Sango Finance — User Guide

Version: April 2026

Table of Contents

---

1. Overview

Sango Finance is an iPad application for personal wealth management. It allows you to:

- Track your portfolios of financial **assets** (ETF, stocks, cryptocurrencies) and real estate (REAL ESTATE FUND, direct real estate).
- Automatically retrieve prices via 7 market data providers.
- Plan and track progressive investment strategies (DCA, Value Averaging).
- Record your banking operations and link them to your investment **transstocks**.
- Measure your performance with two complementary metrics (IRR / Time-Weighted Return).
- Synchronize your data across all your devices via iCloud.

Supported Currencies: EUR, USD, GBP, CHF, JPY, CAD, AUD, SEK, NOK, DKK, HKD, SGD

---

2. Managing Investments

2.1 Adding an Asset

An **asset** (Asset) is a financial instrument identified by its ticker, its ISIN code (optional), and its type. It can be held in one or more portfolio lines within your accounts.

Steps:

1. Access the **Assets** section from the main menu.
2. Press **+** to create a new asset.
3. Fill in the mandatory fields:
    - **Name** and **Ticker** (e.g.: MSFT, BTC).
    - **Type**: ETF, Stock or Crypto.
    - **Currency** of quotation (e.g.: USD, EUR).
    - **API Provider** for price updates.

4. Optionally fill in the **ISIN code** (recommended for European ETFs — improves automatic recognition).
5. Choose the **update method**: manual or automatic.
6. Validate. The asset is created without an initial position.
7. Choose the relevant account.
8. Enter the quantity and average cost per unit (ACU).
9. Indicate the position opening date.

Tip: The ACU entered here serves as the calculation basis for unrealized capital gains. It may differ from the transaction price if you have made several successive purchases.

2.2 Supported Asset Types

Type	Description	Examples
ETF	Exchange-traded index funds	CW8, S&P 500 ETF, MSCI World
Stock	Stocks of listed companies	Apple, Microsoft, ASML
Crypto	Cryptocurrencies	BTC, ETH, SOL
Real Estate funds	Real estate investment funds	Corum, HausInvest
Real Estate direct	Real estate assets in portfolio	Rental apartment

Real estate funds and real estate assets have their own model with **bare ownership** management: the estimated price is linearly interpolated between the purchase date and the full enjoyment date.

2.3 Manual vs Automatic Quote Updates

Automatic Update

The application retrieves prices upon launch for all assets whose method is set to **Automatic**. Retrieval is parallel (several assets simultaneously) via configured API providers.

Functioning:

- The application uses the ISIN if available and if the provider supports it (FMP, Finnhub, EODHD).
- Otherwise, it uses the ticker in the format expected by the provider (e.g.: `AAPL:NASDAQ` for Twelve Data).
- The retrieved price is recorded in the history (one entry per day per asset, without duplicates).

Manual Update

Assets set to **Manual** mode are ignored during automatic updates. You must enter the current price yourself from the asset file. This mode is recommended for:

- Real estate funds and real estate (no real-time prices).
- Assets not covered by available APIs.

- Positions for which you wish to control the price used.

Trigger an On-Demand Update

From the asset list, press the **Refresh Prices** button and choose a filter:

Filter	Assets Updated
All	All assets in automatic mode
Stocks	Stocks only
ETF	ETFs only
Crypto	Cryptocurrencies only
Targets DCA/VA	Assets linked to a DCA or Value Averaging goal

2.4 Configuring an API Provider

Sango Finance supports 7 market data providers:

Provider	Main Coverage	ISIN	History
Alpha Vantage	USA, ETF, Forex	No	Yes
FMP (Financial Modeling Prep)	Global, ISIN search	Yes	Yes
Polygon.io	USA, crypto	No	Yes
Twelve Data	Global (format TICKER:EXCHANGE)	No	Yes
Finnhub	Global, ISIN	Yes	No
Marketstack	Global (EOD)	No	No
EODHD	Global, ISIN + ticker	Yes	No

Configuring an API key:

1. Obtain a key on the chosen provider's website.
2. In Sango Finance, go to **Settings** → **Manage assets**→ **select your asset** → **API**.
3. Enter the key for the relevant provider.

API keys are stored in the **iOS secure keychain** (Keychain), never in plain text in the application preferences.

Selecting the provider for an asset: From the asset file → **API Provider** field, choose from the list. The ticker resolution will be automatically adapted to the format expected by the provider.

