FinLedger

User Manual

Financial management system

Generated: 2026-02-22 23:47:39 UTC

Contents

Overview and Roles

System Overview

FinLedger is a single-page web application with authenticated navigation and role-aware controls.

Current user-facing modules are:

- Dashboard
- Accounts (Chart of Accounts)
- Transactions
- Reports
- Help
- Profile

Important current-state note:

- Journal entry, ledger, adjusting-entry, and full financial-statement workflows are not exposed as complete UI/API user workflows yet, even though related database tables exist.

UI Landmarks

- Header: FinLedger branding, logged-in user label, calendar button, and top navigation.
- Top navigation: Dashboard, Accounts, Transactions, Reports, Help, and Login/Logout.
- Profile menu: top-right avatar/name menu with Go to profile and Logout.
- Global message line: page-level success/error message area below the header.
- Main content area: route content loaded without full page reload.
- Loading overlay: shown during route loads and longer actions.

Role Summary

FinLedger supports three roles: administrator, manager, and accountant.

- Administrator: - Full user administration on Dashboard (approve/reject, create, edit, suspend/reinstate, delete, reset password, email user). - Full Chart of Accounts administration (create/edit/deactivate/reactivate accounts, manage categories/subcategories). - Can access audit log APIs.
- Manager: - No admin user-management controls. - Read-only use of current Accounts list UI. - Can access audit log APIs. - Can use current Transactions, Reports, Help, and Profile pages.
- Accountant: - No admin user-management controls. - Read-only use of current Accounts list UI. - Can use current Transactions, Reports, Help, and Profile pages. - Cannot access audit log APIs.

Module Map

- Dashboard: - Static summary cards plus admin-only user-management workspace.
- Accounts: - Implemented Chart of Accounts list and admin account/category management.
- Transactions: - Current page is a simple activity view with refresh timestamp behavior.
- Reports: - Current page is a lightweight period selector and static summary text.
- Help: - Accordion help topics plus User Manual PDF link.
- Profile: - Personal info updates, password change, and security question updates.

For authentication and password policy details, see docs/User Manual/02LoginandSecurity/README.md. For administrator workflows, see docs/User Manual/03User_Administration/README.md.

Login and Security

This section describes current sign-in, password, reset, and session-security behavior.

Login Screen

The login page is available from the Login link in the header.

The form includes:

- Username
- Password
- Log In
- Request Access
- Forgot Password

After successful sign-in:

- The header shows You are logged in as ....
- The profile button shows name/avatar.
- Navigation updates to show Logout.

Sign-In Process

1. Open the Login page.
2. Enter your username and password.
3. Select Log In.

On success, FinLedger stores session auth data in local storage and routes to #/dashboard.

If login fails, an error message from the message catalog is shown in the global message line.

Session and Access Control

Protected routes require:

- Authorization: Bearer <token>
- X-User-Id

Session behavior:

- Server sessions are tracked in loggedinusers with a sliding timeout.
- Active requests extend session expiration.
-

Session expiry headers are returned to the browser.
- The client warns before timeout and auto-logs out when expired.

If auth is missing or invalid, protected requests fail and the UI redirects to Not Logged In or Login.

Logout

Users can log out from:

- Top navigation Logout
- Profile menu Logout

Logout updates server session state and clears local session data.

Password Complexity Rules

Password validation is enforced on:

- Profile password change
- Temporary-password forced change
- Password reset submit flow
- Admin password set/reset paths

Current required rules:

- First character must be an ASCII letter (A-Z or a-z)
- Minimum length of 8 characters.
- At least one uppercase letter.
- At least one lowercase letter.
- At least one number.
- At least one special character.

Password reuse is blocked by password-history checks.

Failed Login Attempts and Suspension

FinLedger tracks failed login attempts.

- Failed attempts increment on bad password.
- At 3 failed attempts, login is blocked with suspension messaging.
- On successful login, failed-attempt counters reset.

Forgot Password Flow

Use Forgot Password from the login page.

1.

Open Forgot Password from the login page.
2. Enter email and username.
3. Submit reset request.
4. Open the emailed reset link.
5. Answer three security questions.
6. Enter and confirm a new password.
7. Submit.

