Bill Payment System

High Level Requirement

Document Control

Key personnel

Title	**Bank Bill Payment System** **High Level Business Requirement**
Author	Gunjan Prasad
Approver	
Owner	

Version History

Version	Date	Summary of changes	Changes marked
V 0.01	25-May-2109	Initial Draft : High level requirement of Bill payment system	N

Distribution

Name	Area

References

Doc reference	Document title

Table of Contents

1. Introduction

1.1. Objective:

The objective of this project is to enhance the Bank's personal Banking platform by creating an online bill payment system.

With the implementation of new functionality, exiting and new customers will be able to manage payee list, pay bill on time and also maintain the history of all bill payments.

In addition, this new feature on the Personal Banking platform will enhance the bank's overall Cash Management Services by providing the following benefits:

- Improved customer experience, leading to increased customer loyalty and retention
- Increase in revenue from existing and new client
- Development of new capabilities that can be further expanded

1.2. Purpose:

The purpose of this document is to provide an overview of the proposed software product, its parameters and goals. This document defines the high level product features for the business users and stakeholders, with product scope and its functionalities along with assumptions and constrains. A detailed software requirement specification with detailed software design will further be produced.

1.3. In Scope

The Minimal-Viable-Product(MVP) scope definition for the proposed new Bill Payment system is as follows:

- Real-time payments and confirmation to/from merchants(payee / billers)
- Real-time validations when customers add/modify merchants(payee / billers)
- Payments from different accounts(chequing, savings and credit cards)
- Transactions will be in base currency (CAD) only
- Credit card bill payments treated as purchases instead of cash advances
- Bill payment transaction data will be saved in new database for records
- All interfaces must meet the current timed-out limits in bank(120 seconds)

1.4. Out Of Scope

The following are considered out-of-scope for Minimal-Viable-Product(MVP) :

- Recurring dated payments to billers
- Future dated payments to billers
- Batch process and settlement/reconciliation process of bill payments
- Email notification for upcoming bill payments
- Loan payments
- Credit Card Payment to Third-Parties
- Bill payments to foreign billers
- Commercial banking bill-payments

2. Assumptions, Dependencies and Constraints

2.1. Assumptions

- Pre-approved merchant list exist in bank’s database and is ready to use
- The existing merchant list is up to date and is refreshed with correct data(no duplication)
- Bill-payment system will be provided the below information though biller file :-
    - Biller type indicator
    - Tax indicator if tax is required for payee or not
    - Minimum and Maximum Amount limit

2.2. Constraints

- Customers will not be able to set up reoccurring bill-payment
- Customers will not be able to set up future dated bill-payment
- Customers will not get email reminder for upcoming bill-payments
- Commercial banking will not be able to able to use this for bulk payments

3. Requirements Specification

This section covers the two main types of requirements – functional and non-functional.

3.1 Functional Requirements

The requirements provided in this section describe the high-level functionality of Bank bill-payment, which would be publicly assessable.

3.1.1 Add payee:

Customer should be able to add a payee in the **Select Payee** section of new tab “Bill Payments” in their personal banking system UI.

ID	Description	Business Priority
BP R-01	A new user interface for bill-payments will be created as new tab named “Bill Payments” in personal banking system UI. Customers will be able to access the bill-payment screen by logging into their personal banking account and then clicking on “Bill Payments” tab. This new user interface will help the user to add payee and pay bill online. The main functionalities accessible from this interface are ▪ Manage payee list by linking a new payee from the existing merchant list or remove payee ▪ Real-time bill payments using the existing accounts (chequing, savings and credit cards) ▪ View and download historical payment records **Precondition:** ▪ Pre-approved merchant list exist in bank’s database and is ready to use ▪ The merchant to be added must appear as available in the Available Payee drop-down **Rules:** ▪ Account Number field must be alpha-numeric	High

- UI language will be decided according to customer's login session, i.e. English or Spanish
- Account number must be validated in real-time

Flow:

Customer access Payments screen in personal banking application and selects Bill Payment → Manage Payees

1. Personal banking application should display the required fields and rules when customer selects the payee

2. Customer selects the payee to be added from the Available Payees

3. Personal banking application must provide validation for the Payee Information fields:

o Selected Name – name must match the payee selected from the available payee list

o Account Number Required field, minimum/maximum number of alphanumeric characters vary depending of payee, no special characters allowed only hyphen "-", spaces are not valid,

o Spanish characters will be allowed and translated to standard characters. Example: Spanish character 'ñ' will be translated to 'n'.

o Name of Account Holder – customer must be presented with text underneath the field providing basic validation

o ID of Account Holder - customer must be presented with text underneath the field providing basic validation

4. On Click 'Submit', personal banking application should validate that the required information is present and provide validation results:

o Results must be displayed in the existing error section

o If customer does not enter mandatory field information for payees in Name of account holder or ID of account holder, an error message will be displayed on the Bill Payments screen.

