Technical Documentation

XERO Integration

Satish Darade/Pragmatic Techsoft Pvt. Ltd.

15th December, 2016

INDEX

Introduction

This technical document consist of all the technical information for the modules developed for Xero Integration (Xero Accounting) project. It consist detailed information about fields and methods used in all the modules. This project consist of following modules.

1. XERO Integration [prag_xero_accounting]

In next section you will see module wise information

1. XERO Integration (prag_xero_accounting)

This module initially developed to meet the requirements of all business units of xero accounting. It consist of major changes which is common in all business units.

1. Account Invoice

This file consist of classes and methods related with account invoice object. List of classes and methods are as bellow.

1) Invoice [prag_xero_accounting/models/account_invoice.py]

Class account_invoice(models.Model)

Fields :

xero_exported(Boolean) : it will set the invoice is exported or not in xero.

xero_invoice_no(Char) : this field update when invoice exported to xero. This will fill xero invoice no.

xero_invoice_id(Char) : its store unique invoice id from xero.

Methods :

def copy() : inherited to set xero exported false when duplicate in record.

def _get_invoice_lines(lines) : this method accept ids and set the invoice lines in list of dict. 1st is check product is

exported or not to xero if not then it will call product method to export it 1st to xero.

@return: list of dict

def export_invoice(invoices) : this method accept invoices ids from wizard and according to given invoice its fetch the all invoice information from odoo/database and set combine in list of dictionary and once information get all it will do a bunch of 100 invoice record and after bunching it will send or call the next function one by one with bunched invoices to export invoices to xero.

@return: list of dict

def xero_invoice_export(Invoice_list_dic) : this method accept list of dect of invoice information and invoices object list. We established the xero connectivity using calling company get_xero_connection() method and once connection get successful we send the all information of invoices to xero. Xero will create invoices. In that method we also write or update invoice information for xero_export,xero_invoice_no,xero_invoice_id etc in odoo invoice, purpose of that is after exporting many invoices to xero, that invoices will not be repeated for next invoice export attempt.

2. Product

This file consist of classes and methods related with product object. List of classes and methods are as bellow.

1) Product [prag_xero_accounting/models/product.py]

Class product_product(models.Model)

Methods :

def copy() : inherited to set xero exported false when duplicate in record.

def write() : if changes some fields of product then set xero_update field to False.

def _check_product_exported() : this method check the current product is exported or not in xero if not then it will call export_product() method.

def export_product() : this method accept product id and according to given product id its fetch the all product information from odoo/database and set combine in list of dictionary and after fetching all information it will send to xero. Xero will create product and send that xero product id which will create in xero.after getting response we update further information to odoo product which are exported successfully in xero.

def _get_sales_details_product(lines) : this method collect the information of sales details product.

@return: dict

def _get_purchase_details_product(lines) : this method collect the information of purchase details product.

@return: dict

Class exported_product(models.Model)

Fields :

company_id(Many2one) : it associated with company.

product_id(Many2one) : it associated with Product.

xero_product_id(Char) : it will store xero product id.

xero_update(Boolean) : it will store product exported or not.

3. Partner

This file consist of classes and methods related with partner object. List of classes and methods are as bellow.

1) Partner [prag_xero_accounting/models/res_partner.py]

Class account_invoice(models.Model)

Fields :

ref(Char) : it will store unique partner reference.

Methods :

def _get_contact() : this method will check partner is exported or not to xero.

@return : dict (contact id)

def _check_partner_exported() : this method check partner exported or not to xero if exported then it will get stored contact id from odoo, otherwise it will call a next method which identify partner is present or not in xero.

@return : object

def _check_partner_exported() : this method check given partner is available or not in xero if yes then it will create exported partner record to odoo, if partner not found in xero then this method call _export_partner() method to export partner to xero.

def _export_partner() : this method accept partner id and according to given partner id its fetch the all partner information from odoo/database and set combine in list of dictionary and after fetching all information it will send to xero. Xero will create contact and send that xero contact id which will create in xero. after getting response we update

further information to odoo related to partner which are exported successfully in xero.

def write() : if changes some fields of partner then set xero_update field to False.

Class exported_partner(models.Model)

Fields :

company_id(Many2one) : it associated with company.

partner_id(Many2one) : it associated with Partner.

xero_partner_id(Char) : it will store xero contact id.

xero_update(Boolean) : it will store contact exported or not.

4. Comany

This file consist of classes and methods related with company object. List of classes and methods are as bellow.

1) Company [prag_xero_accounting/models/res_company.py]

Class res_company(models.Model)

Fields :

consumer_key(Char) : it will store consumer key which provided by xero.

private_key_file(Char) : it will store path of private key file.

Methods :

def get_xero_connection() : this method will established connection between odoo to xero.

5. Xero Logs

This file consist of classes and methods related with xero logs object. List of classes and methods are as bellow its keep track of error occurred when exporting invoices.

1) Xero Logs [prag_xero_accounting/models/xero_logs.py]

Class xero_logs(models.Model)

Fields :

name(Char) : it will store description of error.

invoice_id(Many2one) : it associated with account invoice.

purchaseorder_id(Many2one) : it associated with purchase order.

product_id(Many2one) : it associated with product.

partner_id(Many2one) : it associated with partner.

import_count(Integer) : it will store no of payments imported.

expoart_count(Integer) : it will store no of invoice export.

6. Payments

This file consist of classes and methods related with payment object. List of classes and methods are as bellow.

1) Payments [prag_xero_accounting/wizard/import_payments.py]

Class account_payment(models.Model)

Fields :

xero_paymentUp_date(Datetime) : it will store date and time at the time of imported payments or payments made through xero.

xero_payment_code(Char) : it will store xero payment code.

Class wiz_import_payments(models.TransientModel)

Methods :

def import_payments() : this method will import the all payments to odoo which those invoices which are made payment through xero.