SUSU APP(API Design)

Monday, June 30, 2025 5:21 PM

Modular API Categories

Module	Description
auth	Login, password reset, logout
users	Admin creates & manages collectors
clients	Register & manage clients
contributions	Collectors record/save contributions
savings_cycles	Start and track 31-day cycles
payouts	Trigger and manage payouts
reports	Summary views per day/collector/client

Rest API Endpoint Design

Auth Module

Method	Endpoint	Auth	Purpose
POST	/auth/login	No	Login for admin or collector
POST	/auth/logout	Yes	Logout and invalidate token
POST	/auth/reset-request	No	Request password reset (email/username)
POST	/auth/reset-confirm	No	Confirm token and set new password
POST	/auth/change-password	Yes	Change password (force or manual)

Clients Module

Method	Endpoint	Auth	Purpose
GET	/clients/	Collector/Admin	List my clients
POST	/clients/	Collector	Register new client
GET	/clients/:id	Collector/Admin	Get client profile
PATCH	/clients/:id	Collector/Admin	Edit client info

User Module (Admin only)

Method	Endpoint	Auth	Purpose
GET	/users/	Admin	List all users (admin, collectors)
POST	/users/	Admin	Create new collector
PATCH	/users/:id	Admin	Update collector details
DELETE	/users/:id	Admin	Deactivate collector account

Contributions Module

Method	Endpoint	Auth	Purpose
GET	/contributions/	Collector/Admin	List contributions (filterable)
POST	/contributions/	Collector	Submit contribution (single/bulk)
GET	/contributions/:id	Collector/Admin	Get one contribution record
PATCH	/contributions/:id	Admin	Edit a contribution (if mistake)

Savings Cycles Module

Method	Endpoint	Auth	Purpose
GET	/cycles/	Admin	View all cycles across clients
POST	/cycles/	Collector	Start new cycle for client
GET	/cycles/client/:clientId	Collector/Admin	Get client's cycles
PATCH	/cycles/:id/close	Admin	Close or mark cycle completed/early exit

Payouts Module

Method	Endpoint	Auth	Purpose
GET	/payouts/	Admin	List all pending/approved payouts
POST	/payouts/request	Collector	Trigger payout for client (optional)
PATCH	/payouts/:id/approve	Admin	Approve payout
PATCH	/payouts/:id/pay	Admin	Mark payout as fulfilled

Reports Module

Method	Endpoint	Auth	Purpose
GET	/reports/collectors	Admin	Daily/weekly total per collector
GET	/reports/clients	Admin	See all clients with cycle status
GET	/reports/contributions	Admin	Contributions filtered by date/client

Authentication & Middleware

- **JWT-based Auth**:
  - Issued on login
  - Role-based checks on each route
  - isAdmin, isCollector guards
- **Rate Limiting** for login & reset routes
- **Audit Logging** for every write/mutate request

Stack

Layer	Tech Stack	Why This Choice Works for You
Backend Framework	**Django + Django REST**	Fast to build, secure out-of-box, admin UI
Database	**PostgreSQL**	Relational, great for reports & joins
Auth	**SimpleJWT** (Django JWT)	Supports token-based auth, scalable
Admin Dashboard	**Django Admin (v1)** → optional **React Admin (v2)**	Rapid internal tooling, later custom UI
Hosting	**Railway** / Render / Supabase	Simple CI/CD, database hosting
Storage	Firebase or S3 (for optional profile pics, documents)	
Version Control	GitHub + GitHub Projects	Collaboration, task tracking
Testing	Pytest / Postman / Swagger UI	API test + auto-docs
Rate Limiting	django-ratelimit	Middleware-level protection