1. merchantlogin/verifyOtp - otp not coming to phone - Chethan to provide option to access OTP to enter for OTP verification on device  
2. merchantlogin/verifyIdentity - should be handled for both Tier-1 & Tier-2 - Personal fields should be option al for Tier 2 - Fixed  
3. onboard/setcredentialorcreateaccount -   
   Blocker 3.1 - Account details not reveceing in response Blocker for Tier-1 - Chethan   
   Blocker 3.2 - Seeing Error frequently - "There was a problem communicating with the core banking system" - Chethan  
4. merchantdashboard/profile - API not yet received - Chethan  
   4.1 - KYCLevel and TierLevel - should have currently upgraded Tier level  
5. accounttier/getBasicInfoDropdown -  Yet to receive the API  
6. accounttier/savePersonalInformation - Yet to receive API  
7. Onboard/saveBusinessInformation - Yet to receive API  
8. sumsub/getApplicatStatus - Yet to receive API  
9. generateTokenOrWebSDKlink - works(Tested in Postman) need to test again in device once all apis are ready

QuoreID swagger : <https://docs.qoreid.com/docs/introduction>

Sumsub swagger : <https://docs.sumsub.com/>

CBA swagger : <https://docs.qore.inc/>

Could you please clarify on the below  
1. Tier 3 - 1. Indemnity form do we need to store it in merchant server database securely?

2. Ensure the re-entered name matches BVN/NIN data - Validation in Indemnity form will be done at Client side?

2. Tier 4 - Verify CAC details via integration with the CAC API (if available) or manual review.   
1. Need clarity on this if CAC API integration available or if need to do manual review will that be done at client side or server side

2. CAC certificate can be stored in merchant server database?  
3. Merchant Menu Controls if need to be assigned from Portal   
 1. Will there be different roles created for Merchant in Portal?  
 2. How the merchant will be mapped to a particular role as while onboarding no role type is specified?  
4. User Management  - What all functionalities should be there in Portal?  
5. Portal access will be only to Admins or Merchants also?  
6. Sub Merchant onboarding process is not clear like what all de

**Merchant Login on another device:**

Can a merchant login to a different device? If allowed how is it taken care of:

1.will first be auto logged out if user is trying to login to different device.

RISKS

If multiple logins allowed, then it may cause transaction issues

How are the records to be maintained then?

If dispute occurs how will it be traced. So every login and logout should be traced right?

Geolocation Is used to prevent frauds

We need to onboard Agent Merchant an Retail Merchant

Retail Merchant: Who will accept the payment against the goods/services he will sell

Agent Merchant: Who will do the Bill payments, money transfer, cash withdrawal etc. in short (Cashin and cash out)

The Difference would be just that the Agent Merchant would need access to like Bills, Money transfer etc whereas in the case on Retail merchant the access would not be there.

Rest I guess onboarding would be same right -> verification of address, KYC etc.

Customer Sends money to the agent merchant’s wallet transfer funds to the Virtual Account Number.

Once transfer hits the Virtual Account number, system identifies the linked e-wallet account of the agent merchant.

The e-wallet platform credits the exact amount into the merchant’s wallet balance.

The virtual account acts only as receiving bank account.

Agent merchant can withdraw or transfer the funds to the bank account or as physical cash.

**MOM Figma Design -10/09/2025**

**Agent – Bill Payment, Airtime, Transfer**

Two Leg operation-

Leg 1- Receive- Customer pays the agent(cash/card/transfer)- Amount credited to the agent’s e-wallet.

Leg 2-Agent initiates the Payment or transfer transaction using wallet balance.

**Airtime Screen:**

1.Select the operator.

2.Enter the phone Number

3.Enter the amount.

Note: No need to showcase the plans.

Kamal to update the Figma design

**Bill Payments**

Example Electricity: Input taken from the user is only “Consumer Meter Number”

No need to take the input of the consumer name.

The system should fetch the name on the account.

**Payment Method: “Pay by Card”**

Once the “Pay by Card” is selected as Payment method- the application is routed to sPOS application. Before the customer taps the card on the sPOS screen, Agent needs to Select the Account Type (Savings, Current or Credit)

**USSD code generation:**

Agent first needs to provide a list of banks. Select the customer bank.

Many banks (List provided by Oluwatobi) have their own fixed USSD codes.

Generally, a code would be generated like

Fixed bank USSD code + Amount + Agent Wallet ID

So based on Bank and Amount the code will change.

Correct me if wrong.

**Fee:**

Merchant Fee: Pre-configured based on the no of transactions.

Agent fee: Two options can be provided

1. System calculates the automatically and credited to agent.
2. Manual input: While initiating a transaction, agent adds a service fee and then deduct the transaction amount + fee from the customer.

@Uloma and @Oluwatobi to discuss this internally in house and provide the update.

**Cash Withdrawal:**

One more input for the agent to receive the amount to be added.

If the customer uses his own mobile banking application to transfer the amount to the agent’s wallet.

On the Business eWallet screen provide one more option like “Bank transfer” where customer will initiate the transfer through his mobile banking application and the sPOS application will check continuously until the payment is received.

Agents confirms the payment and hands over the cash to the customer.

**Pan Number**

First Masked-masked-Last four

**For reference**



Please find the details as below:

1. Given the Demo on:
   1. Tier 4 application flow.
   2. Purchase transaction->Invoking sPOS application -> Tapping Card and getting the receipt.
2. Application will have different types of Merchants to be on boarded
   1. Retail Merchant: Having access to Purchase, Bill Payments module
   2. Agent Banking: Having access to Withdrawal, Fund Transfer, Bill Payments (Transactions related to POS.
   3. Hybrid: Both Retail and Agent Banking module access.

