Business Requirements Specification

Project: Allpay Payment Gateway System

Service:

Date: 01/08/2023

Version No:2.0

Document Information

|  |  |
| --- | --- |
| Criteria | Details |
| Document title: | Allpay Payment Gateway Systemss |
| Document owner: |  |
| Document author: | Surabhi Varma |
| Version: | 1.0 |
| Issue date: | 01-08-2023 |

|  |  |
| --- | --- |
| Criteria | Details |
| Document title: | Allpay Payment Gateway Systemss |
| Document owner: |  |
| Document author: | Vipin Ramakrishnan |
| Version: | 2.0 |
| Issue date: | 03-08-2023 |

Contents

[1.1](#_heading=h.z337ya) Purpose 4

[1.2](#_heading=h.3j2qqm3) Problem Definition 4

[**1.**](#_heading=h.1y810tw) **Chapter 1 – High Level Requirements 4**

[2.1](#_heading=h.4i7ojhp) Current Business Scenario 4

[2.2](#_heading=h.2xcytpi) Proposed Business Scenario 4

[2.2.1](#_heading=h.1ci93xb) Business Context Diagram (To-Be) 4

[2.2.2](#_heading=h.3whwml4) Below context diagram details the proposed system boundaries and how it interacts with any external entities. 4

[2.2.3](#_heading=h.2bn6wsx) Description 6

[2.2.4](#_heading=h.qsh70q) Business Users 6

[2.3](#_heading=h.3as4poj) Business Requirements 14

[2.3.1](#_heading=h.1pxezwc) UC 1.1: Register User and Approval: 14

[2.3.2](#_heading=h.49x2ik5) UC 1. 2: Create Cashbook: 16

[2.3.3](#_heading=h.2p2csry) UC 1.3: Map Cashbook to Bank: 17

[2.3.4](#_heading=h.147n2zr) UC 1.4: Create Head codes: 18

[2.4](#_heading=h.3o7alnk) Non-Functional Requirements 37

[2.4.1](#_heading=h.23ckvvd) Usability Requirements 37

[2.4.2](#_heading=h.ihv636) System Availability/Performance Requirements 38

[2.4.3](#_heading=h.32hioqz) Security & Control Requirements 38

[2.4.4](#_heading=h.1hmsyys) Archive 38

[2.5](#_heading=h.41mghml) Assumptions 38

[**3.**](#_heading=h.2grqrue) **Chapter 2 – Detailed Requirements 39**

[3.1](#_heading=h.vx1227) Requirements 39

[3.1.1](#_heading=h.3fwokq0) Owners, Priority Matrix and Release Classification 39

[3.1.2](#_heading=h.1v1yuxt) Functional Requirements 40

[3.1.3](#_heading=h.4f1mdlm) Non-Functional Requirements 40

[3.1.4](#_heading=h.2u6wntf) Solution Constraints & Standards Compliance 45

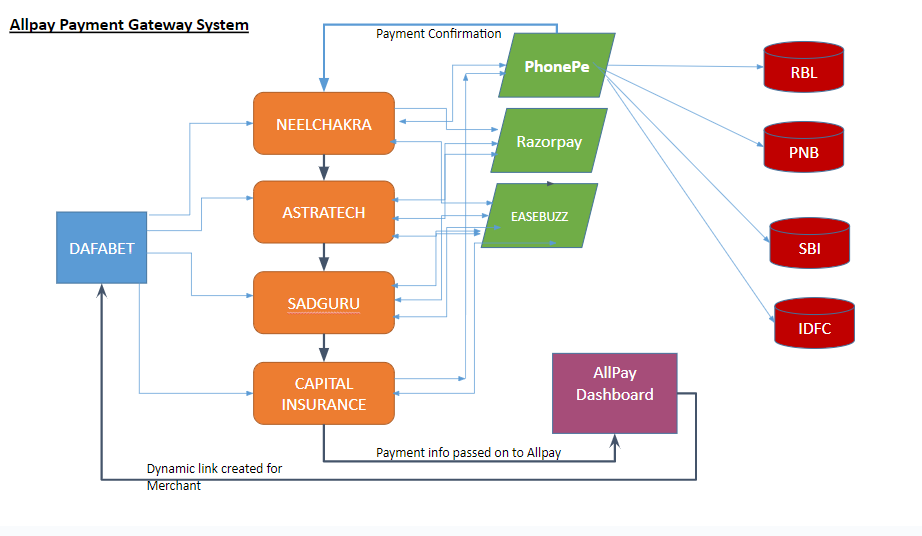
[3.2](#_heading=h.19c6y18) Outstanding Issues 45

[3.3](#_heading=h.3tbugp1) Assumptions 45

[**4.**](#_heading=h.28h4qwu) **Updated Requirements 45**

[**5.**](#_heading=h.nmf14n) **Requirements Traceability Matrix 46**

[**6.**](#_heading=h.37m2jsg) **Appendix 46**

Chapter 1 – Business Concept Diagram 

The steps taken to complete the payments through the All Pay application is as followed

