Consimple loyalty on Creatio

Integration of cash desk software with the loyalty program processing

Contents

Integration with the cash desk software

Integration of the cash desk software with the Creatio loyalty processing is used to solve following issues:

- Preprocessing of purchases and returns
- Processing of purchases and returns
- Getting the information about cards

Interaction model assumes that the source data for the specified objects is being downloaded/updated on a regular base in the JSON format via HTTP or REST web services.

Regular update can be set up in the real time or according to the schedule with specified exchange frequency.

Purchase processing is performed only via AMQP protocol with the RabbitMQ broker.

General business process

Data exchange business process between cash desk software and processing is executed as follows:

At the checkout, the products (including certificates, coupons and loyalty cards) are scanned to the check, then the GetCardBalanceInfo method is called to get the number of bonuses that are ready for charging. The identification of the participant of the loyalty program can also be carried out by searching by the mobile phone number.

The SetPurchaseInfo method is called afterwards. All information on the scanned purchase (product, price, quantity, cost etc.) is passed to this method, and it returns information about the applied mechanics (product, price, quantity, cost and identifier of the mechanics that was applied to a specific product). If more products should be scanned or some products should be removed from the check, the products are being scanned to this check once again and the same method is being called.

Also, the cashier can choose the amount of bonuses that the buyer wants to charge for the purchase. In this case, the processing calculates the total amount to be charged.

After the purchase, the cash desk calls the CommitPurchase method.

If the CommitPurchase will not be called and start scanning new check, the processing will write the purchase to the Creatio database.

General proposals for the cash desk customization

To ensure the efficiency of the specified business process, perform the following modifications for the cashier workplace (cash desk):

1) Add the “Loyalty card number” and “Mobile phone number” fields to the screen. These fields are used to enter information, which identifies the buyer (the participant of the loyalty program). This information is obtained by calling the GetCardBalanceInfo method.

2) Add the "Number of bonuses for charging" field to the screen. The number of bonuses that are currently active for charge can be displayed in this field. This information is obtained by calling the GetCardBalanceInfo method.

3) Develop a method for calling and receiving a response from the SetPurchaseInfo method.

4) Add the possibility of obtaining information about loyalty rules from the accounting system. This information is used to detail the applied loyalty program mechanics for each product on the cash desk. Therefore, it is necessary to make an additional screen, on which the product and the description of the loyalty program mechanics that have been applied to this product by processing will be displayed.

5) Develop functions for offline storing of purchases in case of disconnection with the processing. These functions imply sending offline purchases to the processing on a schedule base after connection is restored (the IsOffLinePurchase flag of the SetPurchaseInfo method).

Preprocessing of purchases and returns

The SetPurchaseInfo web service description

Name	Description
Name	SetPurchaseInfo
Description	This service is used to preprocess purchases and returns to the Creatio

Path to the service	SetPurchaseInfo

Incoming parameters

Required parameters are highlighted.

To avoid unauthorized data transfer to the Creatio, it is necessary to transfer authentication parameters in the HTTP Basic Authentication format (RFC 2617) in the purchase processing service.

Name	Type	Description
CardNumber	String	Loyalty card number
ContactCode	Integer	Contact code. *May not be transferred.*
PointOfSaleCode	String	Outlet code
CashDeskCode	String	Cash desk code
Date	Date/time	Date and time of the purchase
Number	Integer	Number of current purchase on the cash desk (fiscal number). *May not be transferred. The number is automatically generated by the processing.*
TotalAmount	Decimal	General purchase amount. *Further it is recalculated by the processing*

CurrencyCode	String(3)	Currency code. *If not transferred, the default value (USD) will be applied*
BonusesPaidAmount	Integer (cent)	Amount of bonuses to be charged
CashierNameCode	String(50)	Unique cashier code
IsOffLinePurchase	Boolean	Flag of the offline purchase. If true, the purchase will be written to the Creatio database and discounts are not calculated. Calculation of bonuses enabled.
IsPurchaseWithoutProcessing	Boolean	Flag of purchase without processing. Passed only if **IsOffLinePurchase = true** If true, the charge of bonuses disabled and the bonuses are not credited.
TISGuid	String	Unique Id of the purchase (GUID in the accounting system)
Products	**Products** collection	The collection with the data of products in the purchase
GiftCards	**GiftCards** collection	The collection with the gift cards data
Coupons	**Coupons** collection	The collection with the coupons data:

Block of fields with information about the return:

Fields in the block are required when transferring returns.

