Detailed Functional Modules — ECU

*This document presents each module of the ECU application in the same format: **functional objective**, **available actions**, **data and fields** (UI component + status) and **system statuses to plan for**. It is based on the Functional Specifications and the User Story Template.*

Legend

- The priorities `Must` / `Should` / `Could` follow those of the user story template.
- 💡 : design addition (field or behavior inferred, not written as such in the source documents).
- ⚠️ : open or contradictory point between the sources (see Appendix D).
- **Status** of a field: *Required* or *Optional* (with the display condition if needed).

0. Onboarding (Sign-up & Login)

Functional objective

Enable a quick, reassuring start, adapted to the user's level of financial literacy: create an account, personalize the experience, choose the primary currency and create the first accounts. This module is a **prerequisite** for accessing all the other modules.

Available actions

- Create an account by phone (with OTP code) or by email + password `Must`
- Verify the OTP code `Must`
- Log in to an existing account 💡
- Answer a profile questionnaire (optional, can be “skipped”) `Should`
- Choose the primary currency (list + search) `Must`
- Guided creation of the first account(s) (at least one required) `Must`
- Follow a visual tutorial of 3 to 5 screens (can be “skipped” at any time) `Could`
- Arrive at the main dashboard
- Resume an interrupted onboarding where it left off
- Recover one's account through an alternative method (security question or backup code) ⚠️

Data & fields — Sign-up

Field	UI component type	Status
Phone number	Phone field 💡 (with country code)	Required if signing up by phone
OTP code	Short numeric field ⚠️ (number of digits not specified)	Required if signing up by phone
Email	Email field	Required if signing up by email
Password or PIN	Masked field (one or the other is set at creation)	Required

Data & fields — Login 💡

Field	UI component type	Status
Phone number or email	Phone / email field	Required
Password or PIN	Masked field	Required

Data & fields — Profile questionnaire

Field	UI component type	Status
Level of financial literacy	Single choice (Beginner / Comfortable / Advanced)	Optional
Main goal	Single choice (Save / Budget / Get out of debt / Grow one's wealth)	Optional

The template provides for a maximum of 4 to 6 questions; only these two questions are described in the functional specifications ⚠️.

Data & fields — Choice of the primary currency

Field	UI component type	Status
Primary currency	Selector with search bar (common currencies first: FCFA/XOF, XAF, EUR, USD, GBP, NGN, GHS…)	Required

Data & fields — Guided creation of the first accounts

Field	UI component type	Status
Account type	Guided choice (Cash / Mobile money / Bank / Informal savings)	Required
Initial balance	Numeric field + currency (option "I'd rather enter this later"; 0 allowed)	Optional
Other fields (name, currency, icon/color, operator)	See module 1	—

Data & fields — Alternative account recovery ⚠

Field	UI component type	Status
Security question / backup code	Text field (to be defined)	Required for this recovery method

System statuses to plan for

- Registration refused: phone number or email **already in use** → clear message + option to log in.
- Validation error (missing required field, invalid phone or email format) 💡 .
- OTP code being sent / incorrect or expired code 💡 .
- Success (account created and verified).
- Questionnaire skipped → default profile **"regular user"** applied.
- No account created: onboarding cannot be completed.
- Onboarding interrupted: resumes at the exact step the next time the app is opened.

Rules to reflect in the interface

- Registration can be completed in **under 2 minutes**.
- Creating **at least one account is required** to complete onboarding (without an account, no transaction is possible) ⚠️ .
- The primary currency can be changed later; a **warning** then explains the impact on consolidated views. A message states from onboarding that the currency can also be set **per account**.
- The profile can be changed later in the settings.
- The tutorial remains accessible from the settings / help.
- User with no email or verifiable phone number: provide an alternative recovery method ⚠️ .

1. Account management

Functional objective

Provide a faithful, flexible representation of all the places money is held in daily life (cash, mobile money, bank, informal savings), with a currency specific to each account and a consolidated view.