Reset protections:

- Reset token expires in 1 hour.
- Security answers must match stored hashes.
- Reset-answer failures are lock-limited (3 attempts).
- Password complexity and password-history checks apply.

Temporary Password and Forced Change

Some accounts are issued temporary passwords.

When temp_password is active:

- Login redirects to Change Password.
- Most protected routes are blocked until completion.
- User must submit 3 security Q/A entries and a compliant password.

After completion:

- Temporary-password flag is cleared.
- Normal route access resumes.

Password Expiration and Warnings

Password expiration is enforced.

- Standard passwords are set to expire after 90 days.
- Warning emails are sent in the 3-day window before expiry.
- Expired-password accounts can be auto-suspended by scheduled jobs.

Security Storage and Encryption

Security-sensitive values are stored as hashes:

- Password hashes
- Security answer hashes
- Password history hashes

Reset links use time-limited tokens stored in the database.

Messaging Behavior

Auth and security feedback uses message codes resolved from the app_messages table.

Messages are displayed in the global message line and can be dismissed manually or cleared by new input.

User Administration

This section documents current administrator-only user-management features.

Access and Scope

User administration is exposed on Dashboard only for users with role administrator.

Manager and accountant users:

- Do not see admin user-management cards.
- Cannot use admin-only user-management APIs.

Admin Workspace (Dashboard)

The admin User Management area currently includes:

- User Approvals
- Users with Expired Passwords
- Suspended Users
- Currently Logged-in Users
- All Users table
- Create User
- Delete User
- Reset User Password
- Suspend User Account
- Email User

Registration Request Workflow

New users can request access from the login page using Request Access.

Current flow:

1. Requester submits profile details, role, password, and security questions.
2. FinLedger creates the account with pending status.
3. Requester receives a registration confirmation email.
4. An administrator reviews the request in User Approvals.
5. The admin selects Approve or Reject.

Approval behavior:

-

Status is updated to active.
- An approval email is sent to the user.

Rejection behavior:

- Status is updated to rejected.
- A rejection email is sent.

Create User (Administrator)

Use the Create User card to create accounts directly.

Current fields:

- First Name
- Last Name
- Address
- Date of Birth
- Email
- Role (administrator, manager, accountant)
- Profile Image

Current behavior:

- Username is generated automatically.
- If no password is provided, system generates a temporary password.
- New account status starts as pending.
- Temporary-password account details are emailed to the user.
- Profile image upload is validated by file type and size limits.

View and Edit Users

The All Users table supports:

- Pagination (5, 10, 25, 50, 100 rows per page)
- Inline double-click editing for selected columns
- Display of user lifecycle fields (status, role, login/suspension/password-expiry dates, contact info)

Suspend and Reinstate Users

Suspend flow:

1. Open Suspend User Account.
2. Select an eligible active user (not yourself).
3. Enter suspension start/end date-time.
4.

Submit the form.

Reinstatement:

- In Suspended Users, select Reinstate.

Additional behavior:

- Expired suspensions are automatically cleared by background jobs.
- Users can also be suspended by security policies (failed login attempts, password expiry).

Delete User

Use Delete User to remove an account.

Behavior:

- Admin selects a username.
- Confirmation prompt appears before deletion.
- The UI prevents an admin from deleting their own account.

Reset User Password

Use Reset User Password when a user is locked out.

Current behavior:

1. Select user.
2. Submit reset.
3. System generates a compliant temporary password.
4. Temporary password is emailed to the user.
5. User is required to change it at next login.

Email User

Use Email User to send an administrator-authored message.

Required inputs:

- Recipient username
- Subject
- Message

Logged-In Users and Expired Password Reporting

Currently Logged-in Users shows active sessions.

Users with Expired Passwords highlights accounts that have expired credentials.

This aligns with password-expiry automation (warnings and suspension).

Operational Notes

- Admin-only restrictions are enforced by API checks, not just hidden UI.
- Email delivery requires valid SMTP configuration.
- For password policy and recovery details, see docs/User Manual/02Loginand_Security/README.md.

Navigation and UI

This section describes current navigation and shared UI behavior.

