**Personal Accounting Manager with Partners (Electron + React + Node + MongoDB)**

**1. Tech Stack**

* **Frontend (UI)**: React.js (with Bootstrap 5, downloaded FontAwesome, Oswald font) inside Electron renderer.
* **Desktop Runtime**: Electron.js (main process, IPC for backend communication).
* **Backend (API)**: Node.js with Mongoose for MongoDB.
* **Database**: MongoDB (Local or Atlas).
* **Styling**: Black & white theme, Bootstrap 5, shadows on cards, Oswald headings.

**2. Features**

1. **Authentication**
   * Registration: username, email, password
   * Login: username, password
   * Password hashing with bcrypt.
   * JWT for sessions (stored in Electron secure storage).
2. **Accounts**
   * Create Account: accountName, balance
   * View all accounts with running balances.
3. **Transactions**
   * Types:
     + **Send**
       - Directly (sender = user)
       - On behalf of someone (onBehalfOf)
     + **Receive**
   * Fields:
     + senderName: default = login user
     + receiverName: required if send
     + quantity, rate, amount: amount = quantity × rate (editable manually)
     + account: dropdown (choose from user accounts)
     + date: default = today
     + onBehalfOf: optional partner
   * Every transaction has sequential ID starting from 1.
4. **Partners**
   * Add Partner: partnerName, phone
   * Track balances with each partner:
     + Who I need to pay
     + Who needs to pay me?
5. **Journal**
   * Chronological list of all transactions of logged-in user.
6. **Ledger**
   * Shows all accounts with debit/credit.
   * Tracks balances of:
     + Accounts
     + Partners (Pay / Receive)
7. **Dashboard**
   * Daily Summary
   * Monthly Summary
   * Yearly Summary
   * Cards:
     + Total Sent
     + Total Received
     + Net Balance
     + Who Needs to Pay Me
     + Whom I Need to Pay
8. **Reports**
   * Filter by Partner & Date (e.g., export transactions on behalf of *Ali* on today’s date).
   * Export to **PDF**:
     + List of transactions
     + Total Sent
     + Total Received
     + Difference

**3. Database Schemas (Mongoose)**

⚠️ IDs are String (not ObjectId).

**User**

{

\_id: String,

username: String,

email: String,

password: String, // hashed

createdAt: Date

}

**Account**

{

\_id: String,

userId: String,

accountName: String,

balance: Number,

createdAt: Date

}

**Partner**

{

\_id: String,

userId: String,

partnerName: String,

phone: String,

balance: Number, // Positive = owes me, Negative = I owe

createdAt: Date

}

**Transaction**

{

\_id: String,

trxId: Number, // sequential per user

userId: String,

type: String, // "send" | "receive"

senderName: String,

receiverName: String,

quantity: Number,

rate: Number,

amount: Number,

account: String, // accountId

date: Date,

onBehalfOf: String, // partnerId if exists

createdAt: Date

}

**4. Transaction Logic (On Behalf Of)**

Example:

* User: Ali
* Partner: Ahmad
* Ali sends 10 transactions **on behalf of Ahmad** to Abdullah.

👉 Effect:

* Ahmad becomes **creditor** (needs to pay Ali).
* When Ahmad pays back:
  + If he pays **more** → now Ali owes Ahmad (Ali becomes debtor).
  + If he pays **less** → Ahmad is still creditor.

Dashboard shows under:

* **Who needs to pay me** (Creditors)
* **Whom I need to pay** (Debtors)

**5. Frontend Layout**

**Sidebar (Dashboard Layout)**

* Dashboard
* Accounts
* Transactions
* Journal
* Ledger
* Partners
* Reports/Export

**Dashboard Page**

* Cards:
  + Total Sent
  + Total Received
  + Net Balance
  + Who needs to pay me (list)
  + Whom I need to pay (list)
* Filters: Daily / Monthly / Yearly

**Transactions Page**

* Form (Bootstrap modal)
* Transaction Table with shadowed cards.

**Journal Page**

* List of transactions (table).

**Ledger Page**

* Per account → Debit, Credit, Balance.
* Per partner → Balances.

**Partners Page**

* Add Partner Form.
* List partners with current balance.

**Reports Page**

* Filter by Partner + Date.
* Export to PDF.

**6. Export to PDF**

* Library: pdfkit or jspdf.
* Example export:
  + **Transactions on behalf of Ali (14 Sept 2025)**:
    - Table of transactions
    - Total Sent
    - Total Received
    - Difference

**7. Styling Rules**

* **Theme**: Black background, White text, Bootstrap cards with shadows.
* **Font**: Oswald for headings.
* **Icons**: FontAwesome (downloaded, offline).
* **UX**: Simple, formal, accounting-style.

**8. App Flow**

1. User registers & logs in.
2. Creates accounts and partners.
3. Logs transactions (send/receive, direct or on behalf).
4. Journal stores chronological record.
5. Ledger calculates per-account & per-partner balances.
6. Dashboard shows summaries (daily/monthly/yearly).
7. Reports can export transactions on behalf of a specific partner with totals.