Available actions

- Create an account `Must`
- Edit an account
- Delete an account (only if it contains no transactions) ⚠️
- Archive an account
- View the list of active accounts, with each account's balance and the **consolidated total** at the top of the screen `Must`
- Sort the list by account type or balance `Must`
- View an account's details (balance, transaction history filtered on this account)
- View the history of an archived account (e.g. for an export)
- Transfer money between two accounts `Must`
- Manually adjust an account's balance (“Correction” transaction) `Should`
- Change the application's primary currency (see module 12) `Should`

Data & fields — Account creation / editing

Field	UI component type	Status
Account name	Short text field (free-form)	Required
Account type	Selector (Cash / Mobile money / Bank / Informal savings / Other)	Required
Operator	Selector (Orange Money, MTN MoMo, Wave, Moov Money, Airtel Money, Other) — informational field, no technical integration in V1	Optional, visible if type = mobile money
Currency	Currency selector	Required (editable as long as the account has no transactions)
Initial balance	Numeric field + currency (with calculator)	Required (0 allowed)

Field	UI component type	Status
Icon and color	Icon selector + color selector	Required 💡 (default values suggested)
Tontine-specific fields	See module 4	Visible if type = informal savings / tontine

Data & fields — Transfer between accounts

Field	UI component type	Status
Source account	Selector with icon/color	Required
Destination account	Selector with icon/color	Required
Amount	Numeric field + currency (with calculator)	Required (≠ 0)
Conversion rate	Numeric field	Required if the two accounts' currencies differ (stored on the transfer)
Date	Date selector (default: today)	Required 💡

Data & fields — Balance adjustment

Field	UI component type	Status
New balance	Numeric field + currency	Required 💡
Reason	Short text field	Optional

System statuses to plan for

- No accounts 💡 .
- Loading data 💡 .
- Input error (missing required field, invalid balance, amount of 0, source account identical to destination account) 💡 .
- Confirmation required before archiving or deletion.
- **Negative balance** (allowed, never blocking): “negative balance” badge on the account + notification.
- **Transfer linked to an archived account**: “archived account” label in the history.
- **Multi-currency consolidated total**: visual indicator of indicative conversion.
- Successful operation (creation, edit, archiving, transfer, adjustment) 💡 .

Rules to reflect in the interface

- **Balance = initial balance + income − expenses + incoming transfers − outgoing transfers + adjustments**, recalculated in real time and identical on all screens.
- An account containing transactions cannot be deleted: it is **archived** (hidden from active lists, history viewable) ⚠️.
- A currency badge is visible on each account in the list.
- The consolidated total converts foreign-currency accounts using a **manually adjustable reference rate** (no automatic retrieval in V1) ⚠️.
- A transfer **does not affect** expense and income statistics.

2. Income and expense tracking

Functional objective

Make manual entry of every movement of money quick and frictionless, and make it easy to find later in the history.

Available actions

- Add an expense (accessible in **1 to 2 taps from the dashboard**) `Must`
- Add income `Must`
- Use the built-in calculator in the amount field (see module 3) `Must`
- Choose a predefined category, or create custom categories and subcategories `Must`
- Set up a recurring transaction (rent, subscription, salary) `Should`
- Confirm a recurring occurrence before final validation
- Duplicate an existing transaction `Should`
- Edit or delete a transaction at any time
- View the history (chronological list, search, combinable filters) `Must`
- View a transaction's details
- Add free-form tags `Could`
- Export the history (see module 10)

Data & fields — Expense

Field	UI component type	Status
Amount	Numeric field + currency (with calculator icon)	Required (≠ 0)

Field	UI component type	Status
Account	Selector with icon/color	Required
Type	Expense / Income toggle	Required
Category	Selector with icon (expense categories)	Required
Date	Date selector (default: today; future dates allowed)	Required
Note	Text field	Optional
Receipt photo	Image upload	Optional
Tags	Multi-value selector (free-form)	Optional (`Could`)

Data & fields — Income

Field	UI component type	Status
Amount	Numeric field + currency (with calculator icon)	Required (≠ 0)
Account	Selector with icon/color	Required
Type	Expense / Income toggle	Required
Income category	Selector with icon (Salary, Self-employment, Donation/aid, Sale, Other)	Required
Date	Date selector (default: today; future dates allowed)	Required
Note	Text field	Optional
Receipt photo	Image upload	Optional
Tags	Multi-value selector (free-form)	Optional (`Could`)

