

Core Banking

- [1. Core Banking: Overview](#)
- [2. Core Banking: Use Cases](#)
- [3. General Settings: How to Configure Core Banking](#)
- [4. How to Set Company Fees](#)
- [5. How to Set Payment Methods](#)
- [6. How to Set Platforms](#)
- [7. How to Manage Accounts](#)
- [8. How to Manage Assets](#)
- [9. How to Manage Transactions](#)
- [10. How to Manage Banks](#)
- [11. How to Manage Cards](#)
- [12. Configurations](#)
- [13. Invoices](#)

1. Core Banking: Overview

The Core Banking module in Wifox Business Core Solution (WBCS) provides a centralized, API-driven platform for managing all your institution's financial accounts and transactions in real time. Key features include:

- 1. Universal Ledger:**
A single, immutable record of all account balances, deposits, withdrawals, transfers and fees—across projects, desks and client portfolios.
- 2. Account Lifecycle Management:**
Open, suspend, close or reassign digital accounts with full KYC/AML enforcement, custom account types (e.g. checking, savings, escrow) and IBAN assignment.
- 3. Real-Time Transactions:**
Process intra-bank transfers, external ACH/SWIFT payments and QR-code or card-based transactions instantly, with immediate balance updates and notifications.
- 4. Multi-Currency & FX Support:**
Hold and transact in multiple currencies per client or project with live exchange-rate feeds, automatic currency conversion and cross-currency fee schedules.
- 5. Configurable Interest & Fees Engine:**
Define tiered interest-rate schedules (daily, monthly, annual), overdraft penalties, maintenance fees and one-off charges—all computed automatically per account.
- 6. Role-Based Security & Audit Trails:**
Fine-grained permissions control who can view, initiate or approve transactions; every action is logged with user metadata for compliance and forensic review.
- 7. Regulatory & Management Reporting:**
Out-of-the-box balance sheets, P&L reports, transaction journals and custom report builder—exportable in PDF, CSV or JSON for regulators, auditors and executives.
- 8. Extensible API Layer:**
Fully documented REST endpoints for account inquiry, transaction posting, batch settlement and webhook notifications—easy to integrate with mobile apps, payment gateways or partner platforms.

Core Banking module includes the following sections:

- 1. [Accounts](#):** Serve as containers for managing financial activities and transactions within the system.
- 2. [Assets](#):** Represent various forms of value, including cryptocurrencies, traditional fiat currencies, trade-ins, NFTs and more, and are assigned to specific accounts.
- 3. [Banks](#):** Stores information about the details of the banks in use.
- 4. [Cards](#):** Stores information about the details of the cards in use.

5. [Transactions](#): Displays deposits, withdrawals, and other activities involving the client's financial resources. Global model for accounting of cash flow in your system/product.
6. [Payment Methods](#): Contains information about payment systems, allowing users to pay for services and goods on your site.
7. [Platforms](#): Customizes integration with banks and other platforms that store information about accounts and assets.
8. [Configurations](#): Manages general settings for the Core Banking module.
9. [Company Fees](#): Customizes the fees the company charges during transactions and manages fee groups.
10. [Invoices](#): Manages billing records issued to clients, including invoice creation, status tracking (Pending, Paid, Expired, etc.), related transactions, postbacks, and supporting documents.

2. Core Banking: Use Cases

Use Case #1: Customer Account Onboarding & KYC

Automatically create new digital accounts when a client is approved. Trigger identity verification workflows (via integrated AML/KYC services), assign the correct account type and send welcome notifications—all without leaving WBCS.

Use Case #2: Real-Time Payment Processing

Enable tellers, back-office staff or automated jobs to post deposits, withdrawals and internal transfers in seconds. External payments (SWIFT/ACH) flow through the same ledger, with status updates (pending, settled, failed) shown live.

Use Case #3: Automated Interest & Fee Charging

Schedule daily balance snapshots to calculate interest on savings accounts, automatically debit maintenance fees or overdraft charges, and post accrual transactions at month-end—freeing finance teams from manual spreadsheets.

Use Case #4: Multi-Currency Treasury Management

Manage pools of foreign-currency balances for corporate clients or branches. Execute spot conversions or forward-contracts via the FX API, and reconcile P&L on currency spreads with a click.

Use Case #5: Regulatory Reporting & Audit Compliance

Generate end-of-day journals, AML transaction reports and capital adequacy summaries from within WBCS. All ledger entries and user actions are timestamped and tamper-proof, simplifying audits and regulatory filings.

3. General Settings: How to Configure Core Banking

The **Configurations** tab in Wifox Business Core Solution (WBCS) is your central hub for defining the building blocks of the Core Banking module. Every change here immediately becomes available system-wide—whether you’re opening new accounts, posting transactions, or generating reports.

To configure the general settings for **Core Banking**, open the **Configurations** tab.

This tab includes four sections, where you can add, edit, or delete items:

- Types of Account:** Customize account types (e.g., Business, Personal, Crypto, Fiat).
These account types can later be selected when creating accounts through the [Accounts](#) module.
- Currencies:** Define the currencies and networks (for cryptocurrencies) that will be processed by the system, such as USD, USDT, BNB, etc. Without defining currencies here, you will be unable to perform many activities in the CRM: e.g. to set company commissions for different transaction types or to add assets to customer accounts.
- Sub-types of Transactions:** While the system has two main transaction types — **In** and **Out** — you can add sub-types for further classification. By default, sub-types like **Exchange**, **Withdrawal**, and **Deposit** are included. You can also create additional sub-types here based on your custom logic.
- Transaction Statuses:** Track the current state of transactions, with default statuses like Pending, Processing, Completed, and Canceled. These statuses are predefined in our system, and we have the own logic to handle transactions with it. You can also add or modify custom transaction statuses here.

To add a Types of Account

1. Locate the section on the left side of the Configurations page labeled **Types of account**.

2. Add a new type:

Click **Add** (top-right of that panel).

In the drawer that opens:

Label: enter a unique identifier (e.g.)

Name: the human-readable title (e.g. “Corporate Account”)

Properties (optional): use the node picker to attach custom metadata fields for API-driven logic.

3. Click **Save**.

4. Edit an existing type:

Click the pencil (✎) icon on the row of the type you wish to modify.
Update Label, Name or Properties, then **Save**.

5. Delete a type:

Click the trash-can icon on its row.
Confirm deletion in the prompt.

To add a Currency

Defines which fiat or crypto assets can be held and transacted.

1. Find the Currencies panel on the right side of the Configurations page.

2. Add a currency:

Click **Add** in the panel header.

In the drawer:

Type: choose **Fiat**, **Crypto**, or **Custom**.

For Fiat/Crypto:

1. Select the currency code from the dropdown (e.g. USD, BTC).
2. **Limit:** set a minimum transactable amount (optional).
3. **Network fee:** for cryptos, enter the on-chain fee (e.g. BTC).

For cryptocurrencies, you can also specify a **Limit** (minimum amount to process) and a **Network fee**. Transactions below the set limit will not be conducted, and the specified network fee (e.g. ERC20 for USDT) will be charged additionally for the transaction.

For Custom:

Enter **Label**, **Name**, **Network** and **Description** as needed.

Check **Create asset within condition** if you want the system to auto-generate token assets under your business rules.

3. Click **Save**.

4. Edit a currency:

Click its pencil (✎) icon.

Adjust any parameters, then **Save**.

