

Single Dynamic PullPayment Contract

Introduction

The Single Dynamic PullPayment contract simplifies the one time payment for the businesses. Using the Single Dynamic PullPayment contract, any merchant can create the billing model with required configuration for their business to simplify the payment. Customers can subscribe to the billing model to make the payments for the service.

In this case, the payment properties (currency, price, name) can be injected straight onto merchant websites and not through the Business Console. The rest of the details of the billing model are provided dynamically through the code-snippet when the customer subscribes to the billing model.

This type of billing model is most suited to merchants who sell tens / hundreds / thousands of products with different descriptions and prices.

Contract version

pragma solidity 0.8.0

Contract constructor

The Single Dynamic Pullpayment contract's constructor takes one argument i.e the main registry contract address which is used to access the registry methods.

Constructor Signature:

- `constructor(address registryAddress) RegistryHelper(registryAddress);`

Constructor Parameters:

1. **registryAddress:**
 - indicates the address of the registry contract which keeps the record of all the contract addresses in the ecosystem and the required configurations.

Contract Methods

createBillingModel()

Method name:

- createBillingModel

Method detail:

- This method allows merchants to create a new billing model for their business with required configurations.
- The unique Id is created for each billing model.
- Merchant can create as many billing models as he wants.
- Creator specifies the payee address, merchant name, unique reference and the merchant url for the billing model. Rest of the details(ex. Amount, settlement token, name etc.) are provided by the customer at the time of subscription through the code-snippet.
- This emits the **BillingModelCreated** event after creating a billing model.

Method signature:

- ```
function createBillingModel(
 address _payee,
 string memory _merchantName,
 string memory _reference,
 string memory _merchantURL
) public virtual override returns (uint256 billingModelID)
```

Method parameters:

1. **\_payee:**
  - Indicates the address of the receiver(merchant) who will receive the pull payments.
2. **\_merchantName:**
  - Indicates the optional merchant name
3. **\_reference:**
  - Indicates the unique reference for the billing model. It can be kept empty.
  - If an external unique reference is not given then the contract creates the unique reference for the billing model.
4. **\_merchantURL:**
  - Indicates the optional merchant's personal URL.

Returned data:

- This method returns the unique billing model Id.

# subscribeToBillingModel()

Method name:

- subscribeToBillingmodel

Method detail:

- This method allows any customer to subscribe to the billing model with a given billing model Id, name, settlement token address, payment token address, payment amount and the unique reference for the subscription.
- Here the billing model details i.e billing model name, settlement token and payment amount are provided dynamically through the code snippet.
- The user must approve the payment tokens to the Executor contract before subscribing to the billing model.
- The unique subscription id and pullPayment id is created on each subscription.
- When a customer subscribes to the billing model with a given payment token, Immediately the direct pullPayment is executed with the help of the Executor contract.
- This method emits the **NewSubscription** event.

Method signature:

- ```
function subscribeToBillingModel(
    uint256 _billingModelID,
    bytes32 _name,
    address _settlementToken,
    address _paymentToken,
    uint256 _paymentAmount,
    string memory _reference
) public override returns (uint256 subscriptionID)
```

Method parameters:

- **_billingModelID:**
 - Indicates the unique billing model id that the customer wants to subscribe.
- **_name:**
 - Indicates the optional name for the billing model.
- **_settlementToken:**
 - Indicate the settlement token address in which the payee wants to receive the payment.
- **_paymentToken:**
 - Indicates the payment token address in which the customer wants to pay for the subscription.
- **_paymentAmount:**
 - Indicates the amount to pay for the subscription.
- **_reference:**
 - Indicates the unique reference for the subscription.
 - If not provided, the contract creates a unique reference for the subscription.

Returned data:

- This method returns the unique subscription id after subscription.

editBillingModel()

Method name:

- editBillingModel

Method detail:

- This method allows the billing model creator i.e payee to edit the details of the billing model details.
- It allows the payee to update the payee address, merchant name and the merchant url of the billing model.
- Only the owner of the billing model can edit the billing model details.
- This method emits the **BillingModelEdited** event after updating the billing model.