Data & fields — Recurrence (expense or income)

Field	UI component type	Status
Frequency	Selector (Weekly / Monthly / Custom) ⚠️	Required
Date of the next occurrence	Date selector 💡	Required
Stopping the recurrence	“Stop” action available at any time	—

Data & fields — Custom category

Field	UI component type	Status
Name	Short text field	Required
Icon	Icon selector	Required 💡
Type	Selector (Expense / Income)	Required 💡
Parent category	Selector (to create a subcategory)	Optional

Data & fields — History filters

Field	UI component type	Status
Text search	Short text field (applies to the “note” field)	Optional
Account	Selector	Optional
Category	Selector	Optional
Period	Date-range selector	Optional
Tag	Selector	Optional
Type	Selector (expense / income / transfer…) ⚠️	Optional

Filters are **combinable**.

System statuses to plan for

- No transactions recorded 💡 .
- Real-time validation of required fields 💡 .
- **Amount of 0: not allowed**, explicit error message.
- Upload in progress (receipt photo) 💡 .
- Success after validation: the account balance is updated immediately.
- **Future date**: allowed, transaction visually distinguished by a “planned” label.
- **Negative balance after entry**: allowed, with badge and notification.
- **Recurring occurrence to confirm**: confirmation notification before final validation (adjustable amount, e.g. variable bill).
- Filtered list with no results 💡 .
- Deletion: confirmation requested; the account balance is recalculated immediately.

Rules to reflect in the interface

- Quick form: **fewer than 4 required fields** (amount, account, category, date).
- Default expense categories adapted to the local context: Transportation/motorcycle taxi, Food, Health, Housing, Family/money transfer, Tontine, Education, Leisure, Other. Number of custom categories: **unlimited**.
- Deleting a category in use: past transactions keep it ("deleted category" or reclassified as "Other") ⚠️ ; it no longer appears for new entries.
- The "Duplicate" button suggests today's date, which can be changed.

3. Calculator

Functional objective

Make it easy to calculate the total of a composite expense (e.g. 2,000 + 1,500 + 750) directly in the amount field, without leaving the form. The calculator is **cross-cutting**: it is available from any amount input field (expense, income, savings, budget, loan), with strictly identical behavior everywhere.

Available actions

- Open the calculator from the amount field's icon `Must`
- Perform operations: addition, subtraction, multiplication, division `Must`
- Confirm with "=" to automatically carry the result into the amount field `Must`

Data & fields — Calculator

Field	**UI component type**	**Status**
Calculation expression	Numeric keypad with + − × ÷ operations (displayed as an overlay on the amount field)	Required
Result	Numeric value carried into the amount field on confirmation ("=")	Required

System statuses to plan for

- Calculation in progress 💡 .
- Calculation error (e.g. division by zero) 💡 .
- Result successfully carried into the amount field.
- The result follows the **numeric format of the currency** selected.

4. Savings

Functional objective

Encourage active, motivating savings, also suited to informal savings practices: flexible savings (“pocket”), goals with an amount and a due date, tontine.

Available actions

- Create flexible savings (a pocket with no specific goal, linked to an account or standalone) `Must`
- Manually add money to or withdraw money from flexible savings `Must`
- Create a savings goal (target amount + target date) `Must`
- View the list of pockets and goals (with progress bar)
- View a goal's details (contribution history, remaining amount, due date)
- Contribute to a goal
- Close a goal that has been reached (transfer to flexible savings or a new goal) or keep contributing
- Record a tontine contribution `Should`
- View the tontine screen (contributions, next collection round)
- Receive encouragement messages at milestones `Could`
- Track several goals in parallel with a consolidated view and priority order `Could`

Data & fields — Flexible savings / Savings goal

Field	UI component type	Status
Type	Selector (Flexible savings / Goal)	Required
Name	Short text field	Required
Target amount	Numeric field + currency (with calculator)	Required for a goal (absent for flexible savings)
Target date	Date selector	Required for a goal (absent for flexible savings)
Associated account / pocket	Selector with icon/color	Required
Icon	Icon selector	Optional