5. Delete a currency: Click its trash-can icon and confirm.

You cannot delete the one marked **Default** unless you first choose a different default.

To add a sub-type of transaction

(Helps you categorize “In” vs. “Out” further—e.g. Exchange, Fee.)

1. Locate the Sub-types of transaction panel at bottom-left.

2. Add a sub-type:

Click **Add** in the panel header.

Enter a **Label** (machine ID) and **Name** (display text).

3. Click **Save**.

4. Delete a sub-type: Click the trash-can icon next to any non-default entry.

Note: you cannot edit existing sub-types.

To add a transaction status

(Tracks each transaction’s lifecycle: Pending, Completed, etc.)

1. Find the Transaction statuses panel at bottom-right of the **Transaction statuses** section.

2. Add a status: Click **Add** in the panel header.

In the drawer:

Label and **Name** for your new status (e.g. / “Reversed”).

Color: pick from the palette for easy visual scanning.

Make as default: check this to have new transactions start in this status.

3. Click **Save**.

4. Delete a status: Click its trash-can icon.

Note: default system statuses (e.g. Pending, Completed) cannot be deleted.

Each time you **Add**, **Edit**, or **Delete** one of these items, the change is effective immediately across all WBCS Core Banking operations—no system restart required.

By tailoring account types, currencies, transaction sub-types and statuses to your organization’s needs, you ensure that every ledger entry, API call and report aligns precisely with your business logic and compliance requirements.

4. How to Set Company Fees

To set **Network fees** for cryptocurrencies, you can configure them when adding a cryptocurrency in the [Configurations](#) section. If you need to set **Company fees**, navigate to the **Company Fees** section of the Core Banking module. All specified fees, including network and company fees, will be totaled and applied during transactions.

In the **Company Fees** section, you can create both **static (fixed)** or **percentage-based** fees for different transaction types. You can also group these fees. **Fee groups** are useful for assigning specific fees to different types of [Clients](#) when creating or editing client profiles. For example, you can create a group for *developers* that is exempt from fees (0% fee).

Navigate to the Company Fees Tab

In the left sidebar, click the **Bank icon** to open **Core Banking**.

Select **Company Fees** from the submenu.

You'll arrive at a table showing your existing fee groups and fees (if any).

To add a new fee group

1. Click the **Add Group** button in the top-right corner of the **Company Fees** tab.

2. In the drawer that appears:

Name: Enter a human-friendly title (e.g., Spot Trading Fees).

3. Click **Save** to apply the changes.

4. The new group appears in the table.

To add a new fee

You can add a fee either globally (via the top button) or directly under a specific group.

A. Via the Top-Level Button

1. Click the **Add Fee** button in the top-right corner of the **Company Fees** tab.

Alternatively, click the plus icon in the relevant group row to add a fee directly to a specific group.

2. Fill in or select the following fields:

Company Fee Group: This will be pre-filled if you're adding the fee via a specific group.

Type of Transaction: If no transaction types are available, go to **Core Banking > Configurations** to create them.

Currency: For **Exchange** transactions, select two currencies (one for **In** and one for **Out**). If no currencies are available, go to **Core Banking > Configurations** to create them. Alternatively, leave this field blank to apply the fee to all currencies.

3. Next, select the type of fee:

Static: A fixed fee amount.

Percent: A percentage-based fee.

4. Enter the following details:

Min. Value: The minimum transaction amount to which the fee applies (leave blank for no minimum).

Max. Value: The maximum transaction amount to which the fee applies (leave blank for no limit).

Amount: The fixed fee or percentage that will be charged per transaction.

You can add multiple commissions for different thresholds at once. For example:


1. For value from 0 to 100: 10%
2. From 100 to 1000: 15%
3. From 1000 and above: 20%.

To add additional fees, click the plus icon again after entering the first fee. The next line will appear.

To edit or delete created fees, click on the **Edit** or **Delete** icon:

5. Click **Save** to apply the changes.

B. Directly Under a Group

1. Locate the group's row in the table.
2. Click the  icon in that row's **Actions** column.
3. The same **Add company fee** drawer opens with the **Company fee group** pre-selected.
4. Complete steps 2-4 above to define and save the fee.

To view all fees associated with a group

Click the Expand icon next to the group:

1. **To edit a group or fee:** Click on the **Edit** icon in the relevant row.
2. **To delete a group or fee:** Click on the **Delete** icon in the relevant row.

Warning: Deleting of group is only allowed when no client is assigned to it. Deleting a group will automatically delete all fees associated with it. This action is irreversible.

Tips & Best Practices

1. Use clear and descriptive Names.
2. Tiered fees let you incentivize larger volumes: lower percentages for higher thresholds.
3. Leave currency blank on a fee to apply it universally across all assets.
4. Test new fees on a sandbox account before rolling out to production.

By following these steps, you'll have fine-grained control over every commission your company charges—ensuring accuracy, transparency, and the flexibility to adapt as your business evolves.

5. How to Set Payment Methods

The **Payment Methods** module is used to configure payment systems (e.g., ApplePay, LiqPay) that allow users to pay for goods and services on your site. You can set up two types of payment methods in the CRM:

1. Deposit
2. Withdrawal

You can perform the following actions for both deposit and withdrawal payment methods:

1. Adding
2. Activation/Deactivation
3. Searching
4. Editing
5. Deletion

Additionally, deposit methods can be activated or deactivated as needed.

To add a new deposit payment method:

1. In the left-hand menu click **Core Banking** → **Payment Methods**, then select **Deposit**.
2. Click **+Add**: Look at the **upper-right** of the Deposit table header (just above the “Actions” column).

3. Select the following from the drop-down list

Type is the first field on the left:

Auto: Fully set up the payment system on your side, so the user can make payments directly through your site.

Manual: Transfers the user to the payment system’s site via a link for payment.

Integration dropdown sits immediately to its right:

Internal: For your internal payment systems. When Server-to-Server (S2S) integration occurs and your platform allows the user to enter a card on the site.

External: For third-party systems. When the user enters the payment amount and clicks Pay, they are taken to the payment system page where they fill in the card details, etc.

4. Enter a **Name** for the payment method.

5. In the **Label** field, select the required payment system from those already integrated into the CRM to automatically populate the remaining fields. Alternatively, you can fill in these fields manually.

6. If necessary, fill in or edit the following fields:

Label: The user-facing name (e.g. “Bank Transfer”).

Name: Internal identifier (used in API/webhooks).

Merchant ID: Your processor’s account ID.

Currencies: Select which currencies are allowed (expand the array and tick each one).

Language: Choose the default language for this method’s UI.

(Optional) Configure advanced settings:

Rules: JSON object defining fraud checks, min/max amounts, etc.

Credentials: API keys or tokens for your gateway.

Additional: Any custom JSON your integration needs.

Placeholder: Custom “enter details” text.

Note: These fields are used for integration and are configured by developers or the integration team.

7. Click **Save** at the bottom of the panel to create the payment method.

After creation, a payment method is not automatically activated. To activate it, find the method in the list and toggle the switch in the **Status** column.

To add a new withdrawal payment method:

1. Click the **Add** button in the upper-right corner of the **Withdrawal** tab.
2. Enter a **Label** and a **Name** for the method.
3. If necessary, fill in the **MerchantID** and **Currencies** fields.
4. Select the appropriate **Language** from the available options.
5. Click **Save** to apply the changes.

To search for a payment method:


“ Works in both (Deposit and Withdrawal) tabs.