Method signature:

- ```
function editBillingModel(
 uint256 _billingModelID,
 address _newPayee,
 string memory _newMerchantName,
 string memory _newMerchantURL
)
 public virtual override returns (uint256 billingModelID)
```

Method parameters:

- **\_billingModelID:**
  - Indicates the unique valid billing model id which the merchant wants to edit.
- **\_newPayee:**
  - Indicates the new payee address for the billing model.
  - Merchants cannot use Zero address as payee address.
- **\_newMerchantName:**
  - Indicates the new merchant name for the billing model. It can be kept empty.
- **\_newMerchantURL:**
  - Indicates the updated merchant's personal URL. It can be kept empty as well.

Returned data:

- This method returns the id of the billing model whose details are updated.

# getBillingModel()

Method name:

- `getBillingModel`

Method detail:

- This method retrieves all the details of the specified billing model.
- Only the payee of a particular billing model can retrieve the subscription ids of the billing model.

Method signature:

- `function getBillingModel(uint256 _billingModelID)`  
external  
view  
override  
returns (BillingModelData memory bm)

Method parameters:

- **`_billingModelID:`**
  - Indicates the valid billing model id.

Returned data:

- This method returns the details of the billing model which includes the-
  - Payee address
  - Merchant name
  - The unique reference of billing model
  - Merchant URL
  - Subscription Ids (only payee can retrieve)
  - Billing model creation time

# getSubscription()

Method name:

- getSubscription

Method detail:

- This method returns the subscription data for a given subscription id.
- Only the payee and subscriber of the billing model can see the pullPayment ids of subscription.

Method Signature:

- function getSubscription(uint256 \_subscriptionID)  
external  
view  
override  
returns (SubscriptionData memory sb)

Method parameters:

- **\_subscriptionID:**
  - Indicates the valid subscription id

Returned data:

- This method returns the subscription data which includes the following data-
  - Payee address of the billing model
  - Subscriber address
  - Name of billing model
  - Settlement token address
  - Payment token address
  - PullPayment IDs(only payee or subscriber gets the ids)
  - The unique reference of subscription.

## getPullPayment()

Method name:

- getPullPayment

Method detail:

- This method retrieves the pullPayment details for given pullPayment id.
- It gives the payment amount, the pullPayment timestamp, billing model id and the subscription id of the pullpayment.
- Only the granted executor, payee and the subscriber of pullPayment can see the payment amount and execution timestamp.

Method signature:

- function getPullPayment(uint256 \_pullPaymentID)  
external  
view  
returns (PullPayment memory pullPayment)

Method parameters:

- **\_pullPaymentID:**
  - Indicates the valid pullPayment id.

Returned data:

- This method returns the pullPayment data for given pullPayment id which includes-
  - Payment amount
  - Timestamp when pullPayment executed.
  - The billing model id to which pullpayment belongs
  - The subscription id to which pullpayment belongs.

## getBillingModelIdsByAddress()

Method name:

- `getBillingModelIdsByAddress`

Method detail:

- This method retrieves the billing model ids for a given creator address.

Method signature:

- `function getBillingModelIdsByAddress(address _creator)`  
external  
view  
returns (uint256[] memory billingModelIDs)

Method parameters:

- **\_creator:**
  - Indicates the billing model creator address whose billing model id is to retrieve.

Returned data:

- This method returns the lists of billing model ids that a creator has created.

## getSubscriptionIdsByAddress()

Method name:

- `getSubscriptionIdsByAddress`

Method detail:

- This method retrieves the subscription ids for a given subscriber address.

Method signature:

- `function getSubscriptionIdsByAddress(address _subscriber)`  
external  
view  
returns (uint256[] memory subscriptionIDs)

Method parameters:

- **\_subscriber:**
  - Indicates the subscriber address whose subscription id is to retrieve.

Returned data:

- This method returns the lists of subscription ids that a subscriber has subscribed.



## getPullPaymentsIdsByAddress()

Method name:

- `getPullPaymentsIdsByAddress`

Method detail:

- This method retrieves the list of pullPayment ids for a given subscriber.

Method signature:

- `function getPullPaymentsIdsByAddress(address _subscriber)`  
external  
view  
returns (uint256[] memory pullPaymentIDs)

Method parameters:

- **\_subscriber:**
  - Indicates the subscriber address whose pullPayment ids to be retrieved.