5. If all required information to add the payee are included, personal banking application should present the Add Payee Confirmation page, the fields :

- o Text – Overview:
- o Payee – must match the payee that customer is adding
- o Account number – must match the account number entered by the customer
- o Name of Account Holder – must match the name of account holder entered by the customer
- o ID of Account Holder – must match the ID of Account holder entered by the customer

6. Buttons – On Click

- o Cancel – returns to the manage payee screen resetting all information entered by the customer
- o Previous - returns to the manage payee screen keeping all information entered by the customer
- o Confirm – will add the intended payee along confirmation with message confirmation

7. Editing Rules must validated and displayed in the corresponding fields Account number will be validated to ensure it meets editing rules and it is a valid account number. Validation will be done with merchant real-time.

8. On Click 'Submit', personal banking application should validate that the required information is present and the account number is validated in real-time against merchant.

9. Add Payee Results:

- o Error section – will present validation results
- o personal banking application will display an error message in case the account is invalid or connectivity issues with merchant.

10. If all required information to add the payee are included, personal banking application should present the Add Payee Confirmation page, the fields :

- o Text – Overview:
- o Payee – must match the payee that customer is adding
- o Account number – must match the account number entered by the customer

11. Buttons – On Click

	o Cancel – returns to the manage payee screen resetting all information entered by the customer o Previous - returns to the manage payee screen keeping all information entered by the customer o Confirm – will add the intended payee along confirmation with message confirmation	

3.1.2 Manage Payee:

Customer should be able to Edit/Delete Current Payee in “Bill Payments” in their personal banking system UI.

ID	Description	Business Priority
BP R-02	Authorised user will be able to link a new payee to his/her account for bill payments by choosing it from the drop down list. The drop down list of the payee are sourced from the existing bank’s payee list database The user will also be able to edit the existing payee list linked to his account by removing it **Precondition:** ▪ Payee has been successfully added in the list **Rules:** ▪ No transaction pending with payee to be deleted ▪ Account Number field must be alpha-numeric ▪ UI language will be decided according to customer’s login session, i.e. English or Spanish. User can change the language preference.	High

- Account number must be validated in real-time

Flow:

Customer access Manage Payee screen under Bill Payment tab of personal banking application page, Edit/Delete Current Payee

1. UI behaviour:
    - o Pagination should be present
    - o Select – radio button
    - o Date Added – sortable
    - o Payee – sortable
    - o Account Number – sortable
    - o Edit – Delete – on click buttons
2. Customer selects the payee to be edited or deleted
3. On Click – Edit button
    - o Bill Company / Payee – name of the bill company payee will be no editable
    - o Account Number – must be validated real-time against merchant
    - o Name of account holder - same validation as that to personal banking
    - o ID of account holder – same validation as that to personal banking
    - o On click cancel – It will return to Manage Payees screen and display message, "Edit payee cancelled at your request"
    - o On click submit – It will provides confirmation page.

4. On Click – Delete buttons
    - o On click cancel button – return to manage payees screen and display message, "Delete payee cancelled at your request"
    - o On click confirm button – Delete payees from the list

3.1.3 Bill Payment:

The user should be able to successfully perform a bill payment to the payee selected.

ID	Description	Business Priority
BP R-03	The criteria for successful bill payments should be ➢ User has added payee successfully ➢ Sufficient fund on user’s account or available credit limit ➢ User has available daily/weekly/monthly limits for funds transfer The UI should be able to confirm the transaction to the user by a confirmation message in real time and user should be able to validate the debited amount from his same account immediately. An email confirmation to the user and the payee should be sent out automatically **Precondition:** ▪ Payee must be previously created in the manage payees screen successfully **Rules:** 1. The Bill Payment screen will be dynamic. The behaviour will be determined on if the user selects a Payee from the ‘To’ drop down. 2. Customer payments ‘From’ a Commercial Credit Card, the transaction should be processed in the Credit Card system as a purchase 4. No future dated bill payments allowed 5. No recurring bill payments allowed **Flow** :	High

