , system will display following UI and message. **A screenshot of a computer

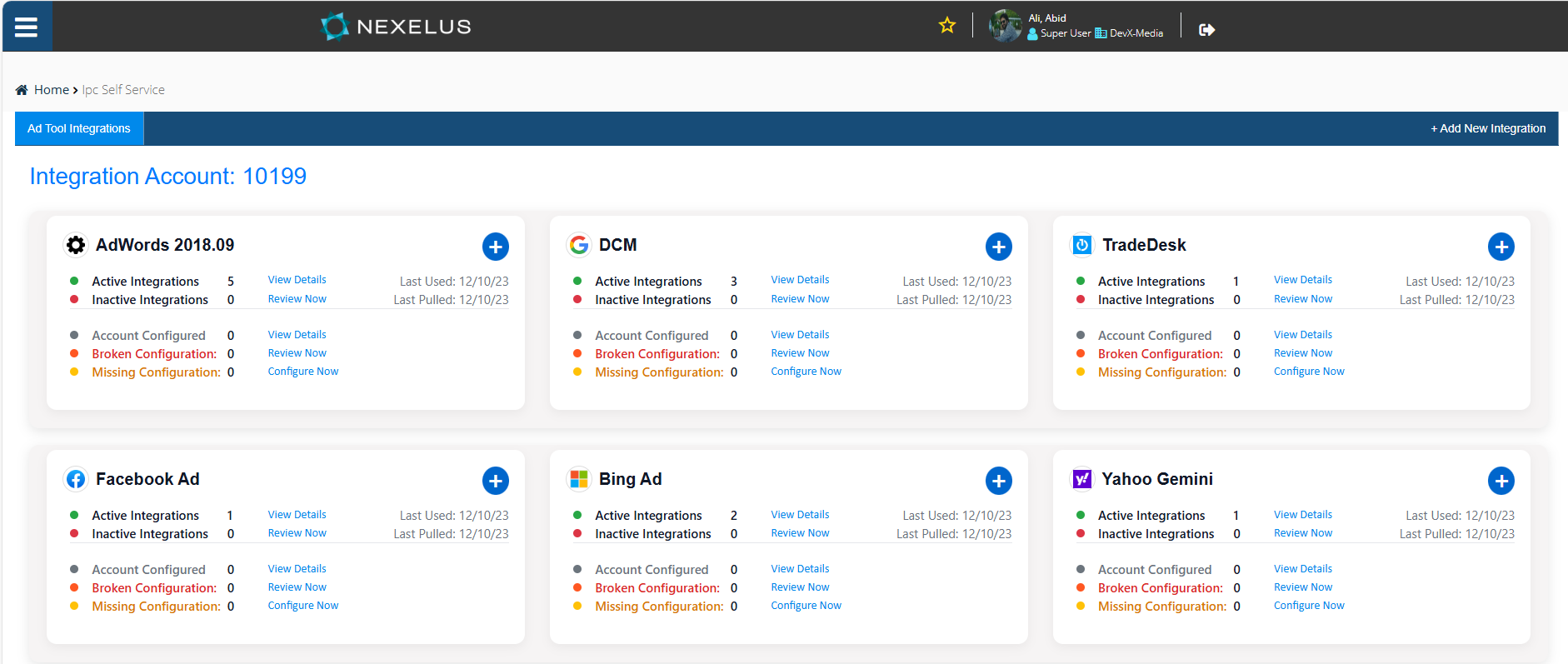
Description automatically generated**

* 1. Media Company with no IPC account Setup  
     if company is media company but IPC account is not created against the Company, system will display following message on UI.  
     A close-up of a computer screen

     Description automatically generated
  2. Media Company with IPC account tagged.  
       
     if company and is media company and IPC account is created and stamped on company, system will display summary page which will display current enabled /configured integration information.



Update UI for Ref – 14-July-2025:



This UI will have the following information for each AdTool.  
  
**Active Integrations:** This will be number of integrations which are active and working for specific adtool. The system will monitor AdTool settings on daily basis and will update this number.   
  
**Inactive Integrations.** Number of integrations which are configured but not working properly based on monitoring verification on daily basis.  
  
**Performance report:**

* Account Configured:
  + Number of accounts configured for performance reports out of configured accounts. System will check performance report for each account if account is inactive based on monitoring tool.
* Broken Configuration:
  + It will be the number of accounts for which the performance report is not working properly based on daily monitoring.
* Missing configuration
  + Number of account for which the performance report is not setup.