Name	Type	Description
IsReturn	Boolean	Flag of returning products
ParentPurchaseNumber	Integer	The number of the initial purchase, products of which are returned. The number was transferred after CommitPurchaseInfo and stored in the cash desk as a flag of transferring the purchase to preprocessing.
ParentPurchaseId	String	Unique Id of the purchase (GUID in the accounting system) Optional, if the ParentPurchaseNumber passed Required if the **TISGuid** is passed in all purchases

Products collection:

Parameters with the data about products in the purchase:

Name	Type	Description
Position	Integer	Position number in the check
ReturnPosition	Integer	Position number in the return *Is optional to pass. The products will be selected according to the passed code, according to which the greatest number of bonuses is credited*

ProductCode	String	Code of the product/service (SKU) Unique product Id
Quantity	Decimal	Quantity
Price	Decimal	Price of the item
BatchNumber	String	Product series number
MinPrice	Decimal	The minimum allowed price of the product, after calculating the product discounts.
Amount	Decimal	Total by position. *Calculated automatically if not passed.*
DontUseInPreprocessing	Boolean	A flag that there will be no discount or bonuses for this product. Pass if the product already has a discount on the cash desk (sticker)

GiftCards collection:

Parameters with the data about gift cards:

Name	Type	Description
CardNumber	String	Gift card number

Coupons collection:

Parameters with the data about coupons:

Name	Type	Description

CardNumber	String	Coupon number

Outgoing parameters

Name	Type	Description
PreprocessingId	Guid	Unique preprocessing Id
Info	String	Query processing result. OK in case of success or Error in case of error.
PurchaseNumber	Integer (long)	Purchase number generated in preprocessing
CashAmount	Decimal	Payment amount (money)
UsedGiftCards	String collection	Collection of the gift card numbers that were used
TotalBonusesPaidAmount	Decimal	Amount paid with bonuses
TotalGiftCardPaidAmount	Decimal	Amount paid with gift cards
TotalDiscount	Decimal	Discount total
TotalRoundSumDelta	Decimal	The total amount of the rounding delta. Rounded to 10 cents down.

		Example: **CashAmount** = 16.73. **3** will be passed in this parameter
Products	The ResultProductInfo collection	Product without any promotion
ComplexCampaignProductsCondition		Products with promotions in complex mechanics (Products Conditions)
ComplexCampaignProductsUsage		Products with promotions in complex mechanics (Products Promotions) and discounts
GiftsOnCashDesk	The GiftOnCashDesk collection	Gifts on the cash desk
ResponseStatus	ResponseStatus	Status
Transactions	Collection ResultTransaction I nfo	Transactions of bonus crediting

CampaingsWithoutUsageProducts	Collection ***CampaingWithoutUsageProducts***	A collection of not triggered promotions. Used to display prompts at the cash desk.
CouponsCondition	Collection	Collection of generated coupons
TotalChargeBonusAmount	Decimal	Amount of credited bonuses for purchase
TotalCouponsPaidAmount	Decimal	General amount is paid with coupons

Products collection:

Parameters with data about products in the purchase, which were not affected by the promotions

Name	Type	Description
Position	Integer	Position
ProductCode	String	Product code. Unique product Id
BatchNumber	String	Product series number
RoundSumDelta	Decimal	Rounding delta according to the number of products **TotalRoundSumDelta** is divided into the number of products in the purchase, depending on the price of the product

Price	Decimal	Initial price for one product
Quantity	Decimal	Quantity
CashPaidAmount	Decimal	Amount of payment with money according to the number of products
BonusesPaidAmount	Decimal	Amount of payment with bonuses according to the number of products
GiftCardsPaidAmount	Decimal	Amount of payment with gift cards according to the number of products
CampaignId	GUID	Id of the promotion trigged
CampaignTitle	string	Name of the mechanics (loyalty program rules)
DiscountAmount	Decimal	Discount amount
PercentageDiscount	Decimal	Discount rate
ChargeBonusAmount	Decimal	Amount of credited bonuses for product
CouponsPaidAmount	Decimal	Amount paid with coupons

ComplexCampaignProductsCondition collection:

Parameters with data about products in the purchase that participated in the mechanics as condition products:

