Samoa Finance App

A full-stack learner management system for remittance transactions with AML/KYC compliance tracking.

Tech Stack

- **Frontend**: Next.js 15, React 19, TypeScript, Tailwind CSS
- **Backend**: Next.js API Routes
- **Database**: PostgreSQL (Supabase)
- **Authentication**: NextAuth.js
- **ORM**: Prisma

Prerequisites

- Node.js 20+ and npm
- A Supabase account (free tier works)

Setup Instructions

1. Clone the Repository

```
git clone <repository-url>
cd samoa-finance-app
```

2. Install Dependencies

```
npm install
```

3. Set Up Supabase Database

1. Go to https://supabase.com (https://supabase.com) and create a new project
2. Wait for the database to provision (~2 minutes)
3. Go to **Project Settings** → **Database**
4. Copy your **Database Password** (set during project creation)

4. Configure Environment Variables

Copy the example environment file:

```
cp .env.example .env
```

Update `.env` with your Supabase credentials:

```
# Get connection strings from Supabase Dashboard → Settings → Database
DATABASE_URL="postgresql://postgres:[YOUR-PASSWORD]@db.[PROJECT-REF].supabase.co:5432/postgres?
pgbouncer=true&connection_limit=1"
DIRECT_URL="postgresql://postgres:[YOUR-PASSWORD]@db.[PROJECT-REF].supabase.co:5432/postgres"

NEXTAUTH_URL=http://localhost:3000
# Generate with: openssl rand -base64 32
NEXTAUTH_SECRET=your-generated-secret-here
```

To generate a secure NEXTAUTH_SECRET:

```
openssl rand -base64 32
```

5. Run Database Migration

```
npx prisma migrate dev
```

This will:

- Create all database tables
- Apply the schema to your Supabase database

6. Seed the Database

```
npx prisma db seed
```

This creates default users:

- **Admin**: `admin@samoafinance.local` / `Admin@123`
- **Staff**: `staff@samoafinance.local` / `Staff@123`

7. Start Development Server

```
npm run dev
```

Open [http://localhost:3000 (http://localhost:3000)](http://localhost:3000) in your browser.

Available Scripts

- `npm run dev` - Start development server with Turbopack
- `npm run build` - Build for production
- `npm start` - Start production server
- `npm run lint` - Run ESLint
- `npx prisma studio` - Open Prisma Studio (database GUI)
- `npx prisma migrate dev` - Create and apply new migration
- `npx prisma db seed` - Seed database with test data

Project Structure

```
samoa-finance-app/
├── prisma/
│   ├── schema.prisma       # Database schema
│   ├── seed.ts            # Database seeding script
│   └── migrations/        # Database migrations
├── src/
│   ├── app/               # Next.js App Router pages & API routes
│   │   ├── api/          # API endpoints
│   │   │   └── auth/     # NextAuth configuration
│   │   └── ...           # Page components
│   ├── lib/              # Utility functions
│   │   ├── db.ts         # Prisma client singleton
│   │   ├── auth.ts       # NextAuth configuration
│   │   ├── ids.ts        # ID generators (CUST-YYYY-NNNNNN)
│   │   └── validators.ts # Zod schemas
│   └── middleware.ts     # Route protection
├── uploads/              # File uploads (git-ignored)
├── .env                  # Environment variables (git-ignored)
├── .env.example          # Environment template
└── package.json
```

Features

- ☑ Role-based authentication (Admin, Staff, AML, Agent)
- ☑ Customer management with E.164 phone validation
- ☑ Transaction tracking with sequential IDs
- ☑ Multi-currency support (WST, AUD, USD)
- ☑ AML/KYC compliance fields
- ☑ Document upload for customer IDs
- ☑ Timezone-aware operations (Pacific/Auckland)
- ⏳ Daily/monthly reporting
- ⏳ CSV export functionality
- ⏳ Exchange rate management

Database Schema

Key models:

- **User** - System users with role-based access
- **Customer** - Customer profiles with KYC data
- **CustomerIdFile** - Uploaded ID documents
- **Transaction** - Remittance transactions with AML tracking
- **ExchangeRate** - Daily exchange rates

- **ActivityLog** - Audit trail

Default Users

After seeding, you can log in with:

Role	Email	Password
Admin	admin@samoafinance.local	Admin@123
Staff	staff@samoafinance.local	Staff@123

Production Deployment

Environment Variables

For production, ensure you set:

- Strong `NEXTAUTH_SECRET` (use `openssl rand -base64 32`)
- Production `NEXTAUTH_URL` (your deployed URL)
- Secure Supabase connection strings

Deployment Options

- **Vercel** (Recommended for Next.js)
- **Railway**
- **AWS/Azure/GCP** with Docker

Database Migration

```
npx prisma migrate deploy
```

Security Notes

- Never commit `.env` file to git
- Rotate `NEXTAUTH_SECRET` regularly in production
- Use Supabase Row Level Security (RLS) for additional security
- Store uploaded files in secure storage (S3/Supabase Storage)
- Enable database backups in Supabase

Support & Documentation

- Next.js Documentation (https://nextjs.org/docs)
- Prisma Documentation (https://www.prisma.io/docs)
- Supabase Documentation (https://supabase.com/docs)
- NextAuth.js Documentation (https://next-auth.js.org)

License

Private - The Get Group