Navigation Model

FinLedger uses a single-page application (SPA) pattern.

- The page frame (header and navigation) stays on screen.
- Main content is loaded into the center content area.
- URL hash routes control which page appears (for example #/dashboard).

Default route:

- If no route is provided, FinLedger loads Dashboard.

Header Layout

The global header includes:

- FinLedger logo and product name
- Logged-in user label (You are logged in as ...)
- Calendar icon (opens pop-up calendar)
- Main navigation links
- Profile button with user name and avatar

Logo behavior:

- Selecting the logo returns the user to Dashboard.

Main Navigation

Primary navigation links:

- Dashboard
- Accounts (route key: #/accounts_list)
- Transactions
- Reports
- Help
- Login or Logout (depends on session state)

Active-page behavior:

-

The current route is highlighted in the navigation bar.

- Browser title updates to FinLedger - <Page Name>.

Route Access and Redirect Behavior

Current route outcomes:

- Valid route: requested page loads into the main content area.
- Unknown route: user is shown the Not Found page.
- Not logged in or expired session: user is redirected to Not Logged In or Login depending on context.
- Insufficient permission: user is redirected to Not Authorized.
- Temporary password required: user is redirected to Change Password.

Global Message Line

FinLedger uses a global message line below the header for success/error feedback.

Behavior:

- Message codes are resolved from the message catalog.
- Error messages use alert-style presentation.
- Success messages use status-style presentation.
- The dismiss button clears the message.
- Editing form inputs automatically clears the current message.

Profile Menu

The top-right profile control opens a menu with:

- Go to profile
- Logout

Interaction behavior:

- Opens on hover/focus.
- Closes on mouse leave, route change, or Esc.

Help and Tooltips

Help:

- Help is a top-level navigation item.
- The Help page is organized as expandable accordion topics.
- Topics cover account access, profile/security, admin tools, reports, and support.
-

Support includes a public link to download the User Manual PDF.

Tooltips:

- Many interactive controls use hover text (title) for quick guidance.
- Examples include page actions, table controls, and form inputs.

Calendar Widget

The header calendar icon opens a pop-up calendar.

Calendar controls:

- Previous month
- Today
- Next month
- Month dropdown
- Year dropdown

Display behavior:

- Shows a month grid with the current day highlighted.
- Closes when clicking outside the calendar popup.

Loading and Page Transition Behavior

FinLedger uses a global loading overlay for page changes and long-running actions.

Behavior:

- Overlay appears during route/page loads.
- Overlay hides after content finishes loading.
- A short minimum display time is applied to prevent flashing.

Current Module UX Status

- Dashboard: operational/admin workspace, plus static summary cards.
- Accounts: implemented account-management UI.
- Transactions: simple activity feed with refresh timestamp update.
- Reports: simple period selector and static summary text.

Layout Consistency

Shared patterns across pages:

- Common header/navigation shell
-

Card and table layout components
- Global message line for feedback
- Global loading overlay for route/action loading
- Role-aware visibility for privileged controls

Chart of Accounts

Scope

The Accounts page (#/accounts_list) is the current Chart of Accounts module.

Role behavior:

- Administrator: full create/edit/status/category management.
- Manager and accountant: read-only table usage.

Account List

The account table includes:

- Account name and number
- Owner
- Status
- Normal side
- Balance and opening balance
- Description and comments
- Category and subcategory
- Statement type
- Account order
- Total debits and credits
- Action column (Audit button rendered per row)

Current navigation behavior:

- Single-click account name or account number opens #/transactions?account_id=<id>.

Filtering, Sorting, and Pagination

Implemented features:

- Header-click filter modal for supported columns.
- Sort ascending/descending for sortable columns.
- Balance range filter (min/max).
- Remove-filters button.
- Pagination controls and rows-per-page selector (5, 10, 25, 50, 100).

Administrator Actions

Add Account

Administrators can add accounts through the account modal.

Key fields include:

- Account name
- Description
- Normal side
- Category/subcategory
- Statement type
- Opening balance
- Account order
- Owner

Account number behavior:

- Generated server-side from category/subcategory/order pattern with suffix handling.