Name	Type	Description
Position	Integer	Position number in the check
ProductCode	Integer	Code of the product/service (SKU) Unique product Id
BatchNumber	String	Product series number
RoundSumDelta	Decimal	Rounding delta according to the number of products **TotalRoundSumDelta** is divided into the number of products in the purchase, depending on the price of the product
Price	Decimal	Price of the item
Quantity	Decimal	Quantity
CashPaidAmount	Decimal	Payment amount (money)
BonusesPaidAmount	Decimal	Paid with bonuses
GiftCardsPaidAmount	Decimal	Paid with gift cards
CampaignId	GUID	Machanic that completed
CampaignTitle	String	Mechanic name (loyalty program rules)

ComplexCampaignProductsUsage collection:

Parameters with the data about products in the purchase that participated as promotion products:

Name	Type	Description
Position	Integer	Position number in the check
ProductCode	Integer	Code of the product/service (SKU) Unique product Id
BatchNumber	String	Product series number
RoundSumDelta	Decimal	Rounding delta according to the number of products **TotalRoundSumDelta** is divided into the number of products in the purchase, depending on the price of the product
Price	Decimal	Price of the item
Quantity	Decimal	Quantity
CashPaidAmount	Decimal	Payment amount (money)
BonusesPaidAmount	Decimal	Paid with bonuses
GiftCardsPaidAmount	Decimal	Paid with gift cards
CampaingId	GUID	Completed mechanics
CampaignTitle	String	Name of the completed mechanics
DiscountAmount	Decimal	Discaunt amount

PercentageDiscount	Decimal	Discount rate

GiftOnCashDesk collection:

Parameters with the data about gifts on the cash desk:

Name	Type	Description
CampaignId	Guid	Id of the promotion that trigged
CampaignTitle	String	Mechanic name (loyalty program rules)

UsedGiftCards collection:

Parameters with the data about used gift cards (certificates):

Name	Type	Description
CardNumber	String	Numbers of the used gift cards

CouponsCondition collection:

Parameters with the data about generated coupons (according to the rules of the loyalty program coupon (generation)):

Name	Type	Description
Title	String	Mechanics name (loyalty program rules with the Coupon generation type) by which the coupon was generated at the purchase

Id	GUID	CampaignId (loyalty program rules)
Code	String	CardNumber – number of generated gift card (stores the number, bar code)
CouponDescription	String	Mechanics description (loyalty program rules with the Coupon generation type) by which the coupon was generated according to the purchase

CampaingsWithoutUsageProducts collection:

Used to display prompts at the cash desk.

Consists of the condition products and the id of the promotions, which can apply to the condition products (there is no bonus products in the purchase):

Name	Type	Description
CampaignId	GUID	Mechanics id, which can work when adding a product to the promotion
CampaignTitle	String	Mechanics name (loyalty program rules)
ConditionProductsInfo	Collection	Condition products that are in the purchase and to which a specific promotion can be applied

"ConditionProductsInfo" collection:

Name	Type	Description
ProductCode	String	Condition product that is in the purchase and to which a specific promotion can be applied

Quantity	Decimal	The number of condition products in the purchase for this product

Transactions collection:

Parameters with the data about crediting transactions:

Name	Type	Description
Amount	Decimal	Amount of credited bonuses for applied discounts
CampaignId	Guid	GUID of the trigged promotion
BonusType	String	Bonus type
CampaignTitle	String	Header of the trigged promotion
ProductCode	String	Product code for which the promotion is trigged

ResponseStatus block:

Information block about errors occurred during the query processing:

Name	Type	Description
ErrorCode	String	Error code
Message	String	Error message

ResponseStatus.Errors collection:

Detail information about errors occurred:

Name	Type	Description
ErrorCode	String	Error code
FieldName	String	Field in which the error occurred
Message	String	Error message text

```
{
  "CardNumber": "1440",
  "PointOfSaleCode": "189",
  "Date": "2016-06-24T13:54:48.447Z",
  "Products": [
    {
      "Position": 1,
      "ProductCode": "70365686",
      "Quantity": "1",
      "Price": 1500
    },
    {
      "Position": 2,
      "ProductCode": "911003",
      "Quantity": "1",
      "Price": 3500
    }
  ],
  "BonusesPaidAmount": 0
}
```

Processing of purchases and returns

The CommitPurchaseInfo web service description

Name	Description
Name ENU	CommitPurchaseInfo
Notes	This service is used to process purchases and returns to the Creatio
Path to the service	CommitPurchaseInfo

Incoming parameters

Required parameters are highlighted.

