

Xavier courtemanche - bug de currection

Journal Général

Description du problème: les identifiants des factures sont données en odres d'apparitions des écritures dans le J.G. Le système detecte que la "bonne" écriture est la facture n. 188 puisque les comptes sont un match parfait avec ce que l'étudiant à écrit. Ce problème origine de l'erreur de l'étudiant (n'a pas écrit le bon compte pour une écriture) et du fait que nous ne faisons pas multiples tentatives de trouvé la bonne facture associé à l'écriture.

Solution: Faire multiple tentatives d'associations.

The screenshot shows a financial software interface for 'xavier courtemanche' with the company 'Entreprise: Xavier & Company'. The main window displays 'Correction - E. Quiz - Le journal général - avec taxes' with a balance of 79.5%. A 'Détails des erreurs' popup is visible over the journal entries.

Date	Nom de compte	Numéro de compte	Débit	Crédit
16 déc. 2024	Services rendus		588.05	
	Clients			588.05
	(Chèque n° 2165)			
	(Autres précisions)			
28 déc. 2024	Frais de four		769.61	
	TPS à recevoir		33.47	
	TVQ à recevoir		66.77	
	Fournisseurs			669.37
	(Facture n° 516)			
	(Autres précisions)			
29 déc. 2024	Encaisse		415.38	
	Services rendus			361.28
	TPS à payer			18.06
	TVQ à payer			36.04

Détails des erreurs

La valeur attendue : Facture n° 188 (-0.9%)

xavier courtemanche
 Entreprise: Xavier & Company

Correction - E. Quiz - Le journal général - avec taxes

79.5 % Journal général Page 2/2

Date	Nom de compte	Numéro de compte	Débit	Crédit
30 déc. 2024	Frais de fournitures d'entretien		490.23	
	TPS à recevoir		24.51	
	TVQ à recevoir		48.90	
	Fournisseurs			563.64
	(Facture n° 188)			
	(Autres précisions)			
30 déc. 2024	Fournisseurs		563.64	

Détails des erreurs X

La facture n° 188 provenant de Rotor inc. est déjà utilisé(e). (- 11.1 %)

xavier courtemanche
 Entreprise: Xavier & Company

Correction - E. Quiz - Le journal général - avec taxes

79.5 % Journal général

Date	Nom de compte	Numéro de compte
	(Autres précisions)	
16 déc. 2024	Services rendus	
	Clients	
	(Chèque n° 2165)	
	(Autres précisions)	
28 déc. 2024	Frais de fournitures d'entretien	
	TPS à recevoir	
	TVQ à recevoir	
	Fournisseurs	
	(Facture n° 516)	
	(Autres précisions)	
29 déc. 2024	Encaisse	
	Services rendus	

Fenêtre du corrigé

Journal général

Date	Nom de compte
30 déc. 2024	Frais de fournitures d'entretien
	TPS à recevoir
	TVQ à recevoir
	Fournisseurs
	(Facture n° 188)
	(Autres précisions)
30 déc. 2024	Fournisseurs
	Encaisse

Marie-Josée Nadon | Simulation sélectionnée | Étudiant xavier | Fenêtre du corrigé

Groupe 101 - A24

Correction - E. Quiz - Le journal général - avec ta

79.5 % Journal général

Date	Nom c
	(Chèque n° 2165)
	(Autres précisions)
28 déc. 2024	Frais de fournitures d'entretien
	TPS à recevoir
	TVQ à recevoir
	Fournisseurs
	(Facture n° 516)
	(Autres précisions)

Date	
28 déc. 2024	Frais de fourniture
	TPS à recevoir
	TVQ à recevoir
	Encaisse
	(Facture n° 516)
	(Autres précisions)

Les montants ne sont pas bon aussi, ce qui peut être une autre cause du problème.

28 déc. 2024	Frais de fournitures d'entretien	5 550	669.37			
	TPS à recevoir	1 105	33.47			
	TVQ à recevoir	1 110	66.77			
	Encaisse	1 010		769.61		
	(Facture n° 516)					
	(Autres précisions)					

