Project Oracle Application

***[Invoice Payment Control Tower]***

***Vementor – Interfacing To Oracle***

v.1, Sep 2020

**PT. Karsa Lintas Buwana**

# Lembar Persetujuan

|  |  |  |
| --- | --- | --- |
|  |  |  |
| User Representative |  | Tanggal ( ) |
|  |  |  |
| Solution Architech |  | Tanggal ( ) |

# Historis Dokumen

| **VERSI** | **DIKELUARKAN PADA** | **KETERANGAN** |
| --- | --- | --- |
| 0.1 | 14 Oct 2020 | Oracle API Program |
| 0.2 |  |  |
| 0.3 |  |  |
|  |  |  |

Daftar Isi

[Lembar Persetujuan 2](#_Toc53739980)

[Historis Dokumen 3](#_Toc53739981)

[Pendahuluan 5](#_Toc53739982)

[1.1. DESKRIPSI 5](#_Toc53739983)

[EDIT KLAIM 6](#_Toc53739984)

[2.1. FSD Validate RFP 6](#_Toc53739986)

[2.1.1 Flow Process 6](#_Toc53739987)

[2.1.2 Program 6](#_Toc53739988)

[2.2. TSD Design 7](#_Toc53739989)

[2.3. QUERY FOR PROCEDURE VALIDATE INVOICE 8](#_Toc53739990)

[2.4. FSD Reject To Cancel Invoice 9](#_Toc53739991)

[2.4.1 flow process 9](#_Toc53739992)

[2.4.2 Validasi dan Package 9](#_Toc53739993)

[2.5. TSD 10](#_Toc53739994)

[2.6. Query yang di gunakan 10](#_Toc53739995)

# Pendahuluan

## DESKRIPSI

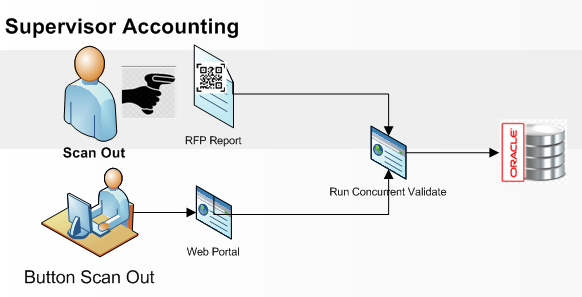
Dokumen ini menjelaskan secara teknis tekait function-function yang perlu ada di system yang merupakan penjabaran/penjelasan dari dokumen FSD dan BRD.

# EDIT KLAIM



## FSD Validate RFP

### Flow Process

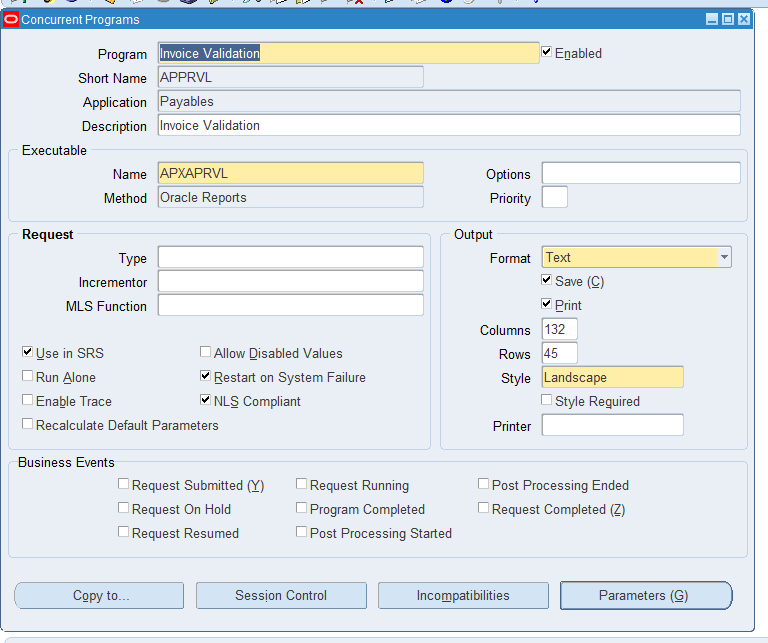


**Description :**

User yang memiliki role Supervisor Accounting melakukan proses Scan Out melalui barcode scanner atau web button setelah selesai melakukan proses pemeriksaan atas invoice tersebut. Secara background proses, akan memanggil program Validate RFP untuk mengubah status RFP menjadi ‘validate’

### Program

Nama program di Oracle yang digunakan untuk melakukan proses validate Invoice.