To avoid unauthorized data transfer in Creatio, it is necessary to transfer authentication parameters in the HTTP Basic Authentication format (RFC 2617) in the purchase processing service.

Name	Type	Description
PreprocessingId	Guid	Id of preprocessing purchase

IsOddToCard	Boolean	A flag of rounding off the purchase total to 1 USD. *Example: the customer must pay 17.80 USD. The customer gives 20 USD, the 0.20 USD is credited as bonuses and the final payment will be 18 USD.* *If the purchase total is round and this parameter is set to true, the total will not be rounded.*
OddToCardAmount	Decimal	Change amount transfered to credit card. If the **OddToCardAmount** = 0, and the IsOddToCard is true, the rest will be rounded to 1 USD. If the **OddToCardAmount > 0**, the bonuses will be credited in amount passed in the OddToCardAmount parameter. The round to 1 USD will not be applied regardless the value passed in the IsOddToCard

Outgoing parameters

Name	Type	Description
Info	String	Query processing result. OK in case of success or Error in case of error.
PurchaseNumber	Integer	Creatio purchase number (same as purchase number returned at preprocessing)
Transactions	Collection ResultTransactionInfo	Transactions of bonus crediting. Passed if incoming parameters have **IsOddToCard = true**

ResponseStatus block:

Information block about errors occurred during the query processing:

Name	Type	Description
ErrorCode	String	Error code
Message	String	Error message

ResponseStatus.Errors collection:

Detail information about errors occurred:

Name	Type	Description
ErrorCode	String	Error code
FieldName	String	Field in which the error occurred
Message	String	Error message text

Transactions collection:

Parameters with the data about crediting transactions:

Name	Type	Description
Amount	Decimal	Amount of credited bonuses for applied discounts
CampaignId	Guid	GUID of the trigged promotion Empty if transactions are for rounding purchase amount
BonusType	String	Bonus type

CampaignTitle	String	Header of the trigged promotion Null if transactions are for rounding purchase amount

Getting the information about cards

The GetCardBalanceInfo web service description

Name	Description
Name ENU	GetCardBalanceInfo
Description	This service is used to obtain information about balances on the card account from Creatio.
Path to the service	GetCardBalanceInfo

Incoming parameters

Required parameters are highlighted.

To avoid unauthorized data transfer in Creatio, it is necessary to transfer authentication parameters in the HTTP Basic Authentication format (RFC 2617) in the purchase processing service.

Name	Type	Description
CardNumber	String	Loyalty card number. *
OnDate	Date/time	The date on which to display the balance
CardAccountNumber	String	Account number to which the card is bind

PhoneNumber	String	Number of the mobile phone for identification in the +xxxxxxxxxxxx form. *

*- required field: the CardNumber or the PhoneNumber

Outgoing parameters

Card status:

Name	Type	Description
CardStatus	String	Card status (Locked, Registered, Not Activated, Activated, etc.)
CardType	String	Type of the scanned card. (Gift card, coupon, bonus/discount card)
ContactName	String	Contact full name

BonusesInfo collection:

Parameters with the data about bonus balance:

Name	Type	Description
BonusTypeCode	String	Bonus type code (money, status, etc.)
BonusStatusCode	String	Bonus status code (active, inactive, burned, etc.)
BonusAmount	Decimal	Amount of credited bonuses

CouponsInfo collection:

Parameters with the data about coupons:

Name	Type	Description
CouponTypeCode	String	Coupon type code (unique, reqular, etc.)
CouponStatusCode	String	Coupon status code (active, inactive, burned, etc.)

CertificatesInfo collection:

Parameters with the data about bonus balance:

Name	Type	Description
CertificateStatusCode	String	Certificate status code (active, inactive, burned, etc.)
CertificateAmount	Decimal	Denomination of certificate (gift card)

Example of a service request and response

1.1. Card balance request: {

```
     "CardNumber": "1440",
     "OnDate": "2016-06-24T11:28:03.815Z"
  }
```

1.2. Responce in case of success:

```
  {
     "CardStatus":"Registered",
     "BonusesInfo":[
       {
         "BonusTypeCode":"Money",
         "BonusStatusCode":"Burnt",
         "BonusAmount":26.40
       },
       {
         "BonusTypeCode":"Money",
         "BonusStatusCode":"Active",
         "BonusAmount":768.00
       }
     ]
   }
```