House Account Services

#### Design Specifications

# Purpose:

The purpose of this document is to provide a comprehensive documentation of the House Account services that will be implemented. Key end to end features will be discussed to include java classes, database tables, tag libraries as well as key components and how these parts interact with each other and how the application functions. This document will serve as an agreement as to what will be delivered to the [internal] client/customer.

# Scope:

The scope of this document is restricted to describing how the House Account web services is designed and how it is able to achieve its intended goal.

# Audience:

The intended audience of this document are the customer, stake holders as well as any technical resources involved in current or future development of this application / web service.

# Requirements:

Design and implement the following web services:

* House Account Balance Web Service
* Payment Authorization Service

## Requirements (Email)

Hariharan Thirumalaisamy

Reply all|

Wed 4/12, 11:53 AM

Casmon Gordon;

Jyothi Pemmasani

Casmon

We will have to develop two new service for June 6th release to get balance & authorize house account payment. I have copied the request/response stubs in the following path.

\\safedata2\Enterprise\Middleware Services\Hari\houseAccount

I will explain you the details in person.

Thanks

Hari

# Design:

## Java Classes

The following java changes were made:

|  |  |  |
| --- | --- | --- |
| Class | Status | Comments |
| flowers.salta.helper.DeepLinkHelper | Deprecated |  |
| flowers.dao.DeepLinkDAO | Deprecated |  |
| flowers.salta.service.CanonicalService | Deprecated |  |
| flowers.salta.service.DeepLinkService | Deprecated |  |
| flowers.salta.helper.AggregateServicesHelper | Added |  |
| flowers.dao.AggregateServicesDAO | Added |  |
| flowers.dto.BalanceDTO | Added |  |
| flowers.dto.PaymentAuthDTO | Added |  |
| flowers.salta.resource.request.AuthorizeHouseAccount | Added |  |
| flowers.salta.resource.request.GetPaymentAuthRequest | Added |  |
| flowers.salta.resource.response.GetHouseAccountBalanceResponse | Added |  |
| flowers.salta.resource.response.AuthorizeHouseAccountResponse | Added |  |
| flowers.salta.service.AggregateServicesService | Added |  |
| flowers.webservice.constants.ApiServiceConstants | Modified |  |
| flowers.salta.resource.response.ObjectFactory | Modified |  |
| flowers.salta.resource.request.EsbSaltaServiceRequest | Modified |  |
| flowers.salta.resource.response.EsbSaltaServiceResponse | Modified |  |
|  |  |  |

Table 1 – Java classes that were modified

# Design Specifics

## Payment Authorization Service

## SQL:

Payment authorize select statement:

SELECT DECODE(SIGN(? - CREDIT\_LIMIT\_AMOUNT), 0, 200, -1, 200, 400) as APPROVAL\_CODE,

DECODE(SIGN(? - CREDIT\_LIMIT\_AMOUNT), 0, 'APPROVED', -1, 'APPROVED', 'REJECTED') as APPROVAL\_STATUS, CREDIT\_LIMIT\_AMOUNT FROM corp\_customers WHERE CUST\_TYPE = 'CUST' AND START\_DATE <= sysdate AND NVL(END\_DATE, sysdate) >= sysdate AND STATUS ='A' AND ACCOUNT\_NUMBER = ? ;

Audit table insert statement:

INSERT INTO PYMTAPPROVALAUDIT VALUES ((SELECT (NVL(MAX(ID),0) +1) as NEXTID FROM PYMTAPPROVALAUDIT), ?, ?, ?, ?, ?, ?, ?, ?, ?) ");

Update available credit balance statement:

UPDATE corp\_customers SET CREDIT\_LIMIT\_AMOUNT = (CREDIT\_LIMIT\_AMOUNT - ?) WHERE CUST\_TYPE = 'CUST' AND START\_DATE <= sysdate AND NVL(END\_DATE, sysdate) >= sysdate AND STATUS ='A' AND ACCOUNT\_NUMBER = ? ;

## NOTES:

* This service will use the SALTA database (only).
* The logic and math for the approval calculation will be done inside the embedded SQL query as JDBC calls are known to be more efficient, especially under load for calculation such as these. The SQL query will be wrapped inside a PreparedStatement to add security against SQL injection hacks among others.
* The approval process will be treated as an atomic transaction; so if any error occurs in any of the component steps, then the entire transaction will be rolled back.
* A successful authorization will be: if (amount <= available balance) then APPROVED otherwise REJECTED.
* Audit and historical information will be captured and stored in the SALTA database. (Please see appendix section for specifics about the audit table and fields that are being recorded.)

|  |  |  |
| --- | --- | --- |
| COLUMN | TYPE | COMMENTS |
| ID | NUMBER(10) | NOT NULL, PRIMARY KEY |
| ACCOUNT\_NUMBER | NUMBER(10) | NOT NULL |
| APPROVAL\_DATE | DATE | NOT NULL, defaults to current date |
| REQUEST\_AMOUNT | NUMBER(8,2) | NOT NULL, defaults to $0.00 |
| AVAILABLE\_BALANCE | NUMBER(8,2) | NOT NULL, defaults to $0.00 |
| ORDERID | NUMBER(10) | NOT NULL |
| SOURCESYSTEM | VARCHAR2(30) | NOT NULL |
| APPROVALCODE | VARCHAR2(5) | NOT NULL |
| APPROVALSTATUS | VARCHAR2(10) | NOT NULL |
| COMMENTS | VARCHAR2(1000) |  |