Data & fields — Contribution / Withdrawal

Field	UI component type	Status
Amount	Numeric field + currency (with calculator)	Required (≠ 0)
Source account	Selector with icon/color	Required
Date	Date selector (default: today)	Required
Note	Text field	Optional

Data & fields — “Tontine / informal savings” account type

Field	UI component type	Status
Usual contribution amount	Numeric field + currency	Optional
Frequency	Selector	Optional
Next collection date	Date selector	Optional
Contribution reminder	Reminder lead-time selector (linked to the Notifications module)	Optional

System statuses to plan for

- No savings created 💡 .
- Loading data 💡 .
- Input error 💡 .
- **Milestones reached (25%, 50%, 75%, 100%)**: congratulatory message.
- **Goal behind schedule**: non-judgmental message offering a concrete suggestion (e.g. temporarily reducing a spending category identified as reducible).
- **Goal reached before the due date**: offer to close it or keep contributing.
- **Due date passed without the goal being reached**: the goal remains active, the suggestion is recalculated and the delay is flagged, with no blocking.
- Alert if the total of planned monthly contributions exceeds the estimated savings capacity (consolidated view, `Could`).

Rules to reflect in the interface

- **Suggested periodic contribution = remaining amount ÷ number of remaining periods** until the due date (e.g. “you need to save 25,000 FCFA/month to reach your goal”).

- Every contribution is a **“Savings”** transaction, visible in the general history and linked to the goal.
- The savings balance is shown **separately from the “available” balance**.
- The tontine has a **separate history** of contributions (paid; received ⚠️).

5. Budget management

Functional objective

Help the user plan their spending and stay in control, with active guidance (not just a display).

Available actions

- Define a budget by category and period `Must`
- Edit a budget at any time
- View the overview of the current month's budgets (progress bars by category)
- Receive a starting budget suggestion and accept, edit or ignore it `Should`
- Duplicate the previous month's budget in one tap `Should`
- View the history of past budgets `Should`
- Compare planned vs. actual, monthly and multi-month (3, 6, 12 months) `Could`

Data & fields — Budget definition

Field	**UI component type**	**Status**
Category	Selector with icon	Required
Limit amount	Numeric field + currency (with calculator)	Required
Period	Selector (monthly by default, configurable)	Required
Scope	Selector (global / per account) ⚠️	Optional
Alert thresholds	Two fields in % (80% and 100% by default)	Optional

Data & fields — Starting budget suggestion

Field	**UI component type**	**Status**
Proposed allocation (50/30/20: essentials / wants / savings)	Editable list by category	—

Field	UI component type	Status
Action	Accept / Edit / Ignore	Required
Explanation	“Why this suggestion?” link	—

System statuses to plan for

- No budget defined 💡 .
- Loading data 💡 .
- Input error (invalid amount) 💡 .
- **80% used**: orange indicator + notification.
- **100% used**: red indicator + immediate notification + message with actionable advice (e.g. “you have 5 days and 3,000 FCFA left on this budget”).
- **New period**: invitation to duplicate the previous budget.
- **Suggestion available**: after 2 to 4 weeks of data, or from onboarding via a self-reported estimate.

Rules to reflect in the interface

- The amount used is **calculated in real time** from the expenses of the relevant period and category.
- If the user takes no action, the budget is **not carried over automatically** (a deliberate choice, to avoid outdated budgets).
- A category **with no budget** is counted in the reports but **triggers no alert**.
- A budget modified during the period, even after an overrun, applies **immediately** to the progress bar.
- Planned vs. actual chart with automatic identification of categories that regularly go over budget.

6. Wealth management

> Module **hidden by default** for a user identified as “Novice”; can be enabled in the settings (“Enable Advanced Asset View”).

Functional objective

Give a comprehensive, rewarding view of what the user owns, beyond the simple balance of everyday accounts: assets, liabilities and net worth.