2.5 Price History

Price history is managed with an adaptive granularity policy to optimize storage:

Period	Conserved Granularity
0 – 6 months	Daily

Period	Conserved Granularity
6 months – 2 years	Weekly (most recent point of the week)
Over 2 years	Monthly (most recent point of the month)

This aggregation runs automatically every 7 days.

Note: Only one entry is kept per day. If you add a price to an existing date, the old value is replaced.

---

3. DCA & Value Averaging Strategies

3.1 What is DCA (Dollar Cost Averaging)?

DCA (fixed-amount programmed investment) consists of investing a constant amount at regular intervals, regardless of the market price.

- **Principle:** Each period, you invest exactly the defined amount, no matter the current value of your portfolio.
- **Advantage:** You naturally buy more shares when prices are low and fewer when they are high, thus smoothing your average cost price over time.

Formula applied in Sango Finance:

(Amount to invest) = (growth per deadline) (always fixed)

3.2 What is Value Averaging?

Value Averaging (VA) is a more active strategy where the amount invested each period is adjusted so that the total portfolio value follows a predefined target growth trajectory.

- **Principle:** You define a target value for each period. If the market has performed well, you invest less (or even sell). If it has dropped, you invest more.

Formula applied in Sango Finance:

Target Value(n) = (initial value) + n × (growth per deadline)
Amount to invest = Target Value(n) – (current value)

- If (Amount to invest) > 0 → recommended purchase.
- If (Amount to invest) < 0 → recommended partial sale (to stay on trajectory).

Key difference with DCA:

	DCA	Value Averaging
Invested amount	Always fixed	Variable depending on performance
Market reaction	None	Automatic (buys more if it drops)
Complexity	Simple	More demanding to follow

	DCA	Value Averaging
Over-buying risk	Low	Very low (dynamic adjustment)

3.3 Creating a Recurring Investment Plan

1. Access **Goals** from the main menu.
2. Press **+** to create a new goal.
3. Fill in:
    - **Mode**: DCA or Value Averaging.
    - **Frequency**: Monthly (12 deadlines/year) or Bi-monthly (24 deadlines/year).
    - **Initial Value**: starting reference value for VA calculation (0 for pure DCA).
    - **Growth per Deadline**: fixed amount to invest (DCA) or target progression (VA).
    - **Target**: a specific asset **or** an entire account (not both simultaneously).
    - **Fractional stocks**: activate if your broker supports it (rounded to 2 decimal places).
4. Validate. The goal starts at the next deadline.

3.4 How does the application calculate the current value for VA?

The **current value** used in the calculation depends on the target:

- **Target = asset**: sum of valuations of all lines holding this asset in all accounts (quantity × lastPrice).
- **Target = account**: total account positions + liquidity (valuation).

The price used is always Asset.lastPrice — ensure you keep your prices updated for the calculation to be accurate.

3.5 Track and validate a deadline

From a goal file:

- The application indicates the **current deadline** (number of periods elapsed since creation).
- It displays the **amount to invest** calculated in real time.
- Once the transaction is made with your broker, press **Validate deadline** to mark the period as processed.
- The application records the number of the last validated deadline to avoid any double counting.

---

4. Banking Operations and investment transactions

4.1 Recording a Banking Operation

Banking operations represent cash movements in your accounts: deposits, withdrawals, transfers, fees, and interest.

Operation types:

Type	Description
Deposit	Deposit of funds from the outside
Withdrawal	Withdrawal to your bank
Transfer	Transfer between two Sango Finance accounts
Bank Fees	Account maintenance fees, etc.
Bank Interest	Credit interest

Steps:

1. Open the relevant account.
2. Access the **Operations** tab.
3. Press **+**.
4. Choose the type, date, amount, and add an optional note.
5. For a **transfer**, select the recipient account and the amount in the destination currency (useful for multi-currency accounts).

4.2 Link between banking operation and investment transaction

Sango Finance distinguishes between:

- **Banking Operation**: cash movement on your account (inflow/outflow of liquidity).
- **Investment Transaction**: purchase/sale of an asset (modifies your position and your PRU).

These two entities are **complementary** but **independent**. A typical asset purchase generates:

1. A purchase **Transaction** (quantity, unit price, fees).
2. An outgoing Deposit type **banking operation** (or a reduction in account liquidity).

The account maintains a **liquidity** balance separate from the capital invested in assets. Each purchase transaction implicitly reduces the available liquidity.