Action Buttons on UI.  
  
Following will be action buttons on UI.  
  
**Add New Integration**: Once user clicks on this button, system will open integration page for specific tool in new mode, which will allow users to add new integration for specific tool.

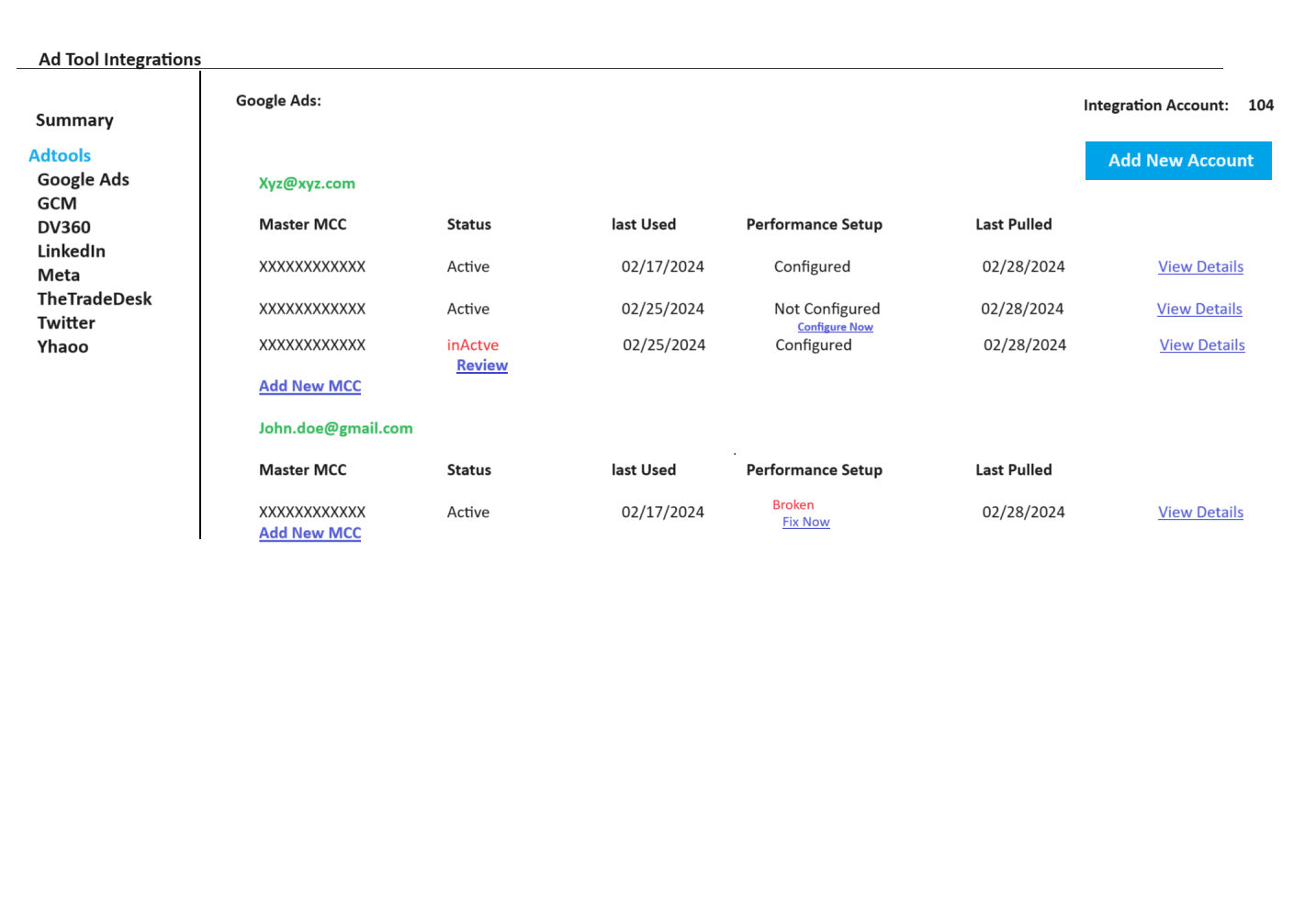
**View Details:** Once user clicks on this button, the system will open integration page which will display details of all integrations setup for specific Adtool.

**Review Now** (Inactive Integration): Once user clicks on this button, system will open integration page for specific site and display all integrations which are not working properly for specific tool.  
  