Available actions

- View the wealth dashboard (assets, liabilities, net worth, breakdown) `Should`
- Add, edit, delete an asset `Should`
- Manually update an asset's value (every change is logged) `Should`
- Add, edit, delete a manual debt (outside the Loans module) `Should`
- View the evolution of net worth (monthly / annual) and compare two periods `Could`

Data & fields — Asset

Field	UI component type	Status
Name	Short text field	Required
Category	Selector (Real estate / Vehicle / Equipment / Investment / Other)	Required
Estimated value	Numeric field + currency (with calculator)	Required
Currency	Currency selector	Required
Acquisition date	Date selector	Required
Liquidity status	Selection (Liquid / Illiquid)	Required

Data & fields — Manual debt ⚠️

Field	UI component type	Status
Name / creditor 💡	Short text field	Required
Amount still owed 💡	Numeric field + currency	Required
Currency 💡	Currency selector	Required
Due date 💡	Date selector	Optional

System statuses to plan for

- No assets or liabilities 💡 .
- Loading data 💡 .
- Input error 💡 .
- **Confirmation required before deleting an asset**; the valuation history is kept.
- **Foreign-currency asset or account**: indicative-conversion indicator.

- Module hidden (novice profile): invitation to enable it in the settings 💡 .

Rules to reflect in the interface

- **Net worth = sum of account balances + value of assets − sum of liabilities** (outstanding loans + manual debts).
- The **outstanding loans appear automatically as liabilities**, with no duplicate entry; a “Paid off” loan leaves the active list.
- No automatic valuation of assets in V1.
- Pie chart by asset/liability type; monthly trend line.

7. Loan management

Functional objective

Help the user track their debts and repayments, and understand their impact on their budget.

Available actions

- Record an outstanding loan or credit `Should`
- View the list of outstanding loans, with repayment progress
- View a loan's details (repayment schedule, repayment history)
- Record a repayment `Should`
- Receive a reminder before each due date `Must` (once the submodule is enabled)
- Simulate an early repayment or a new loan `Could`
- View the history of repaid loans

Data & fields — Loan creation

Field	UI component type	Status
Loan type	Selector (Bank / Informal-family / Microfinance / Other)	Required
Initial amount	Numeric field + currency (with calculator)	Required
Interest rate	Numeric field (%)	Optional
Term	Numeric field (in months)	Required
Start date	Date selector	Required

Field	UI component type	Status
Monthly payment	Numeric field + currency (with calculator)	Required
Account planned for payment	Selector with icon/color	Required 💡
Reminder lead time before due date	Selector (e.g. 3 days before)	Optional

Data & fields — Repayment

Field	UI component type	Status
Amount	Numeric field + currency (with calculator)	Required (≠ 0)
Source account	Selector with icon/color	Required
Date	Date selector (default: today)	Required 💡

Data & fields — Simulator

Field	UI component type	Status
Amount	Numeric field + currency (with calculator)	Required
Term	Numeric field (in months)	Required
Rate	Numeric field (%)	Required

System statuses to plan for

- No loans recorded 💡 .
- Loading data 💡 .
- Input error 💡 .
- **Repayment of an amount different from the monthly payment**: accepted; the outstanding principal is recalculated and the schedule adjusted.
- **Loan paid off before the end of the schedule**: automatic "Paid off" status, removed from active liabilities, viewable in the history.
- **Simulator**: educational warning if the estimated debt-to-income ratio exceeds **33%** of the reported income ⚠️ .

Rules to reflect in the interface

- On creation, a **simple projected repayment schedule** is generated (fixed monthly payment × number of months), without complex compound interest: always presented as an **“estimate”**.
- Each repayment is a transaction linked to the loan: it **reduces the outstanding principal** and **affects the source account's balance**.
- The due-date reminder shows the **amount due** and the **planned account**.
- The simulator compares the available monthly budget **before / after**.

8. Financial education & guidance

Functional objective

Go beyond simply displaying numbers to provide genuine guidance: contextual advice, educational content, decision support, in a supportive, non-preachy tone.

Available actions

- View the “My Financial Coach” feed (personalized advice) `Must`
- Browse the educational content library, by topic and level `Should`
- Mark content as read, track progress, earn badges `Should`
- Receive micro-content tailored to a financial event `Should`
- Use the “Can I afford this?” feature and view the simulation history `Should`
- View the monthly financial review `Could`
- View the financial health score and the details of how it is calculated `Could`

Data & fields — “Can I afford this?”