Click into the **Search...** field at the top-left.

Begin typing the payment method’s label in the **Search** field at the top of the tab.

To edit a payment method:

Works in both (Deposit and Withdrawal) tabs.

1. Locate the row you want to change.
2. Click the  **Edit** icon in the rightmost **Actions** column.
3. Update any editable fields in the panel (all except **Label**).
4. Click **Save** to apply your changes.

Note: You cannot edit the label of a payment method.

To delete a payment method:

“ Works in both (Deposit and Withdrawal) tabs.

For deposit methods, ensure the method is deactivated first. Deactivate a method by clicking the toggle in the **Status** column (deactivated methods will turn gray).

Click the **Delete** icon in in the **Actions** column.

Warning: Deleted payment methods cannot be restored.

6. How to Set Platforms

The **Platforms** module is used to integrate and customize third-party platforms (e.g., banks) that manage customer assets and accounts. If an asset or account is assigned to a platform, connecting to that platform is essential to display relevant information, such as balances.

Use Cases

1. **Switching Between Projects**

Select a project to view or manage its unique financial configurations. If no settings exist, the section displays "No data to display" until configurations are added.

2. **Cloning Configurations for New Projects**

Use the "Clone from project" feature to copy currencies, transaction sub-types, or account types from an existing project, saving time and maintaining consistency across projects.

3. **Cloning Currency Configurations**

Clone currency settings (e.g., USD, EUR, USDT, TRX) from a project like Wibroker to a new one (Incur) to avoid manual currency entries, ensuring quick and consistent setup.

4. **Cloning Transaction Sub-Types and Account Types**

Apply existing configurations by cloning Sub-types of Transactions or Types of Accounts from one project (e.g., Cypress Test Default) to another (Incur), instantly reflecting them without manual input.

To add a new platform:

1. Click the **Add Platform** button in the upper-right corner of the **Platforms** module.
2. Enter or select from the drop-down list:
 - Name:** The name of the platform.
 - Label:** A unique identifier for the platform. The list will include platforms that are already integrated into CRM.
3. Add the necessary **URL(s)** and **Credentials** required to connect to the platform. If you select a label from the drop-down list, the required fields will be automatically populated.
4. Click **Save** to apply the changes.

To edit a platform:

1. Click the **Edit** icon next to the platform.
2. You can change the **Name** and update or add new **URL(s)** and **Credentials** (or modify existing values).
3. Click **Save** to apply the changes.

To delete a platform:

Click the **Delete** icon next to the platform.

Note: You can only delete platforms that have no associated assets. For more information on managing assets, refer to the [\[Assets section\]](#).

Viewing Platforms

Platforms Page

For instance, we have two existing platforms: “**platform**” and “**Utip**” (each card shows a Label and the Project it belongs to, such as “Cypress Test Default”).

The trash icon (for deleting a platform) and the pencil icon (for editing a platform) are visible only if the corresponding user permissions—(platforms.delete) for delete and (platforms.edit) for edit—are set to true.

Platform Details

Each platform has fields for Name, Label, URLs, and Credentials.

The example shows Utip with multiple URL fields (like tradingUrl, accountUrl) and credential fields (tradingSecretKey, accountSecretKey, etc.).

Linking Assets to a Platform

Navigate to “Accounts”: In the **Core Banking MS** menu, go to **Accounts**.

Add or Edit an Asset:

Within an account, you can **Add asset** (e.g., a USDT asset).

A dropdown labeled “**Platform**” appears.

Select “**Utip**” (or whichever platform you created).

Result: Any balance or transaction related to this asset is now stored on the external platform instead of the CRM’s internal database. When you perform a deposit or withdrawal, our platform communicates with the platform via API to update balances.

Why Use External Platforms?

1. External Balance Storage

Internal (no platform): If you create an asset without selecting a platform, its balance lives solely inside the CRM’s database.

External (with platform): When you link an asset to an external platform (e.g. Utip), all balances and holdings are stored on that external system, and the CRM merely mirrors them.

2. Real-Time Transactions

All deposits, withdrawals or trades are executed via the platform’s API.

The CRM only updates its view of the balance once the external system confirms the change, ensuring accuracy and consistency.

3. Flexibility & Scalability

Multiple integrations: You can configure as many platforms as needed (e.g. Utip, Binance, etc.).

New integrations on demand: If your business adopts a new processor, just discuss the requirements and we can add it.

Example Workflow

A. Create a New Platform (Utip)

1) Navigate to Platforms: Go to **Settings** → **Core Banking** → **Platforms**.

2) Click Add: Opens the “Add platform” panel.

3) Configure the fields:

Project: Select your project (e.g.).

Name: Enter .

Label: Enter .

API URLs & Credentials: Paste the endpoint URLs (REST/WebSocket) and your API key/secret.

4) Save

Result: Utip is now registered as an external platform for all future transactions.

B. Add a USDT Asset

1) Go to Accounts: In the CRM sidebar, select **Accounts**.

2) Click Add asset

3) Fill in asset details:

Currency:

Type:

Platform: Choose **Utip** from the dropdown.

4) Save

Result: Your USDT asset is linked to Utip — all USDT movements will route through Utip’s API.

C. Perform a Deposit

1) Initiate deposit in CRM: User clicks “Deposit” on the USDT asset.

2) CRM sends API request: The CRM calls Utip’s endpoint with amount and account info.

3) Utip processes & confirms: Utip updates its own ledger and returns the new balance.

4) CRM reflects balance: The CRM refreshes the USDT balance to match Utip’s confirmed value.

Key Points

- 1. Project-Based:** Every platform is scoped to a specific project.
- 2. Credentials & URLs:** Must be configured correctly for seamless API communication.
- 3. External Balances:** Any asset tied to a platform is managed off-platform; the CRM simply displays what the external system reports.
- 4. API-Driven Transactions:** Deposits, withdrawals, and other operations always go through the platform’s API rather than internal database updates.

In summary, the Core Banking Platforms function allows you to add, edit, and manage integrations like Utip so that asset balances and transactions occur on an external platform rather than within the CRM itself. By configuring Name, Label, URLs, and API credentials, you enable real-time interaction between your chosen platforms, ensuring seamless and secure transaction management.

7. How to Manage Accounts

The Accounts module is designed to manage client accounts and is closely integrated with the [Clients](#) module. While accounts are visible from client profiles, all core account management actions are performed in the Accounts module.

In this module, you can:

1. View all accounts in the system (*for clients visible to you*).
2. Add new accounts.
3. Activate and deactivate accounts.
4. Filter and search for accounts.
5. Edit account properties and manage related assets and wallets.

You'll land on a table showing all accounts you have permission to view.

Columns:

Status — Toggle switch (Active / Inactive)

Account ID — System-generated UUID

UID — System-generated unique identifier

Type — Account type defined in system configuration

External ID — Integrator's reference (if any)

Project — Linked project

Owner — Client name (clickable). The owner of an account cannot be changed after the account is created.

Created — Timestamp

Actions — Context actions such as adding assets, editing the account, deposits, and withdrawals

When you enter the **Accounts** module, you'll see a list of all accounts. To view details about the client who owns an account, click on the name in the **Owner** column of the corresponding row:

To view client details, click the client name in the **Owner** column. This opens the client's profile.