1) All Pay system generates a dynamic link with the params for Betting Site.This Dynamic link is

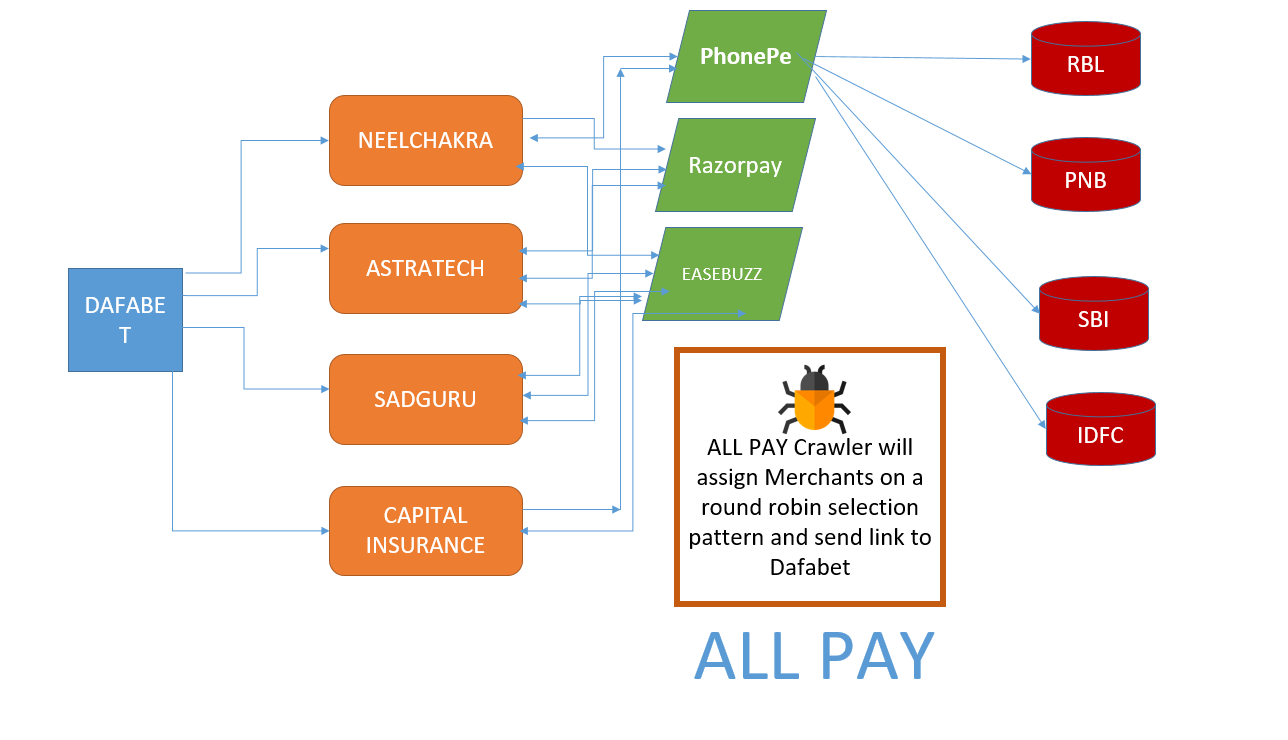
Passed to the Betting site user.

2)- Betting site will replace actual value of amount, callback url, dafabet order id, user details and other details. Then present this link to user.

2) When the user clicks on the link they are directed towards the Merchants website with Checkout page to complete the pay action

3) Once the Payment is processed by the Payment gateway the confirmation for the same is send to All pay system to share the details of payment.

4) Once the Payment status is verified by All Pay system the same data is sent across to Betting partner for synching the information.

1. 

## Dashboard:

## Prerequisites:

## Once an application is created, all users will be created who are going to operate/Use application.

## For creating user profile Name and Email ID of user are used.

## Merchants

* + - * 1. Merchant details must be shown in list view, containing details like Merchant Name, Status (active/inactive), Payment Gateway which is used

## Also, provide a merchant report in any convenient pictorial format like Pie chart/Bar graph. Especially showing transactions of all merchants in one diagram.

## We will have combination of merchant and payment gateway and each combination works independently so we need to work with that. We are showing pictorial representation of this only. And nothing else. On dashboard additionally we will have bar charts for daily, weekly, monthly and yearly transactions

## Payment Gateway

Upi is just a mode of payment provided by all Payment gateways along with debit card, netbanking, credit card etc. So this point can be removed, I think.

* + - * 1. UPI Gateway

UPI Gateway screen must show all UPI gateway details in list view containing details like Gateway Name, Gateway Status.

