**US\_301717\_Cognizant: Enhance Get Transaction Service from RentTrack**

**Understanding of the requirement**

Enhance the Create Payment Account service and the Get Transaction service to retrieve the lease details from RentTrack in order to provide the request parameters for the service.

**Assumptions**:

* Sandbox account and credentials for plaid connection are provided and we are able to access
* Plaid public token will be provided by the invoker
* Any changes made in RentTrack API which is given in the API documentation, will be notified.
* GitLab Apiary access will be provided

**High level Technical design and solution**

* 1. Impacted Modules

bcm-service

bcm-web

bcm-common

renttrack-plugin

* 1. Impacted Workflows
     1. Overview:

The BCM Create Payment Account service includes the BCM Create Transaction Finder and search transactions.

* + 1. APIs Involved

GET /api/tenant/subscriptions/

BCM is the consumer

Error Codes:

201 - Returned when successful

400 - Error validating data. Please check parameters and retry.

409 - Can not create new one subscription

500 - Internal Server Error

POST /api/tenant/plaid/payment\_accounts

* + - * + BCM is the consumer
        + Payload : public\_token, plaid\_account\_id, pay\_type
        + Error Codes:

201 - Returned when successful

400 - Error validating data. Please check parameters and retry.

409 - User exists.

500 - Internal Server Error

POST /api/tenant/leases/{leaseId}/transaction\_finders.{\_format}

BCM is the consumer

Payload : amount\_min, amount\_max, window\_open, window\_close, payment\_account\_url, lease\_url

Error Codes:

201 - Returned when successful

400 - Error validating data. Please check parameters and retry.

404 - Payment Account or Consumer Lease not found

500 - Internal Server Error

GET api/tenant/leases/{leaseId}/transaction\_finders/{id}/search.{\_format}

* + - * + BCM is the consumer
        + Payload : leaseId, transaction finder Id, amount\_min, amount\_max, window\_open and window\_close
        + Error Codes:

200 - Returned when successful

204 - Success with no content

400 - Error validating data. Please check parameters and retry.

404 - Transaction Finder or Consumer Lease not found

500 - Internal Server Error

iii. Messaging Infrastructure: N/A

iv. UI Screens: N/A

* 1. DB Changes:

1. Auditing Tables such as **WebService\_logs, External\_log, Customer\_Usage\_activity**

2. Table Name : **customer\_data**

transaction\_finder Id and payment\_account\_id columns has been added in customer\_data table

Table Description :

* ucid
* brand
* customer\_data\_id
* customer\_email
* created\_date
* modified\_date
* access\_token
* refresh\_token
* token\_expiry
* token\_created
* transaction\_finder\_id
* payment\_account\_id
  1. Dependencies

RentTrack API

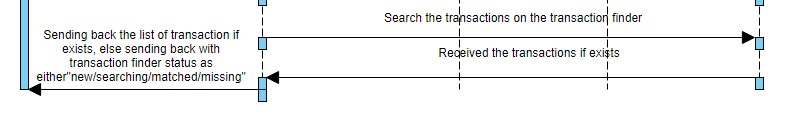
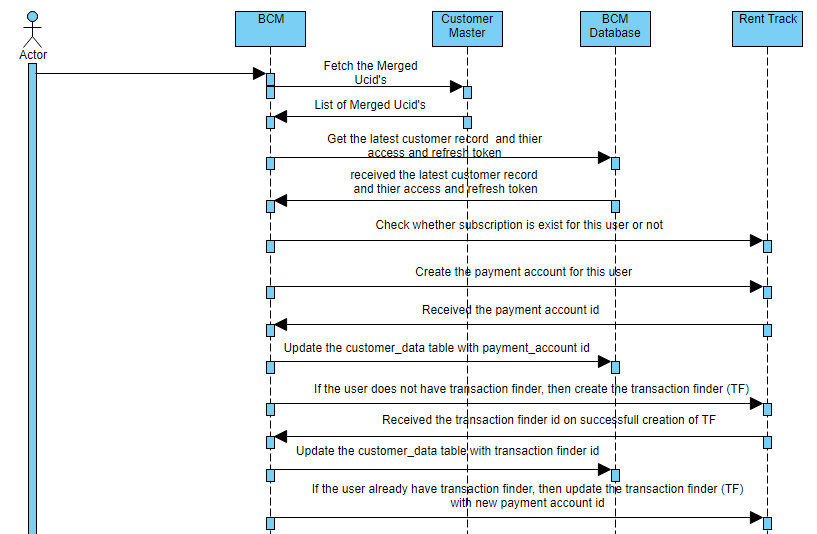
* 1. Blocker Inputs needed