To add a new account:

1. Click the **Add Account** button in the upper-right corner of the **Accounts** tab.

2. Enter or select the following:

Type: The type of account (e.g., Business, Personal, Crypto, Fiat). If no types are available, go to

Core Banking > Configurations to create them.

Owner: The client to whom the account belongs. If no clients are listed, go to the **Clients** module to create one.

External ID: *(Optional)* An identifier for linking the account to a third-party platform via API.

3. Click **Save**. The system generates a unique **Account ID** (UUID).

Note: A unique ID is automatically assigned to each account upon creation. Account status can be managed using the Status toggle after creation.

To activate or deactivate an account:

To activate an account, toggle the switch in the **Status** column of the corresponding row. The switch will turn blue.

Blue (on) = Active

Gray (off) = Inactive

To deactivate the account, toggle the switch again.

To search for an account:

Type the **Account ID** in the Search field located above the Accounts table.

Press **Enter** or click the magnifier to filter the list in real time.

To filter accounts:

The Search field operates independently from structured filters. Search terms are not stored in saved filter views.

1. Click **Filter** next to the search field:

2. Select one or more parameters for filtering:

Created Date: The period during which the account was created. If you know a specific date, set it as the **End Date** and select a **Start Date** before that.

Type: The account type, based on the types you've set in **Configurations**.

Owner: The specific client, searchable by email.

Active or **Inactive:** Filter by account status using the checkbox.

To delete a specific filter, hover over the field and click the x icon.

3. Click **Save** to apply the filters.

To edit the account, click the **Edit** icon in the corresponding row of the account.

In the **Edit Account** tab, you can:

Change the account's **Type**.

Enter or update the **External ID**.

Note: You cannot change the owner of the account.

To view and manage account assets:

1. Click the **Expand** icon in the corresponding account row to view all assets associated with the account.
2. If you need to add an asset, click the **Add Asset** icon.

To add an asset:

1. In the **Add asset** window, select:

Currency: Once selected, its type (fiat or crypto) will be automatically detected.

Balance: The initial balance of the asset.

(For fiat currencies) **Address**, **External ID**, and **Bank** (optional fields). If you are doing integration with a bank, you do not need to specify an address. If there is no integration with the bank, you should specify IBAN in the **Address** field.

Platform. Non-mandatory field. It is only filled in when integrating with third-party platforms. Read more about platforms in [\[How to Set Platforms\]](#).

2. Click **Save** to add the asset.

Note: Once added, the asset will need to be activated. Activation can be done in the [Assets module](#).

For cryptocurrencies with several different networks, you can also add wallets.

How to add a wallet:

1. Click the **Expand** icon in the corresponding account row to view all assets associated with the account.
2. Click on the asset. If the asset is clickable, it will be highlighted in blue.
3. Click the **Add** button in the upper-right corner.
4. Enter or select from the drop-down list **Network** and **Address**.
5. Click **Save** to apply changes.

You can search created wallets by address, using the **Search** field.

You can also **change address of the wallet**. To do that:

1. Click the **Edit** icon in the corresponding wallet row.
2. Enter the new address.
3. Click **Save** to apply the changes.

Note: You cannot change the network of the wallet.

Account Deletion (Permission-Based with Financial Safety Validation)

In Core Banking, account deletion is strictly controlled to protect financial integrity, audit history, and ledger consistency.

Deletion is permanent and cannot be undone.

Permission Requirement

To delete an account, a role must have:

Accounts → Delete

Location:

Roles → Edit Role → Accounts section → Delete

Behavior:

1. If the role does not have this permission, the Delete action is not visible in the UI.
2. Any direct API/backend attempt without permission returns **403 Forbidden**.

Where Delete Appears

If permission is granted:

1. A Delete (trash) icon appears in the Actions column for each account.
2. The action follows the standard confirmation modal pattern.

On click:

A confirmation modal appears:

Title: Delete Account

Body: This action cannot be undone.

Buttons:

Confirm

Cancel

If confirmed, the backend performs eligibility validation before deletion.

Backend Eligibility Rules (Mandatory)

An account can be deleted only if ALL conditions below are met:

1. All assets under the account have:
 - No linked transactions
 - Available balance = 0
 - Bonus balance = 0
 - On hold balance = 0
2. The account itself has no additional system constraints (if applicable).

If any asset under the account contains transactions or non-zero balances:

Deletion is rejected.

Example error responses:

1. Account cannot be deleted: it contains assets with transactions
2. Account cannot be deleted: it contains assets with balance

Status codes:

1. 403 — missing permission
2. 409 — state conflict (financial data exists)

Important: Eligibility is validated server-side at the moment of deletion to prevent race conditions.

After Successful Deletion

If the account meets all eligibility conditions:

1. The account is permanently removed from the system.
2. It disappears from the Accounts table immediately.
3. A success notification is displayed.

No orphan assets or ledger inconsistencies remain.

Why This Restriction Exists

Accounts serve as containers for financial assets.

If assets with balances or transaction history exist, deleting the account would break:

1. Financial reporting
2. Audit trails
3. Reconciliation workflows
4. Compliance obligations

Therefore, accounts with financial history are immutable.

If operational removal is required, the recommended approach is:

Deactivate the account instead of deleting it.

Extra Information

Bulk Import/Export: Available via the CRM API for large-scale onboarding.

API Endpoints:

1. GET /accounts
2. POST /accounts
3. PATCH /accounts/{id}

Pagination & Performance: The UI loads 50 accounts per page; use filters/search for quicker lookup.

Audit Logs: Changes to accounts, assets, and wallets are logged under **System** → **Audit Trails**.

Troubleshooting:

If you cannot see an account, verify your **role permissions**.

“Cannot save” errors often mean a required field is empty or mis-typed.

Wallet creation failures may indicate an unsupported network—ensure network names match those in **Configurations**.

8. How to Manage Assets

The **Assets** module provides centralized management of all assets in the system. It allows you to:

1. View a list of all assets (*for visible clients*) and check their balances.
2. Activate and deactivate assets.
3. Filter and search for specific assets.
4. View transactions for assets.
5. Make deposits and withdrawals.

Note: If you want to add an asset to an account, you need to do it through the [Accounts](#) module.

Use Cases

1. **Segregation of Funds**
Real money and promotional credits remain clearly separated, ensuring transparent financial tracking.
2. **Flexible Promotions**
Companies can create custom bonus programs—such as first deposit bonuses, loyalty rewards, or marketing campaign incentives.
3. **Enhanced User Engagement**
Users can explore platform features with bonus funds before committing their real money, increasing retention and activity.

View All Assets

To see every asset in the system, navigate to the **Assets** module from the sidebar. At the top of the page you'll find:

Shareable Link (next to the "Assets" title) - copy a URL that preserves your current filters and sort.

Total Count - updates dynamically as you filter or paginate.

Language / Calendar / Notifications / User Menu in the top-right corner.

The Assets module displays a read-only table of every asset in your system, with the following columns:

1. **Status** - Toggle to activate (blue) or deactivate (gray) this asset.
2. **Currency** - Asset's currency icon and code (e.g. BTC, USD).
3. **UID** - System-generated, auto-incrementing asset code (e.g. cbas00002d).
4. **Owner's full name** - Client who owns the account (clickable).

5. **Owner's email** – Email address of the client who owns the account
6. **Available** – User's spendable balance.
7. **All time deposited** – Cumulative funds ever deposited.
8. **On hold** – Funds locked by pending transactions or holds.
9. **Actions** – Icons for Show Transactions, Deposit, Withdraw, Refresh Balance, and a "..." menu for Show Wallets.