**Review Now** (Broken Configuration \_ Performance report): Once user clicks on this button, system will open integration page for specific site and display all Performance reports which are not working.  
  
**Configure Now**: Once user clicks on this button, system will open integration page with list of all integration for specific tool for which Performance report is not configured.

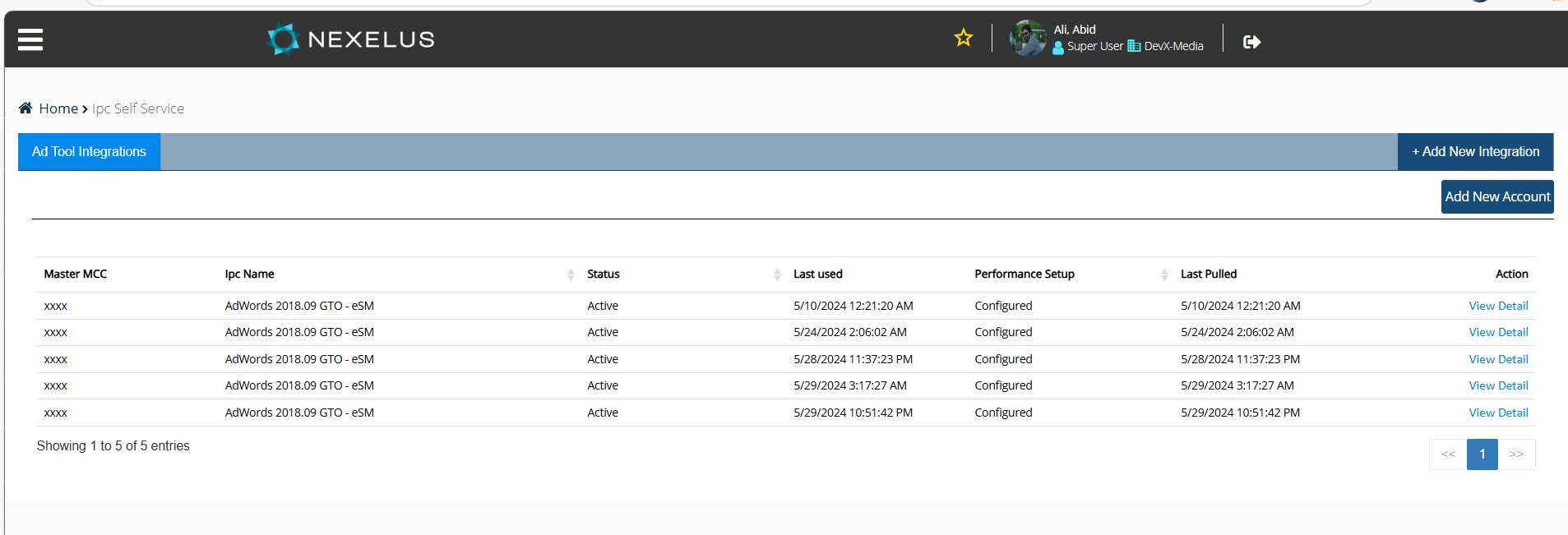
**Integration Page for specific AdTool:**User can click on specific tool to view detail of integrations setup for specific tool. Once user clicks on tool name, system will display following UI.  
  
Following will be the information on this page.  
  
  
**List Grid**: System will display all configured integrations in a grid.  
grid will have following columns.  
  
**MCC Account** (in case of Google Ads, this column can be changed for all integrations). This will be MCC for integration setup.  
  
**Status:** status of integration if it is active of inactive based on last monitoring.  
  
**Last Used:** Date time when last column was used, system will keep track of each integration when and by whom it was used last time.  
  
**Performance Setup:** This column will have 3 value possible as following.

**Configured**: if performance report is configured and working properly.  
**Not Configured**: if Performance report is not configured for account, system will display not configured in this case and will allow user to configure it by clicking on “Configure Now” button, which will open configuration page for performance report.

**Broken**: If Performance report is configured but not working properly, System will display “Fix Now” option to fix the broken configuration.



Update UI for Ref – 14-July-2025:



**Add New Configuration (**Google Ads**):** this UI will allow users to add new integrations. Below if the sample page for Google ads, all integrations will work the same way except the information can be different for each page.  
  
This page will have 2 sections, one for integrations, and other one for performance report, should we have single save button of different for each section??  
  
