

SedonaAPI and SedonaWeb 2.0 - 1.41.1 - 4/15/2022

01/05/2024 12:03 pm EST

Special Upgrade / Install Instructions:

If your company uses the SedonaCloud API, IT will update the SedonaCloud version at the same time as your SedonaOffice upgrade. This is to ensure compatibility with the all modules utilizing SedonaCloud.

Supported Environments:

Version Support

**** This version of SedonaCloud requires SedonaOffice version 6.2.0.4 or above ****

Minimum System Requirements:

- Server is on Microsoft .Net 4.6.1

Bugs:

Note: Development Item Number in Parenthesis (), Customer Case Number in brackets []

Title

Issue:

Solution:

api/paymentmerchant/addcreditcard (and addbankaccount) merges LastName, FirstName in Customer Bill To. (26997)

Issue: If the referenced CustomerBillTo for the CustomerBillId in the POST payload has a Business Name in the format of LastName, Firstname, the endpoint contains logic that will reformat that name and save it back to the database as 'FirstNameLastname'. It should be 'FirstName LastName'.

Solution: Fixed POST api/paymentmerchant/addcreditcard and api/paymentmerchant/addBankAccount that changes a CustomerBillTo Business Name saved in LastName, Firstname is reformatted and saved 'FirstName LastName' not 'FirstNameLastname'

Get - ACH - blank response even when ACHID is provided. (27033)

Issue: The GET api/ach endpoint does not always return data even when the ACHID passed is valid.

Solution: Changed the ACH and AR_ACH models to no longer require the DepositCheckId and PreviousAchId. Also, changed the model

mapping to make the navigation property relationships for these IDs to be optional.

Since database investigations of actual customer database have shown that sometime the ID=1 record does not exist in the corresponding database table, having these navigation relationships as Required resulted in the GET endpoint not returning some of the AR_ACH records because the INNER JOIN of a required relationship resulted in no data.

POST api/APBill allows duplicate invoice numbers. (26605) [41499]

Issue: Using POST to /api/APBill endpoint is not validating invoice number for duplicate invoices.

Solution: Added a validation to make sure that the invoice number isn't in use more than once per vendor. If an invoice is posted that uses an invoice number that is already in the system for that vendor, a Status 400 Bad Request is returned.

POST /api/Credit Error BillCycle required. (26606) [40987]

Issue: Receiving error when using POST /api/Credit without passing a BillCycle.

Solution: BillCycle field was marked as required. Blank values were not being accepted as valid even though SedonaOffice allows this field to be blank. Corrected this issue.

POST /api/CreditDetail/ requires InvoiceID to be set. (27023)

Issue: POST CreditDetail - Does not work if type isn't I. Requires InvoiceId to be set, which typically would be InvoiceId 1 if not type I. If 1 is provided, returns error about can't apply to that invoice. API also doesn't support using InvoiceNumber despite it being in the object model, InvoiceID must be used.

Solution: Setting the defaults for fields tied to the CreditDetail endpoint and allowing code fields present in the view model to be used (examples, Branch Number, Category Code, Invoice Number)

POST /api/Credit Error 400 received if TaxAmount = 0. (26882) [40014]

Issue: Using POST to api/ARCredit endpoint message The formal parameter \"@tax_exempt\" was not declared as an OUTPUT parameter, but the actual parameter passed in requested output.

Solution: Added the TaxExempt property to CreditItem POST view model and the AR_Credit_Item model. Added the missing @tax_exempt stored proc parameter to the method that executes the stored procedure.

POST - Sum of Credits does not match credit amount if tax amount is omitted, even if untaxable credit items. (27024)

Issue: TaxAmount is coming in as NULL, which is throwing off the calculation to determine if the error needs to be thrown.

Solution: Validations were updated and the defaults for fields were set. Where applicable, code lookup fields are now able to be used.