Status:

1. Toggle to **activate** (blue) or **deactivate** (gray) an asset for that account.
2. If there are transactions, a small **chevron** appears here—click to expand and view them.
3. *Deactivated assets are hidden from end-user views and cannot be transacted until reactivated.*

Column Sorting: Click any header's ▲/▼ arrows to sort by that field (e.g. Available, Bonus). Clear by clicking again.

Search for an Asset

Enter the ID of the account in the Search field located above the Assets table to filter rows in real time. You can search by:

1. **Account ID** (full or partial UUID)
2. **Asset ID** (UUID)
3. **Currency code** (e.g. "USD", "BTC")
4. **Owner's name** (first/last) or **email**

The search is case-insensitive and matches substrings as you type.

“ The Search field operates independently from Fast Filters and does not become part of saved filter views.

Filter Assets

The **Assets** module uses the **Fast Filters** panel, located directly above the assets table. This interface allows you to instantly narrow down results by project, type, currency, ownership, and balance parameters without opening a separate drawer.

The panel includes the following elements:

Project – filters assets belonging to a specific project or environment.

Currency – filters by asset currency (e.g., USD, EUR, USDT, BTC).

Types – allows you to choose between asset categories such as *Fiat*, *Crypto*, *Bonus*, etc.

Owners – filters by user or client who owns the asset.

Available – numeric filter showing only assets with balances below or above a given amount

(using *Less Than / Greater Than* selectors).

All time deposited – filters based on total deposit volume recorded for the asset.

On hold – filters assets currently locked or pending withdrawal.

Bonus – filters by available bonus balance, using numeric range options.

Show only – displays only specific subsets such as *Active, Disabled, or Zero Balance* assets.

Apply – applies all chosen filters and refreshes the asset table.

Drop filters – clears all active filters in one click.

Create view – saves the current filter combination and table layout for future access.

Apply Your Filters:

1. In the **Assets** module, locate the **Fast Filters** bar above the table.
2. Enter a keyword or asset name in the **Search** field if needed.
3. Use dropdown filters to refine results:
 - Project** → choose the project the assets belong to.
 - Currency** → filter by specific currencies.
 - Types** → select *Fiat* or *Crypto* to limit the view.
 - Owners** → pick one or more client names or email addresses.
4. Adjust numeric fields if needed: **Available, All time deposited, On hold, and Bonus** fields accept numeric ranges (e.g., “Less Than 1000”).
5. Click **Apply** to update the table and view matching records.

Clear or Adjust Filters:

1. **Modify a field:** Change any value (e.g., select a new currency) and click **Apply** again.
2. **Remove individual filters:** Click the “x” next to any selected filter to remove it, then **Apply**.
3. **Reset all filters:** Click **Drop filters** on the right to clear all fields and show all assets again.
4. **Save a view:** To reuse the same filter combination later, click **Create view**.

Quick Search vs. Filter:

Use the **Search...** field at the top for quick lookups by keyword, currency, or owner name.

Use **Fast Filters** for structured queries combining multiple parameters such as project, asset type, currency, and numeric conditions (e.g., available balance or total deposited).

Actions Column

In the **Actions** column, you can manage the selected asset:

1. Show transactions:

Click the **Show Transactions** icon to view all transactions for the asset.

In the opened window, you can:

View the logs of any transaction from inception to completion (who took what actions, how and when) by clicking the **Show Transaction History** icon in the **Actions** column.

Search transactions by **Transaction ID, Asset ID, or Account ID**.

Filter transactions by **Owner**, **Created Date**, **Account Type**, **Status**, **Transaction Type**, and **Sub-Type**.

Export transactions in CSV format by clicking the CSV icon in the upper-right corner.

Note: For more detailed transaction management, refer to the [Transactions module](#).

2. Make deposit:

Click the **Deposit** (📄) icon.

In the window that opens, you will see the current balance and fields to add an amount. Enter:

Amount: Deposit amount.

Description: Any text description. For example, the purpose of the transaction, etc.

Source: The source where the money came from. Can be a general description (e.g. “company”) or specific details such as IBAN or wallet.

Click **Save** to apply the changes. When you save, a transaction will automatically be created.

3. Make withdrawal:

Click the **Withdrawal** (📄) icon.

In the window that opens, you will see the current balance and fields to withdraw an amount. Enter:

Amount: Withdrawal amount.

Description: Any text description. For example, the purpose of the transaction, etc.

Destination: The destination where the money went. Can be a general description (e.g. “company”) or specific details such as IBAN or wallet.

Click **Save** to apply the changes. When you save, a transaction will automatically be created.

“ Account Balances (Platform-Backed)

By default, any account linked to an external Platform will pull its **Available** balance directly from that platform whenever you see balances in the Accounts module.

Note: You can learn more about platform customization in the [Platforms section](#).

The functionalities for **Show Transactions**, **Show Wallets**, **Deposit**, and **Withdrawal** are also available within the **Accounts** module. To access these, click the triple-dot icon in the corresponding asset row.

Assets with Available and Bonus Balances

In Core Banking, each asset in a user’s wallet can have two separate balances:

1. Available Balance

Represents the user's real, spendable funds.

Deposits or withdrawals of actual money (e.g., via a payment system or direct bank transfer) increase or decrease this amount.

2. Bonus Balance

Represents promotional or virtual credit, often used to let users try features or test the platform. Funds may be non-withdrawable under typical conditions—users might need to meet additional requirements (e.g., reaching a specific trading volume or making a minimum deposit) before withdrawal is possible.

Note: Our system does not regulate this functionality; its usage is entirely up to you.

You can see these two fields on the Assets page for each currency (e.g., USD, EUR, USDT). There's a column for Available and another column for Bonus.

Example: Awarding a Bonus

1. Standard Deposit

Initial state: Available = **9 900**, Bonus = **0**

User deposits **100 EUR** via payment system.

Result:

Available → **10 000** (9 900 + 100)

Bonus remains **0**

2. Bonus Deposit

User makes a **200 EUR** deposit and qualifies for "Deposit 200 → +200 EUR bonus."

The real deposit adds to Available; the promotional credit goes into Bonus.

Result:

Available → **10 200** (10 000 + 200)

Bonus → **200**

Note: "All time deposited" now reads 300 (100 + 200), while "On hold" remains unchanged.

Core Banking Assets Bonus functionality distinguishes real, withdrawable funds (Available balance) from promotional or test credits (Bonus balance). This structure gives businesses the flexibility to offer promotions, demo credits, or loyalty rewards while maintaining separate tracking, ensuring clarity for both the company and the end user.

Asset Deletion (Permission-Based with Financial Safety Validation)

In Core Banking, asset deletion is strictly controlled to protect financial integrity and historical data. Deletion is not a cosmetic action — it is permanently destructive and cannot be undone.

Permission Requirement

To delete an asset, a user role must have:

Assets → Delete

Location:

Roles → Edit Role → Assets section → Delete

If the role does not have this permission:

1. The Delete icon will not appear in the Assets table.
2. Any direct backend attempt will return **403 Forbidden**.

Where Delete Appears

If permission is granted:

1. A Delete (trash) icon appears in the Actions column for each asset.
2. The action follows the standard confirmation modal pattern.

On click:

A confirmation modal appears:

Title: Delete Asset

Body: This action cannot be undone.