Returned data:

- This method returns a list of pullPayment ids for a given subscriber.

## getCurrentBillingModelId()

Method name:

- `getCurrentBillingModelId`

Method detail:

- This method returns the current billing model id of the pullpayment

Method signature:

- `function getCurrentBillingModelId()` external view virtual returns (uint256);

Returned data:

- This method returns the current billing model id.

## getCurrentSubscriptionId()

Method name:

- `getCurrentSubscriptionId`

Method detail:

- This method returns the current subscription id of the pullpayment

Method signature:

- `function getCurrentSubscriptionId() external view virtual returns (uint256);`

Returned data:

- This method returns the current subscription id.

## getCurrentPullPaymentId()

Method name:

- `getCurrentPullPaymentId`

Method detail:

- This method returns the current id of the pullpayment

Method signature:

- `function getCurrentPullPaymentId() external view virtual returns (uint256);`

Returned data:

- This method returns the current pullpayment id.

# Events

## BillingModelCreated

Event Name:

- BillingModelCreated

Event detail:

- This event is emitted when the creator creates a new billing model using **createBillingModel()** method.

Event signature:

```
event BillingModelCreated(
 uint256 indexed billingModelID,
 address indexed payee
);
```

Event data:

- **billingModelID:**
  - Indicates the newly created unique billing model id.
- **Payee:**
  - Indicates the payee address which will receive the payment for subscription.

# NewSubscription

Event name:

- NewSubscription

Event detail:

- This event is emitted when a subscriber subscribes to the billing model using **subscribeBillingModel()** method.

Event signature:

- event NewSubscription(  
    uint256 indexed billingModelID,  
    uint256 indexed subscriptionID,  
    uint256 indexed pullPaymentID,  
    address payee,  
    address payer,  
    uint256 executionFee,  
    uint256 userAmount,  
    uint256 receiverAmount  
);

Event data:

- **billingModelID:**
  - Indicates the valid billing model id that subscriber has subscribed to.
- **subscriptionID:**
  - Indicates the subscription id when a subscriber subscribes to a given billing model.
- **pullPaymentID:**
  - Indicates the pullpayment id for the subscription.
- **payee:**
  - Indicates the payee address of the billing model.
- **payer:**
  - Indicates the subscriber address.
- **executionFee:**
  - Indicates the execution fee charged for executing the pullpayment
- **userAmount:**
  - Indicates the amount of tokens that the user paid for subscription.
- **receiverAmount:**
  - Indicates the amount of tokens that payee receives from the pullpayment.

# BillingModelEdited

Event name:

- BillingModelEdited

Event detail:

- This event is emitted when the billing model is edited using the **editBillingModel()** method.

Event signature:

- event BillingModelEdited(  
uint256 indexed billingModelID,  
address indexed newPayee,  
address indexed oldPayee,  
string newMerchantName,  
string newMerchantUrl  
);

Event data:

- **billingModelID:**
  - Indicates the edited billing model id.
- **newPayee:**
  - Indicates the updated payee address.
- **oldPayee:**
  - Indicates the old payee address
- **newMerchantName:**
  - Indicates the updated merchant's name
- **newMerchantUrl:**
  - Indicates the updated merchant's personal URL.

## Flow of Execution:

1. First merchant creates the billing model with required configuration for his business. Only the basic information (payee address, merchant name, reference and merchant url) of the billing model is stored on the blockchain when the merchant creates the billing model.
2. Subscriber should approve the unlimited payment tokens to the Executor contract before subscribing to the billing model.
3. Subscriber subscribes the required billing model with payment token that he wants to pay in to avail the services offered by the merchant. Also, the remaining details of the billing model (i.e amount, settlement token and the bm name) are provided dynamically from the code-snippet at the time of subscription.
4. When a Subscriber subscribes to the billing model, immediately a single pullPayment is executed.
5. Executor is the contract which gets the tokens from the subscriber, swaps the payment token to the settlement token, transfers the execution fee to the fee receiver and then transfers the settlement tokens to the payee.
6. Whenever you want to avail the service you subscribe to the billing model and make the single pullPayment.
7. Merchants can edit the billing model details i.e payee address, merchant name and merchant url.