					769.61	
					33.47	
					66.77	
						669.37

Description détaillé de la solution proposé

L'algo actuel se fait en une seul passe pour l'association des écritures avec factures (si la facture ne match pas parfaitement la facture indiqué par l'étudiant, prend la prochaine meilleur qui à un taux de match supérieur à 65%). Ce qui peut causé une double erreur puisqu'une écriture peut "voler" une bonne facture à une autre écriture

```

4 function byModuleGradeModule(moduleType, moduleTypeData, actSettings, moduleSettings, pagesSettings) {
1834   function computeGeneralJournalScore(modulePageUid) {
1868     }
1869
1870     var possibleAccountingClosingSectionUids = [];
1871     var wantedExplanationBySectionUids = {};
1872     var minMatchRateToDetermineWantedExplanation = 0.65;
1873
1874     var sectionAccountsAndImpactTypesBySectionUids = {};
1875
1876     for (var sectionUid of tableData.sectionsIndex) {
1877       var sectionData = tableData.sections[sectionUid];
1878       if (!sectionData.rowsIndex) {
1879         continue;
1880       }
1881
1882       //find the explanation that match the rows of the section or the current explanation when no perfect match are found
1883       var sectionAccountsAndImpactTypes = [...];
1886     };
1887     sectionAccountsAndImpactTypesBySectionUids[sectionUid] = sectionAccountsAndImpactTypes;
1888     for (var rowUid of sectionData.rowsIndex) {...}
1889   }
1890
1891   //filter out section associated with accounting closing
1892   var foundRelationMatch = false;
1893   for (var relationUid in relationAccountsAndImpactTypesByRelationUids) {
1894     var relationAccountsAndImpactTypes = relationAccountsAndImpactTypesByRelationUids[relationUid];
1895
1896     var relationMatchRate = getMatchRateOfTwoAccountsAndImpactTypes(sectionAccountsAndImpactTypes, relationAccountsAndImpactTypes, false);
1897     if (relationMatchRate == 1) {
1898       foundRelationMatch = true;
1899       possibleAccountingClosingSectionUids.push(sectionUid);
1900       break;
1901     }
1902   }
1903   if (foundRelationMatch) {
1904     continue;
1905   }
1906
1907   var currentBestBillOrFormUid = null;
1908   var currentBestMatchRate = 0;
1909
1910   var explanationUid = null;
1911   if (inputsHasValue(sectionData.inputs, "explanation_account_uid")) {
1912     explanationUid = sectionData.inputs["explanation_account_uid"].value;
1913     if (billAndFormAccountsAndImpactTypesByUids[explanationUid]) {
1914       var billAccountsAndImpactTypes = billAndFormAccountsAndImpactTypesByUids[explanationUid];
1915       currentBestBillOrFormUid = explanationUid;
1916       currentBestMatchRate = getMatchRateOfTwoAccountsAndImpactTypes(sectionAccountsAndImpactTypes, billAccountsAndImpactTypes, true);
1917     }
1918   }
1919
1920   if (currentBestMatchRate != 1) {
1921     for (var billOrFormUid in billAndFormAccountsAndImpactTypesByUids) {
1922       if (billOrFormUid == explanationUid) {
1923         continue;
1924       }
1925       var billOrFormAccountsAndImpactTypes = billAndFormAccountsAndImpactTypesByUids[billOrFormUid];
1926
1927       var billOrFormMatchRate = getMatchRateOfTwoAccountsAndImpactTypes(sectionAccountsAndImpactTypes, billOrFormAccountsAndImpactTypes, true);
1928
1929       if (billOrFormMatchRate >= minMatchRateToDetermineWantedExplanation && billOrFormMatchRate > currentBestMatchRate) {
1930         currentBestBillOrFormUid = billOrFormUid;
1931         currentBestMatchRate = billOrFormMatchRate;
1932         if (billOrFormMatchRate == 1) {
1933           break;
1934         }
1935       }
1936     }
1937   }
1938
1939   //sections not in wantedExplanationBySectionUids or possibleAccountingClosingSectionUids
1940   //are section without an explanation and we are not able to find a good match
1941   if (currentBestBillOrFormUid) {
1942     wantedExplanationBySectionUids[sectionUid] = currentBestBillOrFormUid;
1943   } else {
1944     //since it doesnt have currentBestBillOrFormUid the explanation is about an accounting closing
1945     possibleAccountingClosingSectionUids.push(sectionUid);
1946   }
1947 }
1948 }

```

La solution que je propose afin de résoudre ce problème va comme suit (en pseudo code):

```

currentThreshold = 1
nextThreshold = 0
wantedExplanationBySectionUids = {}listOfSectionUids = copyOf(tableData.sectionsIndex)
billUidRemainings = copyOf(billUids)

```

```

tant que listOfSectionUids est pas vide
for sectionUid of listOfSectionUids  for billUid of billUidRemainings (ce ne sera pas
exactement représenté comme cela dans le code)
    threshold = getMatchRate(sectionUid,
    billUid)
        if (threshold == currentThreshold)
wantedExplanationBySectionUids[sectionUid] = billUid
listOfSectionUids.remove(sectionUids)
billUidRemainings.remove(billUid)                break billUidRemaining
loop

        nextThreshold = max(threshold, nextThreshold)

currentThreshold = nextThreshold

```

Ce pseudo-algorithme devrait réduire les cas d'erreurs.

Afin d'implémenté ce pseudo code une bonne partie du code restera pareil.

Revision #4

Created Tue, Sep 24, 2024 2:16 PM by [Frédéric Messier](#)

Updated Tue, Jun 10, 2025 1:26 PM by [Frédéric Messier](#)