Buttons:

Confirm

Cancel

Backend Eligibility Rules (Mandatory)

An asset can be deleted only if ALL conditions below are met:

1. Asset has no linked transactions (historical or pending).
2. Available balance = 0.
3. Bonus balance = 0.
4. On hold balance = 0.

If any of the above conditions fail:

Deletion is rejected by the backend.

Example error responses:

1. Asset cannot be deleted: it has transactions

2. Asset cannot be deleted: non-zero balance

Status codes:

1. 403 — missing permission
2. 409 — state conflict (balance or transactions exist)

Important: Validation is executed server-side at the moment of deletion to prevent race conditions.

After Successful Deletion

If the asset is eligible:

1. The record is permanently removed from the database.
2. It disappears from the Assets table immediately.
3. A success notification is shown.

No orphan ledger entries remain.

Why This Restriction Exists

Assets are part of the financial ledger.

If transactions or balances exist, deletion would break:

1. Historical financial reporting
2. Audit trails
3. Reconciliation processes
4. Compliance requirements

Therefore, assets with financial history are immutable and cannot be deleted.

If removal is required for operational reasons, the correct approach is:

Deactivate the asset instead of deleting it.

9. How to Manage Transactions

The **Transactions** section is designed to view and monitor all transactions that were created in CRM via other modules or automatically. You can do the following:

1. Select the information that will be displayed about the transaction.
2. Change the transaction status.
3. Edit the transaction description (permission-based).
4. Search and filter transactions.
5. View the transaction history.
6. Download a list of transactions in .csv format.

By default, the transaction list displays all transaction information, from basic as **Type** (In or Out) to advanced as assigned **Desk**.

The Transactions module shows a table of every transaction in your system, with the following columns (possibly to add more):

Amount - Positive (green) for inbound, negative (red) for outbound.

Type - Direction icon (← for withdrawal, → for deposit).

Project - Linked project (clickable).

Account type - The type of account (e.g. Demo, Personal).

ID - Internal UUID of the transaction.

UID - Auto-incrementing transaction code (e.g. cbtr00002q).

Asset ID - Clickable link to the asset record.

Account ID - Clickable link to the account.

Source email - Remitter's email (if available).

Source - Origin of funds (e.g. External, deposit-source).

Beneficiary - Destination tag or account/crypto address.

Status - Transaction state (Completed, Processing, etc.).

Actions - View full transaction history or edit the transaction (permission-based).

To select which transaction information to display:

1. Open the **Filters** window and click **Customize Columns** at the top of the **Transactions** tab.
2. Check the boxes next to the fields you want to be displayed.
3. Click **Save** to select columns.

You can **view additional information on the field values** that are highlighted in blue. To do this:

1. Click on the field value in the **Currency** column.
2. A drawer with information about the field opens.

You can change all transaction statuses, except **Completed** and **Canceled**. Accordingly, you can approve or cancel transactions in **Processing**, **Pending**, and other custom statuses.

To change the status:

1. Click on the status in the corresponding row.
2. Select a new status from the dropdown. Custom statuses (e.g., "Review," "Escalated") appear here too.

Warning: After selecting the **Completed** or **Canceled** transaction status, the transaction history ends. It is not possible to cancel this action.

Edit Transaction Description

If your role includes the appropriate permission, you can edit the transaction description.

To edit a transaction:

1. Click the Edit icon in the Actions column.
2. A right-side drawer opens.
3. Modify the Description field.
4. Click Save to apply changes.

Notes:

1. The Description field is pre-filled with the existing value.
2. Only users with the required permission can see the Edit action.
3. All edits are logged in the transaction history.
4. Core financial data (amount, currency, IDs) cannot be modified.

Transaction Visibility & Role-Based Access

Transaction visibility is strictly controlled by role permissions and follows the same access logic used in the Clients, Projects, and Desks modules.

Access is determined at backend level and cannot be bypassed by frontend filtering.

Transaction visibility follows these rules:

Clients permissions

1. If a user has **View all clients**, they can see transactions for all clients.
2. If a user has **View own clients**, they can only see transactions for clients assigned to them.

Projects permissions

1. If a user has **View all projects**, they can see transactions across all projects.
2. If a user has **View own projects**, transactions are limited to projects they are assigned to.

Desks permissions

1. If a user has **View all desks**, transactions from all desks are visible.
2. If a user has **View own desks**, only transactions from assigned desks are shown.

Combined Permissions Behavior

When permissions are mixed, visibility is calculated as the intersection of allowed scopes.

Example:

View all projects + View own clients

→ User sees transactions only for their own clients across all projects.

Users without “View all” permissions will never see unrelated transaction data.

This behavior ensures:

1. Full compliance with role-based access rules
2. No overexposure of financial data
3. Consistency with other modules

To search for a transaction: Type into the **Search** field located above the Transactions table to live-filter by any visible text field (e.g., ID, asset code, remitter email, amount). Matches substrings, case-insensitive.

“ The Search field works independently from Fast Filters and is not stored in saved filter configurations.

Manager Snapshot Persistence

Each transaction permanently stores the manager who was assigned to the client at the moment the transaction was created or updated.

This ensures historical accountability and prevents retroactive changes if the client’s manager is reassigned later.

When the Manager Snapshot Is Applied

The manager snapshot logic runs automatically in transaction write operations:

1. On transaction creation
2. On save

3. On findOneAndUpdate
4. On updateOne
5. On any other transaction write path

Bulk update operations (updateMany) must explicitly support this logic if used in the codebase.

Important Behavior

1. If the client's manager changes later, previously created transactions **retain the original manager value**.
2. If a transaction is updated later, the manager field is refreshed from the current client manager.
3. If the client does not exist, the system does not overwrite the existing manager value.
4. If the client exists but has no assigned manager:
 1. The system sets manager to null (if following null convention),
 2. or preserves existing value (based on platform-wide consistency rules).

Performance & Integrity

1. Client lookup uses minimal projection: { manager: 1 }
2. No recursive writes are triggered.
3. No historical transaction records are mutated by later client edits.
4. The manager snapshot is immutable unless the transaction itself is updated.

To filter transactions:

1. Click the **Filter** button.
2. Enter or select from the drop-down list one or more parameters for filtering. Main filters include:

| Field | Type |
|---------------|---|
| Project | dropdown (available projects) |
| Desks | dropdown (available desks inside the project) |
| Owner Manager | dropdown (name) |
| Owners | dropdown (name) |
| Account | free-text UUID |
| Asset ID | free-text UUID |
| Currency | dropdown (e.g., CAD, EUR) |
| Owner | dropdown (client name/email) |
| Created Date | date-range picker |
| Account Type | dropdown (Personal, Demo...) |
| Sub-type | dropdown (from integrations) |

| Field | Type |
|-------------------|-------------------------|
| Status | dropdown (all statuses) |
| Type | dropdown (IN / OUT) |
| Operation Manager | free-text |

3. Click **Apply** to apply filters.

You can also save selected filters and quickly reuse them the next time you open the Transactions table. To do this:

- Select the required filters
- Click Create view
- Enter the name
- Click Submit

To clear a single filter, hover its field and click ×; to reset all, close and reopen the panel.

To view the transaction history:

1. Click the **Show transaction history** icon in the **Actions** column in the corresponding row.
2. You will see the logs of the transaction from inception to completion (who took what actions, how and when).

A side-drawer lists each state change and user action in chronological order.

