
PAYMENT

pay_deposit_direct

- 1-GET DESCRIPTION
 - 2-GET ACCOUNTS
 - 3-RECORD DEPOSIT BETWEEN MASTER/SLAVE/CURRENCY
 - 4-GET MAX_FEES_APPLIES
 - 5-RECORD ACTION
 - 6-TRANSACTIONS AND RECORD PAYMENT [SSI/CC/INTERAC] (PAYMENT_TYPE SPECIFIC)
 - 7-GENERAL TRANSACTIONS
 - 8-RECORD DEPOSIT_PAYMENT
 - 9-UPDATE RECORDED DEPOSIT REAL AMOUNT [BASED ON PAYMENT_TYPE]
 - 1-RETURN DEPOSIT/PAYMENT ID COMBO
-

pay_deposit_direct_player

- 1-GET DESCRIPTION
 - 2-GET ACCOUNTS
 - 3-GET MAX_FEES_APPLIES
 - 4-RECORD DEPOSIT BETWEEN MASTER/SLAVE/CURRENCY
 - 5-RECORD ACTION
 - 6-RECORD PAYMENTS
 - 7-TRANSACCIONS
 - 8-RECORD DEPOSIT
 - 9-UPDATE RECORDED DEPOSIT REAL AMOUNT [BASED ON PAYMENT_TYPE]
 - 10-RETURN DEPOSIT/PAYMENT COMBO
-

pay_invoice_direct

- 1-CHECK INVOICE IS IN READY STATUS
 - 2-GET DESCRIPTION
 - 3-GET INVOICE INFO
 - 4-IF PAYMENT AMOUNT IS GREATER THAT OWING AMOUT
 - 5-GET ACCOUNTS
 - 6-GET MAX_FEES_APPLIES
 - 7-RECORD PAYMENT
 - 8-RECORD INVOICE_PAYMENT
 - 9-RECORD ACTION
 - 10-TRANSACTIONS GENERAL
 - 11-TRANSACTIONS SPECIFIC BASEDON PAYMENT_TYPE
 - 12-UPDATE INVOICE AMOUNTS
 - 13-UPDATE INVOICE STATUS TO PAID IF GOT FULLY-PAID
-

pay_invoice_internal

- 1-GET DESCRIPTION
- 2-EXIT IF INVOICE IS NOT IN READY STATE
- 3-CHECK WHEATHER OR NOT THIS CURRENCY IS SYSTEM OR NOT ?
- 4-GET INVOICE INFO
- 5-EXIT IF TRY TO PAY MORE THATN INVOICE_OWING

6-GET ACCOUNTS
7-CHECK IF PAYING_ENTITY HAS ENOUGH MONEY IN WALLET
8-RECORD PAYMENT
9-RECORD INVOICE_PAYMENT
10-RECORD ACTION
-NOTE : IF CURRENCY IS UNREAL MONEY THEN NO SSI_FEES INVOLVED
11-TRANSACTIONS
12-UPDATE INVOICE_AMOUNTS
13-IF INVOICE GOT FULLY-PAID THEN CHANGE STATUS TO PAID

Emails

//HELP

/*

\$getOrgsAddresses =\$this->getOrgsAddresses(\$orgIdList); //GROUPPED BY
ORGID/[ADDRESS]/ADDRESSTYPE

\$getUsersAddressesContacts =\$this->getUsersAddressesContacts(\$userIdList); //GROUPPED BY
USERID/[ADDRESS]/ADDRESSTYPE & USERID/[CONTACT]/CONTACTTYPE

\$getOrgsUsersRoles =\$this->getOrgsUsersRoles(\$orgIdList); //GROUPPED BY ORGID

\$getUsersRoles =\$this->getUsersRoles(\$userIdList); //GROUPPED BY USERID

\$getUsersInfo =\$this->getUsersInfo(\$userIdList) //GROUPPED BY USERID

*/

//HELP

/*

(1-1)Player->Team ==>Deposit Payment Receipt to Player from SSI(org)& Notification to all Team mgr
RoleUsers from Payer

(1-2)Team->league ==>Deposit Payment Receipt to Team from SSI(org)& Notification to all League mgr
RoleUsers from Payer

(2-1)Team(OT=6)->League(OT=3) payment

(2-1-1)Payment Receipt to Payer from SSI

(2-1-2)Payment Notification to all league Exec and treasurer(3,5) users from Payer

(2-2)League->Team Invoice invoice notification to team manager

(3-1)League(OT=3)->Assoc payment(OT=2)

(3-1-1)Payment Receipt to Payer from SSI(ORG)

(3-1-2)Payment Notification to all Assoc mgr and treasurer(1,5) users from Payer

(3-2)Assoc->League Invoice invoice notification to league exec & treasurer

(4-1)Assoc(OT=2)->SSI payment(OT=1)

(4-1-1)Payment Receipt to Payer from SSI(ORG)

(4-1-2)Payment Notification to all SSI-SYSTEM (14) users from Payer

(4-2)SSI->Assoc Invoice invoice notification to Assoc exec & treasurer

*/