Table 2 – Payment Approval Audit Table (PYMTAPPROVALAUDIT)

|  |  |  |
| --- | --- | --- |
| **APPROVALCODE** | **APPROVALSTATUS** | **COMMENTS** |
| **200** | **APPROVED** | This is returned if the requested amount is less than or equal to the ‘CREDIT\_LIMIT\_AMOUNT’ amount. |
| **400** | **REJECTED** |  |
|  |  |  |

Table 3 – Approval Codes

# House Account Balance

## SQL:

SELECT CREDIT\_LIMIT\_AMOUNT FROM corp\_customers WHERE CUST\_TYPE = ‘CUST’ AND

CUST\_TYPE\_1 = ‘HC’ AND START\_DATE <= sysdate AND NVL(END\_DATE, sysdate + 5) > sysdate AND STATUS =’A’ AND ACCOUNT\_NUMBER = ? ;

## NOTES:

* This service will use the SALTA database.
* Salta has the account balance of the house account. The customer in question will be validated before a balance amount will be returned.
* The CREDIT\_LIMIT\_AMOUNT column will be used as the house account balance. This data will be taken from the CORP\_CUSTOMERS (SALTA) table.
* Requested customer Verification will be done using the following criteria:

account status is active

customer is a house account

customer is valid status date (customer current status with date range)

# Appendix

## XML\_REQUEST

## House Account Balance

### URL:

<http://cpw>70252/salta/process/houseAcctBalanceService

### REQUEST:

<eps:esbSaltaServiceRequest xmlns:eps=”http://www.eps.flws.salta.in.com”>

<eps:getHouseAccountBalance>

<eps:corpAccountNumber>150474523</eps:corpAccountNumber>

<eps:sourceSystem>WEB</eps:sourceSystem>

<eps:orderNumber>5812551626</eps:orderNumber>

</eps:getHouseAccountBalance>

</eps:esbSaltaServiceRequest>

### RESPONSE:

<?xml version=”1.0” encoding=”UTF-8” standalone=”yes”?>

<esbSaltaServiceResponse xmlns=”http://www.eps.flws.salta.out.com”>

<getHouseAccountBalanceResponse>

<balance>500.0</balance>

</getHouseAccountBalanceResponse>

</esbSaltaServiceResponse>

## Payment Authorization

### URL:

<http://cpw>70252/salta/process/paymentAuthorizeService

### REQUEST:

<eps:esbSaltaServiceRequest xmlns:eps=”http://www.eps.flws.salta.in.com”>

<eps:authorizeHouseAccount>

<eps:corpAccountNumber>150474523</eps:corpAccountNumber>

<eps:sourceSystem>WEB</eps:sourceSystem>

<eps:orderNumber>5812551626</eps:orderNumber>

<eps:amount>499.0</eps:amount>

</eps:authorizeHouseAccount>

</eps:esbSaltaServiceRequest>

### RESPONSE:

<?xml version=”1.0” encoding=”UTF-8” standalone=”yes”?>

<esbSaltaServiceResponse xmlns=”http://www.eps.flws.salta.out.com”>

<authorizeHouseAccountResponse>

<approvalCode>200</approvalCode>

<approvalStatus>APPROVED</approvalStatus>

</authorizeHouseAccountResponse>

</esbSaltaServiceResponse>

## Payment Authorization Audit Table

## DDL

Audit table creation script:

CREATE TABLE PYMTAPPROVALAUDIT (

ID NUMBER(10) NOT NULL,

ACCOUNT\_NUMBER NUMBER(10) NOT NULL,

APPROVAL\_DATE DATE DEFAULT (sysdate) NOT NULL,

REQUEST\_AMOUNT NUMBER(8,2) DEFAULT 0.0,

AVAILABLE\_BALANCE NUMBER(8,2) DEFAULT 0.0,

ORDERID NUMBER(10) NOT NULL,

SOURCESYSTEM VARCHAR2(30) NOT NULL,

APPROVALCODE VARCHAR2(5) NOT NULL,

APPROVALSTATUS VARCHAR2(10) NOT NULL,

COMMENTS VARCHAR2(1000),

CONSTRAINT PYMTAPPROVALAUDIT\_PK PRIMARY KEY (ID)

) SEGMENT CREATION IMMEDIATE

PCTFREE 10 PCTUSED 40 INITRANS 1 MAXTRANS 255

NOCOMPRESS LOGGING

STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645

PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1

BUFFER\_POOL DEFAULT FLASH\_CACHE DEFAULT CELL\_FLASH\_CACHE DEFAULT)

TABLESPACE “REFRESH\_DATA” ;

COMMENT ON TABLE PYMTAPPROVALAUDIT IS ‘ House Account Audit Payment Approval’;

# Change Log

|  |  |  |
| --- | --- | --- |
| Changed By | Date | Comments |
| Casmon Gordon | 04-19-2017 | Initial Revision |
| Casmon Gordon | 04-21-2017 | Modification made for feedback comments from Hariharan. Payment authorization accepts amount field. |
| Casmon Gordon | 05-16-2017 | Changes made for atomic process transaction, update available credit balance, wrong code checked in that reference MBP database. |