Note: Need to discuss with @Oluwatobi on the access to be provided to different features of application  to different types of merchants based on the tier level and merchant type and the validation rules  and message prompts accordingly.

1. Once the CAC certificate validation is done using sumsub at tier 4 , need to fetch the business name from CAC and use it on the receipt.The name is considered as the authorative name for the entity.

Note: team need to check whether can we get the name on the certificate in response from sumsub or need to find out the solution on how to

extract it from the pdf.

**Agent and Merchant onboarding**

**Primary Role-Retail Merchant-** Accept customer payments

**Agent Merchant-** Provide financial services

Will the merchant be allowed to proceed to Tier3 and Tier 4

Also for Agent Merchant- Load the wallet should be allowed.

**Indemnity Form**

User finishes Tier 3 upgrade.

At this point user will see 2 CTA buttons.

Go to dashboard

Upgrade to tier 4

If proceed to Dashboard is selected after completing Tier 3 – User has to fill up the indemnity form. Skipping is also allowed while proceeding to dashboard.

I know that if the user skips to fill the Indemnity form then a path should be provided to the user to fill up the indemnity form, through profile settings.

But I want to understand here is, when is the indemnity form mandatory to fill up. Before upgrading to Tier 4?

So that while upgrading to Tier 4, the user can’t skip to fill it.

Also we would need the updated Figma design to be accordingly.

**USSD Code Generation**

 The USSD code already includes the **bank**, **merchant ID**, and **amount** → customer doesn’t need to enter anything except their PIN.

 Reduces user error and makes the process faster.

**USSD String Generation**

* The system builds a **dynamic USSD code** in the format required by that bank.
  + Example (Bank A): \*123\*4567\*1000#
    - 123 = Bank A’s shortcode
    - 4567 = Merchant ID
    - 1000 = Amount
  + Example (Bank B): \*456\*4567\*1000#
    - 456 = Bank B’s shortcode
    - 4567 = Merchant ID
    - 1000 = Amount

## Step-by-Step Flow: Merchant Selects Bank First

### ****1. Initiation****

* Customer requests to pay via **USSD (Business eWallet)**.
* Merchant enters the **transaction amount** in their POS/app.

### ****2. Bank/Wallet Selection****

* POS/app shows a list of supported banks/wallets:
* Select Customer Bank:
* 1. Bank A
* 2. Bank B
* 3. Wallet X
* Merchant asks customer which bank/wallet they use.
* Merchant taps the correct option.

## 🔹 Why This Step Is Needed

* The **merchant POS cannot know** which customer account to debit.
* The **bank controls this menu** inside their USSD gateway.
* Merchant’s role is only to pass **merchant ID + amount** into the USSD string.
* The **account type selection** remains the customer’s responsibility (just like choosing an account when withdrawing at an ATM).

For my Understanding

Customer → Merchant → Merchant Wallet → Belema Wallet → External Biller (bank account)

1. Customer pays through merchant platform.
2. Money lands in the **Pool Account** (via bank/UPI/card).
3. Middleware identifies which **Virtual Account** it belongs to and credits **Merchant Wallet** internally.
4. Middleware will handle the debits/credits.
5. From **Merchant Wallet** it is transferred to **Belema wallet**. This is internal transfer.
6. Once Belema Wallet has the balance, middleware initiates a **payout to the biller’s bank account**.
7. This is where **bank integration/payment gateway APIs** come into play.

* **Customer → Merchant → Merchant Wallet → Belema Wallet → External Biller (bank account)**

Here’s how that works step by step:

## 🔹 Extended Payment Flow

### ****1. Customer Payment****

* Customer pays through merchant platform.
* Money lands in the **Pool Account** (via bank/UPI/card).
* Middleware identifies which **Virtual Account** it belongs to and credits **Merchant Wallet** internally.

### ****2. Merchant to Belema Wallet****

* Merchant owes the biller, so it transfers from **Merchant Wallet → Belema Wallet**.
* This is purely an **internal ledger movement** within your system (no bank involvement yet).
* Middleware handles debits/credits and applies rules (fees, commissions).

### ****3. Belema Wallet to External Biller****

* Once Belema Wallet has the balance, middleware initiates a **payout to the biller’s bank account**.
* This is where **bank integration/payment gateway APIs** come into play.
* Options:
  + **IMPS/NEFT/RTGS/UPI** if biller’s account is in India.
  + **API to partner bank** if in another region.

So middleware:

* Debits **Belema Wallet** (internal ledger).
* Triggers **bank transfer API** (e.g., POST /payout) to push money from Pool/Nodal Account to the biller’s **external bank account**.
* Marks status as **pending** until the bank confirms settlement.

## 🔹 Role of Middleware Here

1. **Internal movement** → Merchant Wallet → Belema Wallet.
2. **External settlement** → Belema Wallet → Biller Bank Account.
3. **Reconciliation** → Ensures the amount leaving Belema Wallet equals amount confirmed by the bank.
4. **Notifications** → Updates both Merchant and Biller on status.

## 🔹 Visual Flow

Customer Payment → Pool Account → Merchant Wallet

↓

Belema Wallet (internal)

↓

[Bank API: IMPS/NEFT/RTGS/UPI]

↓

Biller Bank Account (external)

✅ In short:

* **Internal eWallet movements** happen instantly within your system.
* **External settlements** require integration with **banking rails** via middleware.

Reconciliation

Bank ->VA

Wallet Vs bank Balance

Settlement

Bank = Opening Balance+Credits-Debits=Closing balance

Ewallet

Opening wallet balance + merchant Credits –Belema Debits-Closing balance

Pool Account Closing Balance = Total of all Wallet Balances

Reports

Transaction reports for every merchant

Complaince report= VA inflows=Merchant credits

Settlement report

