Full-Stack Assignment: Financial Analytics Dashboard

Objective

Develop a full-stack financial application with dynamic data visualization, advanced filtering, and configurable CSV export functionality.

Assignment Requirements

Use Case

Financial analysts need to track and analyze company transactions with:

1. Interactive dashboard with visualizations
2. Transaction listing with search/sort/filter capabilities
3. Error handling via alert chips
4. Configurable CSV export downloads directly through user's browser

Figma Design

Link: here

User Flow

1. **Authentication**: User logs in with JWT authentication
2. **Dashboard Access**: View financial analytics with charts and transaction tables
3. **Data Interaction**: Filter, search, and sort transactions dynamically
4. **Report Generation**: Configure columns and export CSV reports
5. **File Download**: Download file directly once the report is generated

Core Features

1. Authentication & Security

- JWT-based login/logout system
- Secure API endpoints with token validation

2. Financial Dashboard

- **Visualizations**: Revenue vs expenses trends, category breakdowns, summary metrics
- **Transaction Table**: Paginated display with responsive design
- **Filtering**: Multi-field filters (Date, Amount, Category, Status, User)
- **Sorting**: Column-based sorting with visual indicators
- **Search**: Real-time search across transaction fields

3. CSV Export System

- **Column Configuration**: User selects which fields to export
- **Auto-download**: Automatic file download when ready

Technical Stack

Frontend

- **Framework**: React.js with TypeScript
- **State Management**: Use any preferred libraries or native libraries
- **Charts**: Chart.js or Recharts – or any other
- **UI Library**: Material-UI, Ant Design, or Chakra UI – or any other

Backend

- **Server**: Any NodeJS based framework with Typescript
- **Database**: MongoDB
- **Authentication**: JWT
- **File Processing**: CSV generation libraries

Expected Deliverables

1. Frontend Application

- Login interface with JWT handling
- Interactive dashboard with charts and transaction table
- Advanced filtering and search functionality
- CSV export configuration modal – Be creative, come up with UI which may be best for this functionality.

2. Backend Application

- RESTful APIs with proper authentication
- MongoDB integration with optimized queries
- CSV generation with configurable columns

3. Documentation

- **README**: Setup instructions, usage examples
- **API Documentation**: Endpoint specifications
- Proper CSV formatting with headers

Sample Data Structure

Sample data is provided with this assignment; you can use that to load into your dB instance.

PS: Use any coding/ai tool you might want to use