The sandbox access need to be provided

* 1. Design Patterns considered and the chosen one

SOA

g) Sequence Diagram



**Business** :

**Step 1:** BCM will receive the public token and plaid account id from the RSP(UI).

Request Payload:

{

"public\_token":

"public-sandbox-b0e2c4ee-a763-4df5-bfe9-46a46bce993d”

“plaid\_account\_id”:”i544dsd561g13”

}

**Step 2:** BCM will validate the active subscription for the user by using the below RentTrack API call. If the subscription is active, then BCM Module proceed with the create payment Account Service. Otherwise, BCM will respond back to the invoker with the message “User is not subscribed”.

GET /api/tenant/subscriptions/

| **Parameter** | **Type** | **Versions** | **Description** |
| --- | --- | --- | --- |
| 200 - Returned when successful |  | | |
| [] | array of objects (UserSubscription) | \* |  |
| [][id] | integer | \* |  |
| [][url] | string | \* |  |
| [][payment\_account\_url] | string | \* |  |
| [][plan\_name] | string | \* | {'score\_track'|'credit\_track'|...} |
| [][plan\_features] | array | \* | ['reporting, ‘score’, ...] |
| [][period] | string | \* | {'monthly'|'yearly'} |
| [][status] | string | \* | {'active'|'cancelled'|'delinquent'} |
| [][last\_payment\_date] | DateTime | \* | Format "YYYY-MM-DD" |
| [][next\_payment\_date] | DateTime | \* | Format "YYYY-MM-DD" |
| [][next\_payment\_amount] | float | \* |  |
| [][discounted\_until] | DateTime | \* | Format "YYYY-MM-DD" |
| [][free\_until] | DateTime | \* | Format "YYYY-MM-DD" |
| [][enabled\_at] | DateTime | \* | Format "YYYY-MM-DD" |
| [][cancelled\_at] | DateTime | \* | Format "YYYY-MM-DD" |
| [][external\_property\_id] | string | \* |  |

**Step 3:**

Using the public token and plaid account id, BCM will invoke below RentTrack API to create a new payment account

**RentTrack sandbox : POST /api/tenant/plaid/payment\_accounts**

**Input** :

| Parameter | Type | Required? | Format | Description |
| --- | --- | --- | --- | --- |
| public\_token | string | true | {not blank} | Plaid public token |
| plaid\_account\_id | string | false |  | Selected Id of Plaid Payment Account (optional) |
| pay\_type | string | false | [general|subscription|billpay|nopay] |  |

**Output** :

| Parameter | Type | Versions | Description |
| --- | --- | --- | --- |
| 201 - Returned when successful |  | | |
| [] | array of objects (PaymentAccount) | \* |  |
| [][id] | integer | \* |  |
| [][url] | string | \* |  |
| [][nickname] | string | \* |  |
| [][type] | string | \* |  |
| [][last\_four] | string | \* |  |
| [][expiration] | string | \* | Format: "YYYY-mm" |
| [][institution\_id] | string | \* |  |
| [][plaid\_account] | boolean | \* |  |
| [][needs\_reconnect] | boolean | \* |  |
| [][card\_brand] | string | \* |  |

**Step 4** : Once the payment Account has been created successfully, Update the customer\_data table with the payment\_account\_id.

