

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 contract, only the payee address of the billing model is stored on the blockchain when any merchant creates the billing model. The rest of the details of the billing model are provided dynamically through the code-snippet when the customer subscribes to the billing model.

Contract version

pragma solidity 0.8.0

Contract constructor

In this contract the initialize() method is used in place of the constructor to allow the contract to be upgradable via proxy.

Method Name:

- initialize()

Method Usage:

- The initialize method is used to initialize the Recurring PullPayment contract and set the registry address.

Method Signature:

- function initialize(address registryAddress) external;

Method Parameters:

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

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 only specifies the payee address for the billing model. Rest of the details are provided to the customer through the code-snippet.

Method signature:

- function createBillingModel(
 address _payee,
) public override returns (uint256 billingModelID)

Method parameters:

1. _payee:

Indicates the address of the receiver(merchant) who will receive the pull payments.

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 given billing model Id, name, settlement token address, payment token address and the payment token.
- Here the billing model details i.e billing model name, settlement token, payment amount and the payment token are provided dynamically through the code snippet.
- The user must approve the payment tokens to the Executor contract before subscribing 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.

Method signature:

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

Method parameters:

- **\_billingModelID:**
  - Indicates the unique billing model id that the customer wants to subscribe.
- **\_name:**
  - Indicates the name of 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 biling model amount.

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.
- It allows the payee to update the payee address of the billing model.
- Only the owner of the billing model can edit the billing model details.

Method signature:

- ```
function editBillingModel(  
    uint256 _billingModelID,  
    address _newPayee  
) public override returns (uint256 billingModelID)
```

Method parameters:

- **_billingModelID:**
 - Indicates the unique valid billing model id which merchant wants to edit.
- **_newPayee:**
 - Indicates the new payee address for the billing model.
 - Merchant cannot use Zero address as payee address.

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
 - Subscription Ids (only payee can retrieve)

getSubscription()

Method name:

- getSubscription

Method detail:

- This method returns the subscription data for 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-
 - Subscriber address
 - Name of billing model
 - Settlement token address
 - Payment token address
 - PullPayment IDs(only payee or subscriber gets the ids)

getPullPayment()

Method name:

- getPullPayment

Method detail:

- This method retrieves the pullPayment details for given pullPayment id.
- It gives the payment amount and the pullPayment timestamp.
- Only the payee and the subscriber of pullPayment can see the payment amount and execution timestamp.

Method signature:

- function getPullPayment(uint256 _pullPaymentID)
external
view
returns (PullPaymentData 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.

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.

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 billing model id.
- **Payee:**
 - Indicates the payee address who created the billing model.

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,  
    address indexed subscriber  
);
```

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.
- **subscriber:**
 - Indicates the subscriber address who subscribed to the billing model.

BillingModelEdited

Event name:

- BillingModelEdited

Event detail:

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

Event signature:

```
event BillingModelEdited(  
    uint256 indexed billingModelID,  
    address indexed newPayee  
);
```

Event data:

- **billingModelID:**
 - Indicates the edited billing model id.
- **newPayee:**
 - Indicates the updated payee address.

Flow of Execution:

1. First merchant creates the billing model with required configuration for his business. Only the payee address 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 billing model.
3. Executor is the contract which gets the tokens from the subscriber, swaps the payment token to the settlement token and then transfers the settlement tokens to the payee.
4. 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 are provided dynamically from the code-snippet at the time of subscription.
5. When a Subscriber subscribes to the billing model, immediately a single pullPayment is executed.
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 only.