Field	**UI component type**	**Status**
Amount	Numeric field + currency (with calculator)	Required
Category	Selector with icon	Required

Data & fields — Content library

Field	**UI component type**	**Status**
Topic	Selector (Savings, Budget, Debt, Wealth, Mobile money, Tontine…)	Optional
Level	Selector (Beginner / Intermediate / Advanced)	Optional

Field	UI component type	Status
Format	Article / Video / Quiz (2 to 5 minutes)	Optional 💡

System statuses to plan for

- **Too little data** (first days of use): no "empty" advice; the general content library takes over.
- Loading data 💡 .
- Loading error 💡 .
- **Content already viewed**: marked "read", still accessible, may be re-suggested if relevant.
- **"Can I afford this?" result**: Yes / Be cautious / Not recommended, with a one-sentence justification.
- **Monthly review available**: generated automatically on the 1st day of the following month.

Rules to reflect in the interface

- Advice engine **based on rules** (V1): e.g. more than 30% of income on a single category, 3 consecutive months of overspending in the same category, no savings contribution in the last 30 days.
- Contextual suggestions limited to **1 to 2 per week at most**; the library remains freely accessible.
- "Can I afford this?" cross-references the **remaining budget** of the category, the **planned savings** and the **loan payments due in the next 30 days**.
- The monthly review contains: a summary of income / expenses / savings / debt, strengths, points of attention, **2 to 3 prioritized recommendations**; viewable in the app and sendable by notification or email.
- The **financial health score** relies on transparent, weighted criteria (saving consistency, budget adherence, debt-to-income ratio); the calculation is explained ("How this score is calculated") and its evolution is visible over time.

9. Notifications / alerts

Functional objective

Maintain engagement and prevent financial problems in a timely manner, without being intrusive.

Available actions

- Receive a (push) notification or an in-app alert `Must`
- View the notification center (all notifications sent) `Must`

- Enable or disable each notification type independently `Should`
- Configure the time and frequency of the entry reminder `Should`
- Configure alert thresholds and reminder lead times `Should`
- Choose the channel (push, email) if available `Should`

Data & fields — Notification preferences

Field	UI component type	Status
Entry reminder	Toggle	Optional
Budget alert	Toggle	Optional
Low balance alert	Toggle	Optional
Loan / tontine reminder	Toggle	Optional
Educational content	Toggle	Optional
Entry reminder time	Time selector	Optional
Entry reminder frequency	Selector (daily / weekly)	Optional
Low balance threshold	Numeric field + currency (e.g. 5,000 FCFA)	Optional
Budget alert thresholds	Two fields in % (80% and 100% by default)	Optional
Loan / tontine reminder lead time	Selector (e.g. 3 days, 1 day before)	Optional
Channel	Selector (push / email if available)	Optional

System statuses to plan for

- No notifications 💡 .
- Unread notifications present 💡 .
- **Three visual levels**: info / attention / urgent.
- **Critical alerts** (budget at 100%, negative balance): sent immediately as push.
- **Entry reminder**: sent at the configured time; frequency reduced if the user is already entering transactions regularly (e.g. at least 1 transaction/day over the last 7 days).

Notification catalog

Notification	Level	Trigger
Entry reminder	Info	Configured time and frequency

Notification	Level	Trigger
Budget at 80%	Attention	80% of the budget used
Budget at 100%	Urgent (immediate)	Budget overrun
Negative balance	Urgent (immediate)	Balance < 0 after an entry
Low balance	Attention	Balance < configured threshold
Recurring occurrence to confirm	Info	Occurrence date reached
Loan due date	Attention	X days before (e.g. 3), with amount and planned account
Tontine contribution	Attention	X days before, with amount and account
Savings milestone	Info	25 / 50 / 75 / 100% of a goal
Goal behind schedule	Info	Delay relative to the due date
Advice / contextual content	Info	Event (1 to 2 per week maximum)
Monthly review	Info	1st day of the following month

10. Reports / analytics (Statistics)

Functional objective

Turn the entered data into actionable insights for decision-making, and give a quick view of the month's situation right from the home screen.