“ **Tip:** You can also jump directly to history from the **Transactions** module’s own drawer (see next section).

Inspect Full Transaction Details

Click any **Transaction ID** cell to open the **Transaction Information** drawer:

General (Currency, Amount, Type IN/OUT, Provider, IDs, Dates, Description)

Owner (Client name, email, profile link)

Source (Remitter details, External ID, Order ID, Hash, Fees)

Destination (Beneficiary, Destination address or account)

Use the ☰ chevrons to expand/collapse each section.

To export the transactions list in .csv format: Click the **CSV** icon in the upper-right of the Transactions tab to download every row currently visible (respecting your filters, columns, and pagination) as a comma-separated file.

By mastering **Columns**, **Search**, **Filter**, **Status**, and **Drawer** controls, you can tailor the Transactions view to any reconciliation, audit, or support workflow—then export or update statuses in just a few clicks.

10. How to Manage Banks

The **Banks** section stores information about the details of the banks in use. This section is directly related to [Assets](#). At the stage of adding an asset (provided that it is a fiat currency), you can (optionally) select the bank to which it belongs from a given list.

For example, you have a platform where a user can create an EUR asset with its own IBAN and select a banking system for it. You can use this functionality to identify which bank this asset's IBAN is from. Then you can divide assets across different banks, create integrations, and understand which bank and its credentials need to be queried from CRM and other systems.

Accessing the Banks Module

In the left-hand **Core Banking** sidebar, click the **Banks**.

The header shows “Banks” with:

A **Total** count of bank records.

A **Search** field to quickly locate banks by any visible column.

Add a bank button on the upper-right.

To Add a Bank:

1. Go to the **Banks** section.
2. Click the **Add Bank** button in the upper right corner of the tab.
3. Fill in the following fields:

Name (Required): Full legal name of the bank.

Address: Street address or headquarters location.

Country (Required): Select from the dropdown of country names/flags.

Bank code (e.g., routing number).

Licence: Regulatory licence number, if applicable.

SWIFT/BIC: International bank code.

IBAN: Default IBAN format for this bank (used when auto-populating assets).

Make as default (checkbox): If checked, this bank is automatically pre-selected when adding new fiat assets (you can only have one default at a time).

4. You can also make this bank the default bank. To do this, click the **Make as default** checkbox.

5. Optionally, fill in the **Rules** and **Credentials** fields for additional settings:

Rules: Custom rules. For example, you can define how to use a given bank on the platform, with what conditions.

Credentials: Developer field. Here you can save the bank's credentials for integration.

6. Click **Save** to apply the changes.

You can do following actions with all created banks:

1. Searching
2. Editing
3. Deletion

To Search For a Bank:

Search: Begin typing any part of a bank's Name, IBAN, SWIFT/BIC, or Country ISO code. The list filters in real time.

Sorting: Click the ▲/▼ icon next to any column header (e.g., **Bank**, **Country**, **Licence**) to sort ascending/descending.

Note: Banks cannot be “filtered” by advanced criteria—use the live Search and column sorting to narrow the list.

To Edit a Bank:

1. In the **Actions** column of the bank's row, click the **Edit** (✎) icon.
2. The same drawer appears, pre-populated with that bank's details. Update any editable field—Name, Codes, Rules, Credentials, etc.
3. Click **Save** to apply changes.

“ If you change **Make as default**, the previous default bank is automatically unset.

To Delete a Bank:

Click the **Delete** (🗑) icon in the **Actions** column.
Confirm the prompt.

Warning: Deletion is permanent. Any assets previously linked to this bank will lose their bank association (you'll need to re-assign them manually).

Linking Banks to Assets

When you **Add Asset** (fiat) in the **Accounts** or **Assets** modules, the **Bank** dropdown lists every bank you've created—default first, then alphabetically. Selecting the correct bank ensures:

1. **Automated IBAN validation** (based on bank code).
2. **Platform integrations** (via the bank's stored credentials).
3. **Accurate reconciliation** when querying external banking APIs.

Quick Tips

1. **Default Bank:** Always ensure you have one default bank for your most-commonly used fiat currency.
2. **Rules & Credentials:** Use these JSON editors to configure country-specific routing rules or to store sandbox/live API keys.
3. **Audit Trail:** All create/edit/delete actions appear in your system's audit logs—perfect for compliance reviews.

With this setup, your platform can precisely route fiat deposits and withdrawals, segment assets by banking partner, and leverage integrations seamlessly.

11. How to Manage Cards

The **Cards** section stores information about the details of the cards in use. This section is directly related to [Accounts](#), [Assets](#), and [Banks](#). You can create a card for a client and connect it through the account to a fiat and crypto asset at the same time (or to one of that asset). All this is done when adding a new card to the CRM.

Accessing the Cards Module

Click the **Cards** icon (☰) in the left-hand sidebar.

The header shows:

Total: Number of cards in the system.

Search by ID... field (to filter by card ID or masked number).

Add card button in the upper-right.

Searching & Sorting Cards:

1. **Search:** Type any portion of the **Card** (e.g. “4567”) or **ID** to narrow the list in real time.
2. **Sort Columns:** Click the ▲/▼ icons in any column header—such as **Owner**, **Bank**, or **Priority**—to reorder ascending/descending.

To Add a Card:

1. Click the **Add card** button in the upper right corner of the tab.
2. Enter **Card number**.
3. Then step-by-step add:
 - Owner of the card:** from the list of Clients.
 - Account ID:** from the list of the client’s accounts.
 - Crypto asset ID** and/or **Fiat asset ID:** from the lists of the account’s assets.
4. In case you select both crypto and fiat asset, select the priority for the asset.
5. Select from the drop-down list Bank.
6. Optionally enter **External ID** for third-party integration.
7. Click **Save** to apply the changes.

You can do following actions with all created cards:

1. Searching
2. Editing
3. Deletion

To search for a card: Start typing its **ID** in the **Search** field.

To Edit a Card:

1. Click the **Edit** icon in the corresponding row.
2. All field in the opened window, except the **Owner**, are changeable. ange account links, asset assignments, bank selection, or external IDs as needed.
3. Click **Save** to apply changes

To Delete a Card:

Click the **Delete** (🗑️) icon in the card's Actions column.
Confirm in the prompt.

Warning: Deleted cards are not restored.

Quick Tips

1. **Brand Detection:** The first six digits of the card number auto-select the correct brand icon (Visa, Mastercard, etc.).
2. **Priority Logic:** Use Priority to ensure, for example, that expenses draw from a user's crypto wallet before their fiat balance.
3. **Integration IDs:** Store external gateway IDs in **External ID** to reconcile events from payment processors.

With the Cards module, you maintain a clear, auditable link between physical/virtual cards and the underlying client accounts, assets, and banking relationships.

12. Configurations

The Configurations module in Core Banking allows managing project-specific settings. Unlike a global configuration affecting all projects simultaneously, this system ensures that each project has its own independent settings. This modular approach enables better organization, customization, and scalability.

Use Cases

#1. Switching Between Projects

Users can quickly switch between projects, with the system displaying configurations specific to the selected project. If no settings are defined, the section shows "No data to display" until configurations are added.

#2. Cloning Configurations for New Projects

When creating a new project, users can use the "Clone from project" feature to copy configurations (e.g., currencies, transaction sub-types, or account types) from an existing project, saving time and ensuring consistency.