Inline Edit

Administrators can double-click and edit selected columns, including:

- Account name
- Account number
- Owner
- Normal side
- Description
- Category
- Subcategory
- Statement type
- Comment

Category and Subcategory Management

Administrators can:

- Add category (with prefix and initial subcategory)
- Add subcategory to existing category
- Delete category or subcategory

Deletion safeguards:

- Category/subcategory deletion is blocked when accounts are tied to them.

Deactivate/Reactivate

Administrators can deactivate and reactivate accounts.

Status safeguards:

- Deactivation is blocked if account balance is non-zero.

Audit and Event Logging

Current logging behavior:

- Account changes are captured in audit_logs (before/after images).
- Account-specific metadata/debit-credit-balance trails are written to account audit tables.

Current UI status:

- The row Audit button is rendered but not wired to an audit-log viewer.
- Audit access is currently API-based.

Current Gaps

- No dedicated account-audit viewer page is implemented yet.

Journal Entries

Current Status

A complete journal-entry user workflow is not implemented in the current web UI/API.

The expected capabilities below are not yet available as user-facing features:

- Creating journal entries from a dedicated page
- Multi-line debit/credit entry workflow
- Submit/approve/reject workflow with rejection comments
- Journal status views with date filters
- Journal search by account/date/amount
- Source-document attachment from journal-entry UI
- Manager notification on journal submission

What Exists Today

- Database tables exist (journalentries, journalentry_lines).
- Database-level audit triggers exist for journal tables.
- The Transactions page is currently a lightweight activity view and does not create journal entries.

Documentation Note

When journal-entry workflows are implemented, this section should be updated with page-level procedures, role permissions, and validation rules.

Ledger

Current Status

A dedicated ledger screen is not implemented in the current UI.

Not yet available as user-facing features:

- Ledger table with running balance by account
- Ledger filtering by date/account/amount
- Post-reference (PR) navigation to journal entries
- Full ledger drill-down workflow

Current Navigation Behavior

From Accounts:

- Clicking account name/number routes to #/transactions?account_id=<id>.

Current limitation:

- The Transactions page does not currently render account-specific ledger data.

What Exists in the Database

- ledger_entries table exists.
- Audit triggers include ledger tables.

Documentation Note

Update this section when ledger UI/API functionality is exposed to end users.

Adjusting Entries

Current Status

Adjusting-entry workflows are not currently implemented in the user-facing UI/API.

Not yet available:

- Create adjusting-entry form workflow
- Submit/cancel/reset actions before submission
- Manager approval/rejection workflow
- Adjusting-entry status list and filtering
- Posting integration to ledger/statements from UI workflow
- Manager notification on adjusting-entry submission

What Exists Today

- Database tables exist (adjustmentmetadata, adjustmentlines).
- Database audit triggers include adjustment tables.

Documentation Note

This section should be expanded once adjusting-entry pages and APIs are implemented.

Financial Reports

Current Status

The Reports page is currently a lightweight placeholder UI.

Current behavior:

- User can select a period value from a dropdown.
- Page label updates to match selected period.
- Summary text is static.

Not Yet Implemented

The following report workflows are not yet implemented as end-user features:

- Trial balance generation and viewing
- Income statement generation and viewing
- Balance sheet generation and viewing
- Retained earnings statement generation and viewing
- Save/email/print report workflows
- Date/date-range driven report calculations

Role Access (Current)

- Any authenticated user can open the current Reports page.
- Role-specific report authorization for full reporting workflows is not implemented yet.

What Exists in the Database

- Reporting-related tables exist (trialbalanceruns, trialbalancelines, statement_runs).
- Audit triggers include reporting tables.

Documentation Note

Expand this section when true report-generation workflows are delivered.

Dashboards and Ratios

Dashboard (Current)

The Dashboard page is the current landing page after login.

Shared elements:

- Static cards for Cash on Hand, Payables, and Receivables
- Last-updated timestamp
- Refresh button behavior for dashboard data section(s)

Role-based rendering:

- Administrator: - Full User Management workspace (approvals, suspended users, user table, create/delete/reset/suspend/email tools).
- Manager and accountant: - Simplified Your Workspace card only.