Available actions

- View the home dashboard (summary of the month) `Must`
- View the breakdown of expenses by category over a period `Must`
- Compare income, expenses and savings over 3, 6 or 12 months `Should`
- Export the data (CSV and PDF at minimum) `Should`
- View advanced analytics: savings rate, debt-to-income ratio, disposable income `Could`
- Filter all charts by period, account and category
- Customize the dashboard (rearrange / hide widgets) `Could`, planned for the advanced version (V5) ⚠️

Data & fields — Filters

Field	UI component type	Status
Period	Selector (Week / Month / Year / Custom with date range)	Required
Account	Selector	Optional
Category	Selector	Optional

Data & fields — Export

Field	UI component type	Status
Format	Selector (CSV / PDF)	Required
Period and filters	Taken from the active filters; the file header states them	Required

Data & fields — Home dashboard

Element	UI component type	Status
Total consolidated balance	Numeric value + indicative-conversion indicator if several currencies	Displayed by default
Progress of the month's budget	Progress bar(s)	Displayed by default
Progress of active savings goals	Progress bars	Displayed by default
Current alerts	List of cards (info / attention / urgent)	Displayed by default
Rearranging / hiding widgets	Movable widgets	Optional (V5)

System statuses to plan for

- Loading data 💡 .
- No data for the selected period 💡 .
- Loading error 💡 .
- **Multi-currency consolidated view**: indicative-conversion indicator.
- **Privacy mode active**: amounts hidden.

Rules to reflect in the interface

- Charts are **dynamically recalculated** according to the filters; no frozen pre-aggregated data.
- Expense breakdown: pie or bars, with the **top 3 to 5 categories** highlighted.
- Formulas for the advanced indicators:
  - **Savings rate** = total saved ÷ total income (over the period).
  - **Debt-to-income ratio** = total monthly loan payments ÷ total income (over the period).
  - **Disposable income** = income − (identified fixed expenses + monthly loan payments) ⚠️.
- An educational tooltip defines each indicator.
- The export uses the filters active at the time of export.

11. Security / account

Functional objective

Ensure trust and the protection of personal financial data, which is sensitive by nature.

Available actions

- Protect access to the app with a PIN or biometrics (fingerprint / Face ID) `Must`
- Automatically lock the app after a configurable period of inactivity `Must`
- Enable privacy mode, which hides amounts ("eye" icon) `Should`
- Back up data (cloud, with consent) and restore it after a reinstall or a device change `Should`
- Manage account information: email, phone, password `Must`
- Permanently delete the account and data `Should`

Data & fields — Lock

Field	UI component type	Status
PIN code	Masked numeric field (set up during onboarding or in the settings)	Required on activation
Biometrics	Toggle (if the phone supports it)	Optional
Inactivity delay before locking	Selector (default: immediately on every app restart)	Optional

Data & fields — My account

Field	UI component type	Status
Email	Email field	Optional
Phone number	Phone field	Optional
Password	Masked field (💡 old, new, confirmation)	Optional
Confirmation code	Numeric field (OTP) or email link	Required for any sensitive change

Data & fields — Privacy, backup and deletion

Field	UI component type	Status
Privacy mode	Persistent "eye" icon / toggle	Optional
Automatic cloud backup	Consent toggle	Optional
Restore	Action button, after authentication	Optional
Account deletion	Explicit double confirmation	Required to delete

System statuses to plan for

- Lock screen on every opening after the inactivity delay.
- **Privacy mode active**: all amounts replaced by asterisks or equivalent; state remembered between sessions.
- Backup successful / restore in progress 💡 .
- Validation error (incorrect code, confirmation does not match) 💡 .
- **Account deletion**: permanent, preceded by a double confirmation; all personal data is deleted.

12. Settings

Functional objective

Centralize the settings used by the other modules: primary currency, reference rates, profile, categories, notifications, security and help.