#3. Cloning Currency Configurations

Users cloning currency settings (e.g., USD, EUR, USDT, TRX) from one project (e.g., Wibroker) to another (e.g., Incur) avoid manual currency additions, enabling faster project setup.

Accessing the Configurations Module

In the left sidebar, expand **Core Banking MS** and click **Configurations**.

Each project can have specific settings. Once inside, you'll see four panels:

| Panel | Purpose |
|---------------------------------|---|
| Types of account | Define labels for different account categories (e.g. Personal, Business). |
| Currencies | Pick which currencies the project supports (e.g. USD, EUR, USDT). |
| Sub-types of transaction | Add custom transaction sub-types (e.g. Deposit, Fee). |
| Transaction statuses | List possible statuses (e.g. Pending, Completed, Canceled). |

This modularity helps businesses handle multiple projects with ease, allowing them to customize financial settings for each project separately.

1. Switching Between Projects

At the top is **Settings by project**:

Click the project pill or type in the search field to pick a project.

All four panels update to show that project's configurations.

Each project can have unique configurations. When selecting a project, the system updates the displayed settings specific to that project.

Note: If a project doesn't have any settings defined yet, it will show "No data to display" for its sections.

2. Cloning Configurations from Another Project

To make configuration setup easier, the system includes a Clone from Project functionality.

How It Works

The user selects a section, such as **Currencies**, **Sub-types of Transactions**, or **Types of Accounts**.

“ Set a Default Account Type:

In the **Types of account** panel only, each row has a **Make as default** link (or a "Default" badge if already set). Clicking **Make as default** marks that type as the project's default—only one type can be default at a time, and the UI will highlight it accordingly.

Click "**Clone from project**" in that section.

A popup appears where the user selects a project from which they want to clone data.

Clicking "Clone" copies all configurations from the selected project to the current project.

Note: If data has already been cloned (or created manually), the Clone button disappears to prevent duplication.

Cloning Example: Currencies

For example, the user clones currency configurations from Wibroker into the Incur project.

“ Once cloned, the system automatically displays the available currencies (e.g., USD, EUR, USDT, TRX).

This prevents the need to add each currency for new projects manually.

Editing & Deleting Entries

Edit: Click ⇨ next to an item, update its fields, then **Save**.

Delete: Click ☒ and confirm.

Warning: Deletions are permanent and cannot be undone.

Cloning Other Configurations

The same cloning process applies to Sub-types of Transactions and Types of Accounts. For instance, the user clones Sub-types of transaction from Cypress Test Default into Incur. Once cloned, the configurations instantly appear under the selected project. This saves time and ensures consistency across projects while allowing each one to operate independently.

Best Practices & Tips

1. **Clone all panels** when launching a new project to maintain consistency.
2. **Customize statuses** per project (e.g. “Held,” “Reconciled”).
3. Use **clear naming conventions** for account types (e.g. “Demo,” “Live”).
4. Regularly **review** and **prune** unused currencies or statuses to keep the UI clean.

This project-based approach ensures each environment has tailored financial settings, speeds up onboarding of new projects, and prevents cross-project configuration conflicts.

13. Invoices

The **Invoices** section is designed to create, track, and manage invoices issued to clients. It allows you to monitor payment status, associate invoices with transactions, attach supporting documents, and follow the full lifecycle of each invoice from creation to settlement or expiration.

In this section, you can:

1. Create new invoices for clients
2. View and manage all existing invoices
3. Change invoice statuses
4. Search and filter invoices by multiple criteria
5. Attach and manage invoice documents
6. View related transactions and postbacks

By default, the invoice list displays all invoices you have permission to view, including their current status, client, amount, and key dates.

Invoice List and Columns

The Invoices module displays a table of all invoices in the system with the following columns:

1. **Currency** - Invoice currency (with asset icon if applicable)
2. **Amount** - Invoice amount
3. **Project** - Linked project (clickable)
4. **Client** - Client associated with the invoice (clickable)
5. **External ID** - External or integration identifier (if provided)
6. **Description** - Invoice description (if provided)
7. **Expires At** - Invoice expiration date and time
8. **Created at** - Date and time the invoice was created
9. **Status** - Current invoice status (Pending, Paid, Overpaid, Underpaid, Expired)
10. **Actions** - Context actions such as documents, edit, and related records

Columns can be sorted by clicking on the column headers.

Creating a New Invoice

To create a new invoice:

Click the **Add** button in the upper-right corner of the Invoices section.

In the **Add invoice** window, fill in the following fields:

Client - Select the client the invoice is issued to

Currency - Select the invoice currency

Amount - Enter the invoice amount

Network - Optional field for blockchain or payment network

Wallet address - Destination wallet address (if applicable)

External ID - Optional identifier for external systems

Expires At - Expiration date of the invoice

Status - Initial invoice status

Comment - Internal comment

Description - Invoice description or notes

Click **Save** to create the invoice.

The newly created invoice will appear in the invoice list with the selected status.

Editing an Invoice

To edit an existing invoice:

1. Locate the invoice in the list.
2. Click the **Edit** (✎) icon in the **Actions** column.
3. Update any editable fields such as amount, expiration date, status, comment, or description.
4. Click **Save** to apply changes.

Invoice Status Management

Invoices support multiple statuses, including **Pending**, **Paid**, **Overpaid**, **Underpaid**, and **Expired**.

To change the status:

1. Click the **Status** badge in the corresponding invoice row.
2. Select a new status from the dropdown list.

The status updates immediately and is reflected in the table.

Searching for Invoices

Use the **Search** field located above the Invoices table to quickly find invoices by visible text fields such as client name, currency, or identifiers. The list updates in real time as you type.

“ The Search input is independent from Fast Filters. Saved filter views include only structured filters and do not store search terms.

Filtering Invoices

To filter invoices:

Click **Select any filter** to open the filter panel.

Configure one or more of the available filters:

1. **Project** – Filter invoices by project
2. **Clients** – Filter by one or more clients
3. **Currency** – Filter by invoice currency
4. **Amount** – Filter by amount (e.g., less than a value)
5. **Status** – Filter by invoice status
6. **Created date** – Filter invoices created within a date range
7. **Wallet address** – Filter by wallet address (contains)
8. **Comment** – Filter by comment text (contains)

Click **Apply** to apply the selected filters.

To remove all filters at once, click **Drop filters**.

You can also save a filter configuration using **Create view** for reuse.

Invoice Documents

Each invoice can have documents attached.

To manage documents:

Click the **Documents** icon in the **Actions** column.

In the **Documents** window, drag and drop a file or click **Browse** to upload it.

Supported formats include pdf, doc, docx, xls, xlsx, txt, csv, jpg, jpeg, and png.

Uploaded files appear in the documents table with file name, extension, size, creation date, and actions.

Related Transactions

To view transactions related to an invoice:

Click the **Actions (⋮)** menu in the invoice row.

Select **Related transactions**.

You will see a read-only table listing all transactions linked to the invoice, including currency, amount, type, project, account type, IDs, asset details, status, and actions.

Related Postbacks

To view postbacks associated with an invoice:

Click the **Actions (⋮)** menu in the invoice row.

Select **Related postbacks**.

The **Related postbacks** page displays all postbacks linked to the invoice, including ID, URL, status, creation date, and available actions.

The Invoices module centralizes invoice creation, tracking, and reconciliation. With powerful filtering, status management, document attachments, and direct links to related transactions and

postbacks, it provides full visibility into invoice lifecycle and payment outcomes within the Banking system.