Script to call concurrent :

FND\_REQUEST.SUBMIT\_REQUEST( 'SQLAP',

'APPRVL',

'Invoice Validation',

SYSDATE,

FALSE,

NULL,

'ALL', *-- Option*

v\_batch\_id );

## TSD Design

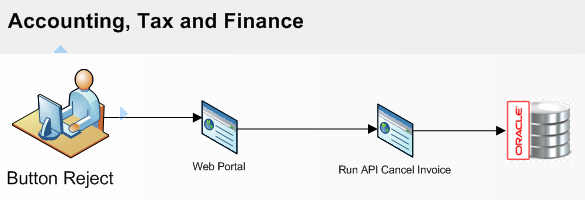
|  |  |  |
| --- | --- | --- |
| Object Name | Object Type | Description |
| XXVMT\_INTF\_INVOICE\_PKG | Package | Nama package validate rfp dama dengan cancel invoice |
| validate\_invoice\_rfp | procedure | Procedure untuk validate RFP Num |
| chk\_inv\_cancellable | Function | Untuk cek apakah invoice bisa di cancel |
| gl\_code\_descr | Function | Unuk mendapatkan description segment GL Code |
| cancel\_invoice\_rfp | Procedure | Procedure yang digunakan untuk cancel invoice |

## QUERY FOR PROCEDURE VALIDATE INVOICE

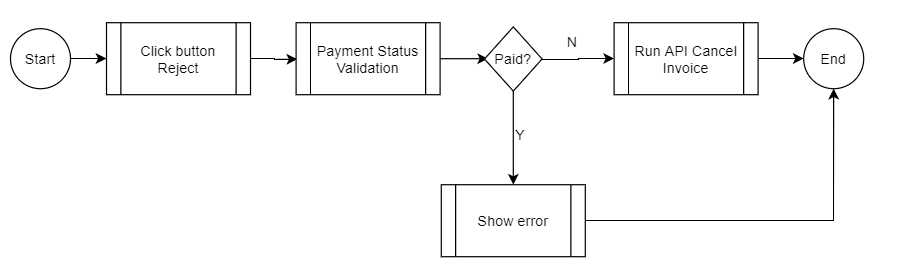
|  |
| --- |
| SELECT ab.attribute1 rfp\_no, ai.invoice\_id, ab.batch\_id  FROM ap.ap\_invoices\_all ai,  ap.ap\_batches\_all ab,  ar.hz\_parties hp,  fnd\_user fu,  ap\_suppliers aps,  iby\_ext\_bank\_accounts ieba,  iby\_ext\_banks\_v ieb  WHERE ai.batch\_id = ab.batch\_id  AND ai.party\_id = hp.party\_id  AND ab.created\_by = fu.user\_id  AND ai.vendor\_id = aps.vendor\_id  AND ap\_invoices\_pkg.get\_approval\_status (  ai.invoice\_id,  ai.invoice\_amount,  ai.payment\_status\_flag,  ai.invoice\_type\_lookup\_code) NOT IN ('CANCELLED')  AND ap\_invoices\_utility\_pkg.get\_approval\_status (  ai.invoice\_id,  ai.invoice\_amount,  ai.payment\_status\_flag,  ai.invoice\_type\_lookup\_code) NOT IN ('CANCELLED')  AND ieba.ext\_bank\_account\_id(+) = ai.external\_bank\_account\_id  AND ieba.bank\_id = ieb.bank\_party\_id(+)  AND ab.attribute1 = p\_rfp\_num  AND ai.org\_id = g\_org\_id  GROUP BY ab.attribute1, ab.batch\_id, ai.invoice\_id  ORDER BY ab.batch\_id; |
| fnd\_request.submit\_request ('SQLAP',  'APPRVL',  'Invoice Validation',  SYSDATE,  FALSE,  NULL,  'ALL', *-- Option*  v\_batch\_id); |