Available actions

- Change the primary currency (with a warning) `Should`

- Edit the reference rates used for indicative conversions
- Edit one's profile (literacy level, main goal)
- Enable the advanced wealth view
- Manage categories and subcategories (see module 2)
- Configure notifications (see module 9)
- Configure security (see module 11)
- Replay the tutorial / access help `Could`
- Access the data export (see module 10)

Data & fields — Settings

Field	UI component type	Status
Primary currency	Currency selector	Required
Reference rate (for each currency different from the primary currency)	Manually editable numeric field	Optional ⚠️
Level of financial literacy	Single choice (Beginner / Comfortable / Advanced)	Optional
Main goal	Single selection	Optional
Enable the advanced wealth view	Toggle	Optional

System statuses to plan for

- **Warning when changing the primary currency**: explains the impact on consolidated views (wealth, reports); accounts keep their own currency.
- Successful save after a change 💡 .
- Validation error (e.g. invalid rate) 💡 .

Appendix A — Legends

Generic system statuses to include for every list or form

- **Empty state** (no data) — with a message and a call to action.
- **Loading state** (skeleton / spinner).
- **Error state** (field validation or network error).
- **Success / confirmation state.**
- **“No results” state** for filtered lists.
- **Privacy mode active**: amounts hidden everywhere.

- **Multi-currency view**: currency badge and indicative-conversion indicator.
- **Confirmation** before any deletion; **double confirmation** for deleting the user account.

UI components used

Component	Usage
Short / long text field	Names, notes, reasons
Numeric field + currency	Amounts, balances, thresholds (with **calculator icon** on every amount field)
Masked field	Password, PIN code
Phone / email field	Sign-up, login, account
Selector (single / multiple)	Types, periods, frequencies, reference lists
Selector with icon/color	Accounts, categories
Date / date-range / time selector	Transactions, filters, due dates, reminders
Icon / color selector	Accounts, categories, goals
Toggle	Preferences, activations, consents
Image upload	Receipt photo
Progress bar	Budgets (orange at 80%, red at 100%), savings goals, repayments
Badge	Negative balance, currency, “planned”, “archived account”, “read”, “Paid off”

Appendix B — Cross-cutting rules

Rule	Content
Multi-currency	Any consolidated view (dashboard, wealth, reports) that aggregates different currencies displays a visual indicator showing that an **indicative conversion** has been applied.
Calculator	Available from any amount field (expense, income, savings, budget, loan), with strictly identical behavior everywhere.
Balance consistency	Any operation that changes a balance (transaction, transfer, adjustment, loan repayment) recalculates and displays that balance **in real time** on all relevant screens, with no stale cache.
Historical data	No data is silently deleted: any deletion requires confirmation; related items remain traceable or are explicitly managed (**archiving rather than physical deletion** when appropriate).

Appendix C — Reference lists (value lists)

Reference list	**Values**
Account types	Cash · Mobile money · Bank · Informal savings / tontine · Other
Mobile money operators	Orange Money · MTN MoMo · Wave · Moov Money · Airtel Money · Other
Common currencies	FCFA (XOF) · XAF · EUR · USD · GBP · NGN · GHS · (others via search)
Literacy level	Beginner · Comfortable · Advanced
Main goal	Save · Budget · Get out of debt · Grow one's wealth
Default expense categories	Transportation/motorcycle taxi · Food · Health · Housing · Family/money transfer · Tontine · Education · Leisure · Other
Default income categories	Salary · Self-employment · Donation/aid · Sale · Other
Transaction types	Expense · Income · Internal transfer · Adjustment (“Correction”) · Savings · Loan repayment ⚠️
Recurrence frequencies	Weekly · Monthly · Custom
Loan types	Bank · Informal/family · Microfinance · Other
Asset categories	Real estate · Vehicle · Equipment · Investment · Other
Asset liquidity	Liquid · Illiquid
Savings milestones	25% · 50% · 75% · 100%
Budget alert thresholds	80% · 100% (configurable)
Suggested budget allocation	50% essentials · 30% wants · 20% savings
“Can I afford this?” response levels	Yes · Be cautious · Not recommended
Notification levels	Info · Attention · Urgent
Educational content levels	Beginner · Intermediate · Advanced
Filter periods	Week · Month · Year · Custom; comparisons over 3 / 6 / 12 months
Export formats	CSV · PDF (Excel mentioned in the template)