Customer access Payments – Bill Payment - Create Bill Payment

1. From* - [drop down, required] accounts available for selection will be banking customer's deposit accounts(Saving and Checking accounts) and Commercial Credit Card only

2. To* - [drop down, required] payees previously created by customer

3. Amount [Field, required] – Editable and defaulted to 0.00 - mandatory. Amount is in CAD

   o Customer is not allowed to enter anything less than the minimum or greater than the maximum amount allowed regardless of the balance

   o The payee amount specifications to be available from merchant details, are as follows :

   - Merchant requiring minimum amount limit
   - Merchant requiring maximum amount limit

4. On Click – Submit -> If customer enters an amount which does not correspond to the required amount specifications setup for the merchant in DB, system should display '**ERROR**' next to amount field and a message at the top of page to indicate the rules that need to be follow for this payee when enter an amount (i.e. Minimum amount required. Please enter an amount greater than 10.00 CAD).

   o Amount defaulted value is 0.00 and will clear out when customer clicks in field to enter amount

   o Same behaviour as currently applied will be followed if customer enters an invalid amount(negative value or leave the field blank)

5. Date – defaulted to current business date – Non Editable

   o Effective date of the payment. Date will default to the current business date on system. Date field will not be an editable field for Payees. Process of Bill payment transaction will take effect real time on current calendar date.

6. Frequency – defaulted to 'One Time' Complete 1 Payment(s) – No Editable

   o The frequency will default to "One Time" for "1" payment(s) as recurring payments are not available and out of scope. This should not be an editable field.

7. On Click Submit – Confirm Bill Payment page must be displayed

8. If no tax requirements the fields to present in the confirmation screen are as follows:

   o From – matches the account from account

   o To – payee name along with account number

   o Amount – bill payment amount and currency (i.e. 100 CAD)

o Date – business date
o Frequency - One Time
o Complete # of Payments - 1 - Last Payment on May 27, 2019
o The following piece of the message to be removed [or next business day]

9. If tax requirements the fields to present in the confirmation screen are as follows:

o From – matches the account from account
o To – payee name along with account number
o Amount – bill payment amount and currency (i.e. 100 CAD)
o 'Tax '-<Tax amount><Currency> (i.e. 50.00 CAD)
o 'Total Amount'- <Total amount: sum amount + tax><Currency> (i.e. 150.00 CAD)

10. System will display the tax amount and total amount in the Bill payment confirmation screen. Tax amount and Total amount will be displayed with 2 decimal places, amount will be rounded down if 3rd decimal place is less than five and rounded up if 3rd decimal place is greater than or equal to five(i.e. 10.123 rounded down to 10.12, and 10.126 rounded up to 10.13)

o Date – business date
o Frequency - One Time
o Complete # of Payments - 1 - Last Payment on May 27, 2019
o The following piece of the message to be removed [or next business day]

11. On click buttons – Cancel – Previous – Confirm

o On Cancel – system must return to the Create Bill Payment Screen, values entered by the customer will not be present

o On Previous – system must return to the Create Bill Payment Screen, values entered by the customer will be maintained

12. On click buttons – Confirm.

Once customer clicks 'Confirm' button to process the Bill payment, one of the following will occur in Real time :-

-> If Bill payment is successfully processed :

o A confirmation number will be generated and system will display the confirmation number along with sucess message which will include the reference number from merchant

o System database will retain the confirmation number along with Reference number

o The successful message displayed on the default Bill Payments screen will contain the following information:

Line1. Bill Payment <Reference number> from <Account Number> to <Merchant Name> for <Amount><Currency> successfully completed

Line2. The Confirmation number is <Confirmation Number>

o 'Post Payment' API will debit customer account in real time and display effective date as current business date including weekend or holiday.

o 'Post Payment' API will credit payee GL suspense account and Customer account will be debited on current business date of Bill payment transaction.

o Amount debited from Customer's account will be the amount of the Bill payment transaction OR Total amount which includes tax of the Bill payment transaction to merchant with tax requirement

o History of the Bill payment and Account activity under reporting will display amount of Bill payment transaction to merchant OR Total amount which includes tax of the Bill payment transaction to merchant with tax requirement

o System will change the status of Bill payment transaction as ‘Processed’ and display the Confirmation number in the Bill Payment History details page.

-> If Bill payment is NOT successfully processed :

o If bank cannot connect to merchant OR merchant cannot complete the transaction, system will send error message with reason code to be displayed on UI as confirmation message (customer friendly error message will be displayed).

o System should not reverse a payment without an error message.