## FSD Reject To Cancel Invoice

### flow process



User yang memiliki role Finance, Accounting dan Tax memiliki otoritas untuk melakukan Cancel RFP melalui button “Reject”. Ketika button “Reject” di klik, maka akan melakukan validasi terlebih dahulu untuk memerika payment status dari Invoice tersebut. Apabila payment status sudah ‘Paid’ maka akan muncul notifikasi error **“Invoice is already paid. Please void the payment first”**, namun apabila status masih ‘Unpaid’ maka akan berlanjut ke jalankan API Cancel Invoice



### Validasi dan Package

Memeriksa payment status pada invoice terebut, apabila statusnya **“Paid”,** maka akan mengarahkan error **Invoice is already paid. Please void the payment first”**, namun apabila status masih ‘Unpaid’ maka akan berlanjut ke jalankan API Cancel Invoice.

Package Cancel Invoice :

AP\_CANCEL\_PKG.AP\_CANCEL\_SINGLE\_INVOICE

## TSD

|  |  |  |
| --- | --- | --- |
| Object Name | Object Type | Description |
| XXVMT\_INTF\_INVOICE\_PKG | Package | Nama package validate rfp dama dengan cancel invoice |
| validate\_invoice\_rfp | procedure | Procedure untuk validate RFP Num |
| chk\_inv\_cancellable | Function | Untuk cek apakah invoice bisa di cancel |
| gl\_code\_descr | Function | Unuk mendapatkan description segment GL Code |
| cancel\_invoice\_rfp | Procedure | Procedure yang digunakan untuk cancel invoice |

## Query yang di gunakan

|  |
| --- |
| SELECT ab.attribute1 rfp\_no,  ai.invoice\_id,  ab.batch\_id,  ai.payment\_status\_flag  INTO v\_rfp\_no,  v\_invoice\_id,  v\_batch\_id,  v\_payment\_status\_flag  FROM ap.ap\_invoices\_all ai,  ap.ap\_batches\_all ab,  ar.hz\_parties hp,  fnd\_user fu,  ap\_suppliers aps,  iby\_ext\_bank\_accounts ieba,  iby\_ext\_banks\_v ieb  WHERE ai.batch\_id = ab.batch\_id  AND ai.party\_id = hp.party\_id  AND ab.created\_by = fu.user\_id  AND ai.vendor\_id = aps.vendor\_id  AND ap\_invoices\_pkg.get\_approval\_status (  ai.invoice\_id,  ai.invoice\_amount,  ai.payment\_status\_flag,  ai.invoice\_type\_lookup\_code) NOT IN ('CANCELLED')  AND ap\_invoices\_utility\_pkg.get\_approval\_status (  ai.invoice\_id,  ai.invoice\_amount,  ai.payment\_status\_flag,  ai.invoice\_type\_lookup\_code) NOT IN ('CANCELLED')  AND ieba.ext\_bank\_account\_id(+) = ai.external\_bank\_account\_id  AND ieba.bank\_id = ieb.bank\_party\_id(+)  AND ab.attribute1 = p\_rfp\_num  AND ai.org\_id = g\_org\_id  GROUP BY ab.attribute1, ab.batch\_id, ai.invoice\_id  ORDER BY ab.batch\_id; |
| chk\_inv\_cancellable (v\_invoice\_id); |
| ap\_cancel\_pkg.ap\_cancel\_single\_invoice (  p\_invoice\_id => v\_invoice\_id,  p\_last\_updated\_by => p\_last\_updated\_by,  p\_last\_update\_login => p\_last\_update\_login,  p\_accounting\_date => v\_accounting\_date,  p\_message\_name => x\_message\_name,  p\_invoice\_amount => x\_invoice\_amount,  p\_base\_amount => x\_base\_amount,  p\_temp\_cancelled\_amount => x\_temp\_cancelled\_amount,  p\_cancelled\_by => x\_cancelled\_by,  p\_cancelled\_amount => x\_cancelled\_amount,  p\_cancelled\_date => x\_cancelled\_date,  p\_last\_update\_date => x\_last\_update\_date,  p\_original\_prepayment\_amount => x\_orig\_prepay\_amt,  p\_pay\_curr\_invoice\_amount => x\_pay\_cur\_inv\_amt,  p\_token => x\_token,  p\_calling\_sequence => NULL); |