Also, provide Daily UPI Transaction report and UPI Performance report (Daily and monthly) in any convenient pictorial format.

* + - * 1. Payment Gateway

Payment gateway screen must show all Payment gateway details in list view like Payment Gateway name, Gateway status.

## We will have combination of merchant and payment gateway and each combination works independently so we need to work with that. We are showing pictorial representation of this only. And nothing else. On dashboard additionally we will have bar charts for daily, weekly, monthly and yearly transactions

* + - 1. *Transaction*
         1. Transaction details must be shown in list view, containing list view like Order ID, Amount, Payment gateway/UPI Gateway, Status (pending/completed).
         2. Daily transaction and monthly transaction (average) must be shown in any convenient pictorial format.

## Betting Partner

## Betting partner details must be shown in list view containing all detail like Partner name, API Status, location, Contact manager

## Betting partner transactions must also be shown in any convenient pictorial format.

* + - 1. *Reports*
         1. All required reports list must be available in this screen like Merchant report, Payment Gateway report like UPI Gateway report (Daily and monthly) and Payment gateway report (Daily and monthly), Betting partner report,

1. Chapter 3 – Allpay API Documentation
   1. Get available merchant and payment gateway

Get available merchant and payment gateway

Method: GET

Url: /PaymentGateway/availablePGM

Body: None

Headers: {

“Authorization”: “Bearer {BettingPartnerToken}”

}

Status Code: 200

Response Type: JSON

Response: {link: “RESULT\_LINK”}

Link Params:

|  |  |
| --- | --- |
| Mid | Merchant Id |
| Pg | Payment Gateway |
| pgid | Payment Gateway Id in All-Pay server |
| Amount={amount} | Replace {amount} with the actual amount in the link |
| bpid | Betting Partner Id |
| Partner\_orderid={poid} | Replace {poid} with order id in dafabet portal for unique identification |
| Prod\_desc={desc} | Replace {desc} with description of order detail. If many fields are present then send encoded json |
| Username={username} | Name of user on dafabet |
| Useremail={useremail} | Email of user on dafabet |
| Userphone={userphone} | Mobile number of user on dafabet |
| Extra\_info | Extra Info in json format and encoded as string |
| allpayCallbackUrl={callbackurl} | Callback url where Allpay server can notify betting partner about the status of transaction |

Note: BettingPartnerToken can be generated from the AllPay portal. This token is alive for a long time, so one should take care of securing it properly on server. It should NOT be shared by anyone. In case it is compromised then new token should be generated and updated on server for sending further request.

1. Submitting transaction to server

Method: POST

Url: /Transaction

Body:

|  |  |  |
| --- | --- | --- |
| Parameter | Type | Description |
| bpid | String | Present in the link in query params. |
| partnerOrderId | String | Obtained from Betting partner portal for identifying transaction has happened against which order |
| partnerUserId | String | Obtained from Betting Partner portal for identifying user for which transaction has happened |
| partnerUserDetail | Json | Obtained from Betting portal. Username, email, mobile etc |
| Amount | Number | Obtained from betting portal |
| paymentGatewayId | String | Present in the link in query params. |
| paymentMode | String | UPI/CREDIT CARD/DEBIT CARD/NETBANKING |
| paymentDetails | JSON | Payment Details as much as possible, obtained from payment gateway |
| extra\_info | JSON | Additional Information from betting partner |
| comment | String | Any extra notes |
| status | STRING | COMPLETED, PENDING, CAPTURED etc Status from payment gateway |
| allpayCallbackUrl | STRING | Obtained from betting portal |

Status Codes:

|  |  |
| --- | --- |
| Success | 201 |
| Permission Denied | 403 |
| Internal Server Error | 500 |

Headers: {

“Authorization”: “Bearer {MerchantToken}”

}

Response Type: JSON

Response: {“msg”:”Transaction Created”}

Note: MerchantToken can be generated from the AllPay portal. This token is alive for a long time, so one should take care of securing it properly on server. It should NOT be shared by anyone. In case it is compromised then new token should be generated and updated on server for sending further request.

**Get Transaction Status - For betting partner, in case they want details of any transaction later**

Endpoint Url: /Transaction/getTransactionByPartnerOrderId

Method: GET

Headers: Authorization: Bearer {PartnerToken}

Query Params: poid: {Betting partner order id}

|  |  |  |
| --- | --- | --- |
| s | String | Transaction Status |
| createdAt | number | Creation Timestamp |
| id | string | All Pay transaction Id |
| poid | string | Partner Order Id |
| puid | string | Partner User Id |
| pud | json | Partner User Detail |
| a | number | Transaction Amount |
| pg | string | Payment gateway id on Allpay server |
| pm | string | Payment Mode |
| pd | json | Payment details |
| ei | json | Extra Info provided by betting partner |
| c | string | comment |
| bp | string | Betting Partner Id |
| m | string | Merchant Id |