**Step 5:** BCM Module get the active leave for the user by using the below RentTrack API

GET /api/tenant/leases

**Step 6:** If there is no active Lease, then the BCM Module respond back to the invoker with the error message has “Lease Not Found”.

**Step 7:** if there is an active lease, then BCM will invoke the below RentTrack API call to create the transaction finder

POST /api/tenant/leases/{leaseId}/transaction\_finders.{\_format}

Request Body :

| **Parameter** | **Type** | **Required?** | **Format** | **Description** |
| --- | --- | --- | --- | --- |
| amount\_min | float | true | {not blank} |  |
| amount\_max | float | true | {not blank} |  |
| window\_open | integer | true | {not blank} | Day of the month to open transactions window 1-31 |
| window\_close | integer | true | {not blank} | Day of the month to close transactions window 1-31 |
| payment\_account\_url | custom handler result for (ResourceUrlEncodedRelation) | true |  | Resource url for Payment Account. Can be included on url. |
| lease\_url | custom handler result for (ResourceUrlEncodedRelation) | true |  | Resource url for Consumer Lease. Can be included on url. |

**Response :**

| Parameter | Type | Versions | Description |
| --- | --- | --- | --- |
| 201 - Returned when successful |  | | |
| id | integer | \* |  |
| url | string | \* |  |
| status | string | \* | Status of transaction finder: {'new'|'matched'|'searching'|'multiple'|'missing'|'closed'} |

**Step 8:**If the transaction finder has been created successfully, then customer data table will updated with the transaction finder id.

**Step 9:** BCM will invoke the below RentTrack API call to search the transactions, Otherwise respond back to the invoker with the error message.

GET /api/tenant/leases/{leaseId}/transaction\_finders/{id}/search.{\_format}

Filters

|  |
| --- |
| **Name** |
| **amount\_min**   |  |  | | --- | --- | | Datatype | float | | Description | Minimum amount of transaction. | |
| **amount\_max**   |  |  | | --- | --- | | Datatype | float | | Description | Maximum amount of transaction. | |
| **window\_open**   |  |  | | --- | --- | | Datatype | integer | | Description | Day of the month. | |
| **window\_close**   |  |  | | --- | --- | | Datatype | integer | | Description | Day of the month. | |

##### **Response:**

| **Parameter** | **Type** | **Versions** | **Description** |
| --- | --- | --- | --- |
| 200 - Returned when successful |  | | |
| [] | array of objects (PlaidTransaction) | \* |  |
| [][transaction\_id] | string | \* |  |
| [][amount] | string | \* |  |
| [][name] | string | \* |  |
| [][category\_id] | string | \* |  |
| [][date] | DateTime | \* | Format: "YYYY-MM-DD" |

**Step 10:** BCM will respond back to the invoker with the search transaction response in the below format.

{  
 "transaction\_search\_status": "Multiple",  
 "plaid\_account": false,  
 "reconnect\_to\_plaid": false,  
 "historical\_update\_complete": false,  
 "transactions": {  
 "transaction\_id": "G5K3gvJql1TlWzEmKwE7HwMv6wkMaqs15EvXB",  
 "transaction\_amount": 5.12,  
 "name": "Uber 063015 SF\*\*POOL\*\*",  
 "transaction\_date": "2019-09-23"  
 },  
 {  
 "transaction\_id": "bJP3GvZlA7FA8ZgRGzgPu4eJ64MeZViVXDj1j",  
 "transaction\_amount": 4.52,  
 "name": "McDonald's",  
 "transaction\_date": "2019-09-27"  
 },  
 {  
 "transaction\_id": "mk3LGdvNzEuajZlGE7lzC1mxR1JmDVtLDqNna",  
 "transaction\_amount": 10.52,  
 "name": "Starbucks",  
 "transaction\_date": "2019-09-02"  
 }  
 }

**Clarifications**:

BCM will save the transaction finder id and the payment account id against the BCM customer Id in customer\_data table as per the discussion.

**Code / Framework References**

**Misc:**