Ratios (Current Status)

Financial ratio calculations are not currently implemented in the dashboard UI.

Not available yet:

- Ratio formulas and computed outputs
- Ratio threshold rules
- Green/yellow/red ratio state coding

Alerts and Notifications

Current in-app feedback uses the global message line for action results.

Not available yet:

- Dedicated persistent dashboard notification center
- Ratio-driven alert widgets

Documentation Note

Update this section when dynamic financial metrics and ratio logic are implemented.

Documents and Attachments

Current Status

Document/attachment workflows are not fully implemented in the user-facing app.

Current API route status:

- GET /api/documents/:filename returns "not yet implemented."
- POST /api/documents/upload returns "not yet implemented."

Not Yet Implemented

- Journal-entry source document upload from UI
- Document retrieval/download with permission checks
- User-facing attachment metadata management
- Enforced allowed-file-type rules for user documents

What Exists Today

- documents table exists in the database.
- Audit triggers include documents.
- Cleanup utilities remove unreferenced files from user-docs/.

Related implemented upload feature (separate from documents):

- User profile image upload is implemented with file-type and size validation.

Documentation Note

Update this section when document upload/retrieval workflows are connected to journal or other user flows.

Notifications and Email

Email Features (Implemented)

Current email-trigger scenarios include:

- Registration confirmation email to new requester
- New access-request notifications to administrators
- Approval email after administrator approval
- Rejection email after administrator rejection
- Forgot-password reset-link email
- Administrator reset-user-password email with temporary password
- Administrator direct email to a selected user (Email User)
- Password-expiry warning emails (scheduled)
- Account suspended due to expired password email (scheduled)

Additional account-creation behavior:

- If an admin creates a user without a password, a temporary password is generated and emailed.

In-App Notifications (Implemented)

In-app feedback is currently provided by the global message line:

- Success and error messages are message-code driven from the app_messages catalog.
- Message line supports manual dismissal.
- Message line auto-clears on form input events.

Current Gaps

Not currently implemented:

- Dedicated in-app notification center/inbox
- Journal submission/approval notification workflows
- Adjusting-entry notification workflows

Timing and Automation

Current server background jobs:

- Every 10 minutes: - inactive-session cleanup - unsuspend users whose suspension window ended
-

Every 1 hour: - password-expiry warning processing - suspension of users with expired passwords - user-data cleanup tasks

- Every 24 hours: - log cleanup cycle

Configuration Note

Email delivery depends on SMTP environment configuration (SMTPHOST, SMTPPORT, SMTPUSERNAME, SMTPPASSWORD, SMTPEMAILFROM).

Audit and Event Logs

Current Logging Model

FinLedger currently records change/audit data in multiple tables:

- audit_logs
- account_audits
- accountmetadataedits

The generalized audit trigger captures:

- action (insert, update, delete)
- changed_by
- changed_at
- entitytype/entityid
- before image (b_image)
- after image (a_image)
- metadata

Sensitive fields are sanitized from audit row images (for example password hashes and reset tokens).

Coverage

Audit triggers are configured for many core tables, including:

- users
- accounts
- account categories/subcategories
- documents
- password history and password-expiry tracking
- journal/ledger/reporting-related tables
- app messages

Access (Current)

Audit data is currently exposed through API endpoints:

- GET /api/audit-logs
- GET /api/audit-logs/entity/:entityType/:entityId

Supported filters include:

-

- entity_type
- entity_id
- changed_by
- action
- start_at
- end_at
- limit
- offset

Current access roles:

- Administrator
- Manager

UI Availability

No dedicated audit-log viewer page is currently implemented.

Current UI note:

- The Accounts table renders an Audit button, but it is not wired to an audit-log action yet.

Retention and Cleanup

Log cleanup runs on a daily interval.

Current behavior:

- applogs retention is controlled by APPLOGSRETENTIONDAYS.
- auditlogs retention is controlled by AUDITLOGSRETENTIONDAYS when set to a positive value.
- If AUDITLOGSRETENTION_DAYS is unset or non-positive, audit-log deletion is skipped.