Once user provide the required information and click save, system will display list of all companies setup and will provide user an option to select the companies for which user wants to configure this integration (current company will be selected by default). once user click save, system will create integration in IPC and Nexelus and will verify if integration is working by pulling accounts from Google Ads. If integration is not working, system will display proper error message and will stamp it as non-functional. Screen will go into “Update Mode” and will display “Update” And “Reverify” option. Which will allow users to update the settings and/ or reverify.

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

System Monitoring:  
  
Following columns will be added in integration table on Nexelus side.

1. total\_no\_of\_entities\_sycned
   1. it will be incremental column which system will increment on every unique push which means if will be incremented only on creation of campaign / BO not on revision.
2. last\_entity\_pushed\_date
   1. date when last entity was pushed, this will include revision push as well, means system will update this date on every sync with Adserver regardless of if it is creation of revision.
3. Last\_verified\_on
   1. Date when integration was last time verified.
4. verification\_status
   1. status of verification – will have following 2 values
      1. 0
         1. Means verification was unsuccessful.
      2. 1
         1. Means verification was successful.
5. verification\_error
   1. varchar field, error description if verification was not successful.

The following columns will be added for performance report monitoring.

1. Daily\_performance\_report\_status
   1. This will be a column which will indicate the status of daily performance. This column will have the following values.
      1. **0** 
         1. Not configured
      2. **1**
         1. Working properly.
      3. **2**
         1. Configured but not working.
2. Daily\_performance\_report\_status\_verified\_on
3. Daily\_performance\_report\_verification\_error
4. Monthly\_performance\_report\_status
   1. This will be a column which will indicate the status of daily performance. This column will have the following values.
      1. **0** 
         1. Not configured
      2. **1**
         1. Working properly.
      3. **2**
         1. Configured but not working.
5. monthly\_performance\_report\_status\_verified\_on
6. monthly \_performance\_report\_verification\_error

Note: Please note that Nexelus will maintain a history of these verifications as well.

**Monitoring Process:**Following will be the process to monitor integration and performance reports verification.  
  
**1 – Integrations.**  
  
On every login in application, system will check if integrations were verified today or not. And if integrations are not verified today, the system will run a thread which will verify all integrations by trying to pull the accounts / advertisers for each integration and will set their statuses accordingly. Please note that this thread shouldn’t block login process and shouldn’t impact on performance of the system.  
  
for this, we may need a last verification date column at the company level in pdi\_sites. Please discuss with DB.

**2 – Performance verification.**we pull deliveries on daily basis based on schedule. This is to continue working the same way it is currently working. However, currently we are scheduling schedule for individual client and adTool separately.  
  
Nexelus will update the performance pull application to pull performance data based on this configuration for all clients / companies.  
  
during the performance pull based on schedule, system will update status of each performance report in their respective columns which will later be used to verify the data.  
  
system will perform following checks on each report.  
  
 1- Report ran successfully without throwing any exception.

2- report had valid data.

a – if there is no report configured, system will mark it as not configured.

b – for the daily report, it should have data from last date.

c – for last month’s report, report should have data for last month starting from 1st of last month. Please remember adtool may not have any data for 1st of last month, that’s why if it contains data for last months for 4-5 days, we will consider it successful pull.

**Current Process Flow to Set Up an iPC Account [July-14-2025]**

1. **Client provides Ad Tech Platform API credentials.**
2. **We set up the iPC account for the client using the following database tables:**

a. **pdm\_consumer**

* + Set up client (consumer) information.

b. **pdm\_consumer\_account**

* + Set up client account details and note down the **Account ID** for use in the Nexelus client database.

c. **pdm\_account\_user**

* + Set up a user account under the client account and note down the **User ID** and **Password** for use in the Nexelus client database.

d. **pdm\_account\_ad\_tool**

* + Establish the relationship between the client account and the Ad Tech platform.
  + Note down the **Access\_Tag (Token)** for use in the Nexelus client database.

e. **pdm\_account\_user\_ad\_tool**

* + Define the relationship between pdm\_account\_user and pdm\_account\_ad\_tool.

f. **pdim\_ad\_tool\_api\_access**

* + This is a meta table that defines the required fields to connect to the Ad Tech Platform via API.

g. **pdm\_account\_ad\_tool\_api\_access**

* + Stores the actual Ad Tech Platform API credentials (e.g., Client ID, Secret Key), structured according to the meta information from step **f**.
  1. **We collect the Account ID, User ID, Password, and Access\_Tag from step 2, and configure the integration in the client database using the company setup screen as shown below.**