4.3 Linking an outgoing Transfer to an asset purchase

Recommended flow for recording a purchase financed by a bank transfer:

1. **Record the incoming transfer** to the brokerage account:
    - Type: Deposit.
    - Amount: sum received on the securities account.
2. **Record the purchase transaction** in the same account:
    - Type: Purchase.

- Asset, quantity, unit price, brokerage fees.

The **Cash** column of the account reflects the difference between incoming payments and the capital deployed in assets.

Advice: Use the **Transfer** feature to record a movement from your current account to your brokerage account — this maintains the consistency of your global wealth.

4.4 Split an operation between several investments

There is no automatic split of a single banking operation towards several assets. The recommended flow is:

1. Record **a single banking operation** for the global deposit.
2. Create **as many purchase transstocks** as assets concerned, each with its quantity and price.

Account liquidity decreases as transstocks are recorded, giving you a clear view of residual cash.

4.5 Banking Fee Management during transstocks

Fees are treated at two levels:

Brokerage fees (per transaction):

- Transaction fee field.
- Integrated into the total amount calculation: (Total amount) = (quantity) × (Unit price) + fees.
- Fees increase the average PRU of the position (they are part of the acquisition cost).

General Bank Fees:

- Recorded as an operation.
- Reduce account liquidity.
- Are not attached to a specific transaction.

Real estate fees (Real estate Fund):

- Recorded as a Transaction.
- Deducted from total income in the Real Estate profitability calculation.

---

5. Data and Privacy

5.1 Local Storage of your data

All your financial data is stored **on your device** in a SwiftData database (iOS-encrypted SQLite). No data is sent to third-party servers, with the exception of

requests to price APIs (which only transmit tickers or ISIN codes — never your personal data).

Entities stored locally:

- Your investment accounts
- Your financial assets
- Your positions by account
- History of purchases, sales, dividends
- Banking Operations
- Price history
- Your DCA / Value Averaging plans
- Your real estate investments
- Historical snapshots of total valuation
- Exchange rate cache

In case of database corruption: the application detects the error at startup, automatically deletes corrupted files and recreates an empty database. Your iCloud data is then resynchronized.

5.2 iCloud Synchronization between your devices

Sango Finance automatically synchronizes all your financial data via **iCloud** on all your devices connected to the same Apple ID.

Functioning:

- Synchronization is **transparent** and **automatic**: any modification made on one device is propagated to the others within seconds.
- Your data is encrypted end-to-end via Apple's CloudKit protocol — neither Sango Finance nor Apple can read your financial information.
- Synchronization covers all entities: accounts, assets, positions, transstocks, price history, DCA/VA goals, and banking operations.

Prerequisites:

1. Be connected with the same Apple ID on all devices.
2. Have **iCloud Drive activated** in Settings → [Your name] → iCloud.
3. Have sufficient iCloud space (the database size depends on your data volume and price history).

In case of desynchronization:

- Close and relaunch the application on the relevant devices.
- Check your Internet connection and iCloud status in System Settings.

5.3 Exporting Your Data

Sango Finance allows you to export all your data in **XLSX (Excel)** format.

Export content: The export generates a multi-sheet Excel file containing all your entities: accounts, assets, positions, transstocks, banking operations, price history, VA goals, Real estate funds investments, and wealth snapshots.

Procedure:

1. Access **Settings** → **Data** → **Export**.
2. Press **Export to XLSX**.
3. Choose the save location (Files, AirDrop, etc.).

Import: The application supports import from **.xlsx** and **.xls** files (Excel 2003 format), allowing you to restore your data or migrate from another source.

Note: For the first import of your data into Sango Finance, you can first perform an export to directly fill in the correct Excel file structure. The first sheet of the file contains acceptable names for variables by language.

5.4 API Key Security

Your API keys (Alpha Vantage, FMP, Polygon, Twelve Data, Finnhub, Marketstack, EODHD) are stored in the **iOS Keychain (Keychain)** with the service identifier com.sangofinance.apikeys.

Security guarantees:

- Keys are **never stored in plain text** in preferences or the database.
- They are **never included in request URLs** (transmission by HTTP header only, avoiding logging in proxies).
- Access to the Keychain is protected by the iOS system **Face ID / Touch ID**.
- In case of migration from an older version using UserDefaults, the key is automatically transferred to the Keychain upon first launch.