If no error message was received, the assumption is that the payment was successfully processed.

3.1.4 Bill Payment History:

ID	Description	Business Priority
BP R-04	The user should have the ability to view and download the historical transaction details from the user interface. The historical data download will be available in excel and pdf format. The user should have ability to filter out the records by date and by payee **Precondition:** o Customer has made a payment to a payee **Rules:** **NA** **Flow:** Bill payment history will display payments history with following fields. By default, transactions will be displayed in	High

	descending order of the Effective Date. o Reference Number – Reference Number of the transaction – sortable o Payee – Name of Payee along with account number – sortable o Account number – account number – sortable o Debit Account – Customer’s account from which payment was made – sortable o Amount – Amount debited from account number – sortable o Effective Date – Date of transition – sortable o Status – Transaction status (Success/Fail) – sortable	

3.1.5 Bill-payment APIs:

This section describes Application Programmer’s Interface (API) requirement for accessing Bill Payment system :-

Any API request made to bill-payment services must contain the following HTTP headers:
ContentType: application/json
Authorization: OAuth Bear authorization token
x-bank-request-timestamp

URL Format: https://bill-payments-<environemnt>.apps.cloud.bankname/<version>/<domain>/<function>/<key>

ID	Description	Business Priority

BP R-05	**Getting a list of available billers :**
	
	Get Available Billers
	
	HTTP Method: **GET**
	
	URL Format: https://bill-payments-ist.apps.cloud.bankname/v0/billers
	
	Example Response:
	"data": [
	 {
	        "id" : "ESL",
	        "name" : "EDESUR",
	        "category" : "Utility"
	 },
	 {
	        "id" : "APL",
	        "name" : "APEC",
	        "category" : "Education"
		High

	}] If all goes well, we receive a 200 OK status code, otherwise a response object with an error would be returned, similar to: Example Error Response. {"error":{"code":"00XX","description":"An error"}} **Possible Responses Codes:** 200 OK. 400 Bad Request. 401 Unauthorized. Authorization header missing or invalid token. 403 Forbidden. Token has incorrect scope or a security policy was violated 404 Not Found. 500 Internal Server Error.	
BP R-06	**Posting a new payment :** Post Payment	High

HTTP Method: **POST**

URL Format: https://bill-payments-ist.apps.cloud.bankname/v0/payments

Example Request:

```
{
"local-tx-id" : "12345ZRET-CXAX",
"biller" : "TRI",
"contract" : "985125020",
"amount" : 500.00,
"currency" : "DOP"
}
```

Balance check along with customer’s daily limit check will happen for Saving & Chequing account w.r.t to amount in post request, before debiting the accounts.
If all goes well, we receive a 201 Created status code, otherwise a response object with an error would be returned, similar to:

Example Error Response.

```
{"error":{"code":"00XX","description":"An error"}}
```

Possible Responses Codes:

201 Created.

400 Bad Request.

401 Unauthorized.

	Authorization header missing or invalid token. 403 Forbidden. Token has incorrect scope or a security policy was violated 404 Not Found. 500 Internal Server Error.	
	Getting information on a posted payment : Posted Payment Information HTTP Method: **GET** URL Format: https://bill-payments-ist.apps.cloud.bankname/v0/payments/{paymentrefno} Example Response: "data": { "biller" : "TRI", "contract" : "985125020", "amount" : 500.00, "currency" : "CAD", "status" : "PENDING", "local-tx-id" : "12345ZRET-CXAX", "payment-date" : "2019-05-27T12:23:50.153-0400", "payment-ref-no" : "1234567890"	High

	} If all goes well, we receive a 200 OK status code, otherwise a response object with an error would be returned, similar to: Example Error Response. {"error":{"code":"00XX","description":"An error"}} **Possible Responses Codes:** 200 OK. 400 Bad Request. 401 Unauthorized. Authorization header missing or invalid token. 403 Forbidden. Token has incorrect scope or a security policy was violated 404 Not Found. 500 Internal Server Error.	

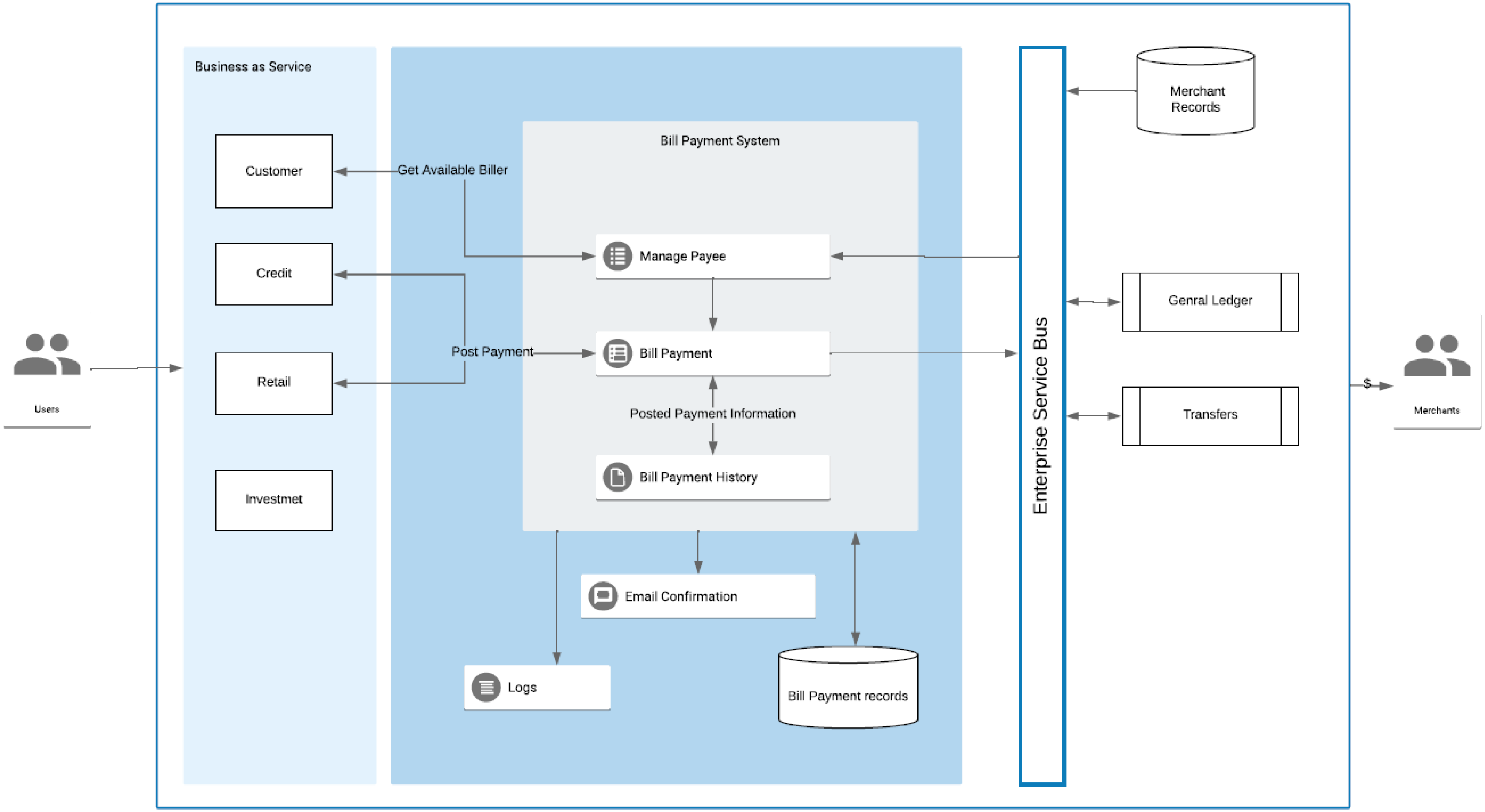

Figure-1: High level diagram for the Bill-Payment system

3.2 Non-Functional Requirements (NFR)

3.2.1 Privacy:

a. Project shall comply with bank’s privacy requirements.

3.2.2 Information Security Requirements

b. Project shall comply with bank’s information security requirements (infrastructure, operations and application security).

3.2.3 Service Level / Performance / Capacity

Below data is provided by the Business and Sales team from market analysis & forecast :-

- Number of concurrent user(Peak): 5000
- Number of transactions: 1TPS
- Work load distribution:

Activity Name	Activity Percentage	Transactions per hour(as per 1tps)	Comments
User logged into system	15%	540	
Idle user	20%	720	
Add Payee	15%	540	
Modify payee	5%	180	
Bill payment	35%	1260	
History	10%	360	

3.2.4 Logging / Monitoring

All transactions occurring in the system will be logged as per bank’s current standards.

Annexe