Perfect — you're now aiming to build a **real-life, end-to-end accounting solution** for **small businesses**. That’s an ambitious and valuable project, and I’ll help you structure it in a way that's both technically solid and business-friendly.

**🧭 Goal Recap**

Build a **complete accounting software** that:

* Works for **non-accountants** (small shopkeepers, freelancers, etc.)
* Automates journals, ledgers, trial balance, and reports
* Supports real-life business scenarios (sales, purchases, expenses, etc.)
* Is flexible, modular, and extensible
* Is easy to use and mobile/desktop friendly

**🏗️ High-Level Architecture**

**🎯 Modules**

| **Module** | **Description** |
| --- | --- |
| Chart of Accounts | Dynamic, customizable accounts |
| Sales & Invoicing | Create & send sales invoices, auto-post journal |
| Purchases | Record purchase bills, track payables |
| Payments & Receipts | Handle money in/out, cash, bank |
| Journals | Auto-generated (from docs), manual override optional |
| Ledgers | Account-wise transactions |
| Trial Balance | Full debits/credits per account |
| Financial Reports | P&L, Balance Sheet, Cash Flow |
| User & Role Mgmt | Different access for staff, owners |
| Settings & Templates | Customize categories, GST/VAT, currencies, templates |

**🗃️ Suggested Database Schema (Core)**

**1. Accounts (Chart of accounts)**

sql

CopyEdit

Id INT PK

Code NVARCHAR(20)

Name NVARCHAR(100)

Type ENUM('Asset', 'Liability', 'Equity', 'Revenue', 'Expense')

ParentId INT FK -> Accounts(Id)

**2. JournalEntries (Transaction header)**

sql

CopyEdit

Id INT PK

EntryDate DATE

Description TEXT

Reference VARCHAR(50)

CreatedBy VARCHAR(50)

CreatedAt DATETIME DEFAULT GETDATE()

**3. JournalEntryLines (Debit/credit details)**

sql

CopyEdit

Id INT PK

JournalEntryId INT FK -> JournalEntries

AccountId INT FK -> Accounts

Debit DECIMAL(18,2)

Credit DECIMAL(18,2)

Narration TEXT

**4. Customers, Suppliers, Invoices, Receipts, Payments, etc.**

Modular tables like:

* SalesInvoices
* PurchaseInvoices
* Receipts
* Payments

These are what the **user actually sees/uses**. Behind the scenes, they generate journal entries.

**🧠 Real-Life Examples You Must Support**

| **Scenario** | **Documents** | **Journal Effect** |
| --- | --- | --- |
| Sell a product on cash | Invoice | Dr: Cash, Cr: Sales Revenue |
| Buy office supplies on credit | Bill | Dr: Office Supplies, Cr: Accounts Payable |
| Pay electricity bill | Payment | Dr: Electricity Expense, Cr: Cash/Bank |
| Receive money from customer | Receipt | Dr: Cash, Cr: Accounts Receivable |
| Owner invests capital | Entry | Dr: Cash, Cr: Capital |

All of these can be handled dynamically — but behind the scenes, each triggers a **balanced journal entry**.

**📊 Reports You Should Include**

* Trial Balance
* General Ledger (per account)
* Day Book (chronological journal)
* Profit & Loss Statement
* Balance Sheet
* Cash/Bank Book
* GST/VAT Report (if applicable)
* Customer/Supplier Balance

**🧱 Tech Stack Suggestion**

Since you're a **.NET Core developer**, here's a suggested stack:

| **Layer** | **Tool/Framework** |
| --- | --- |
| Backend API | ASP.NET Core Web API |
| ORM | EF Core (Code First) |
| DB | SQL Server or PostgreSQL |
| Frontend (web) | Blazor (for internal) or React/Angular (for UI-rich SPA) |
| Desktop (optional) | WinForms/WPF/.NET MAUI |
| Mobile (optional) | .NET MAUI or Flutter |
| Reporting | Stimulsoft, RDLC, or export to Excel/PDF |
| Authentication | ASP.NET Identity + Roles |
| Hosting | Azure / VPS / On-prem |

**📦 Business Features to Add Later**

* Inventory tracking
* Payroll (basic salary & expense reimbursement)
* Tax compliance (GST, TDS, VAT etc.)
* Recurring invoices or expenses
* Notifications (e.g., due payments)
* Multi-user / multi-business support
* Export/import (Excel, Tally JSON, etc.)

**✅ Your Next Steps**

Let’s start **practically**, module-by-module.

Here’s a good order:

1. 🔑 Set up Accounts, JournalEntries, JournalEntryLines
2. 🧾 Add SalesInvoice and auto-generate journal from it
3. 💵 Add Payments and Receipts
4. 📊 Build Ledger, Trial Balance, and Day Book
5. 🧪 Add validations (e.g., journal must be balanced)
6. 👤 Add basic user authentication and permissions

-- Chart of Accounts Table

CREATE TABLE Accounts (

Id INT IDENTITY(1,1) PRIMARY KEY,

Code NVARCHAR(20) NOT NULL UNIQUE,

Name NVARCHAR(100) NOT NULL,

Type NVARCHAR(20) NOT NULL CHECK (Type IN ('Asset', 'Liability', 'Equity', 'Revenue', 'Expense')),

ParentId INT NULL,

CONSTRAINT FK\_Accounts\_Parent FOREIGN KEY (ParentId) REFERENCES Accounts(Id)

);

-- Journal Entry Header Table

CREATE TABLE JournalEntries (

Id INT IDENTITY(1,1) PRIMARY KEY,

EntryDate DATE NOT NULL,

Description NVARCHAR(255),

Reference NVARCHAR(50),

CreatedBy NVARCHAR(50),

CreatedAt DATETIME DEFAULT GETDATE()

);

-- Journal Entry Lines Table

CREATE TABLE JournalEntryLines (

Id INT IDENTITY(1,1) PRIMARY KEY,

JournalEntryId INT NOT NULL,

AccountId INT NOT NULL,

Debit DECIMAL(18, 2) DEFAULT 0,

Credit DECIMAL(18, 2) DEFAULT 0,

Narration NVARCHAR(255),

FOREIGN KEY (JournalEntryId) REFERENCES JournalEntries(Id),

FOREIGN KEY (AccountId) REFERENCES Accounts(Id)

);

INSERT INTO Accounts (Code, Name, Type) VALUES

('1000', 'Cash', 'Asset'),

('1100', 'Bank', 'Asset'),

('1500', 'Equipment', 'Asset'),

('2000', 'Accounts Payable', 'Liability'),

('3000', 'Capital', 'Equity'),

('4000', 'Sales Revenue', 'Revenue'),

('5000', 'Office Supplies Expense', 'Expense');

-- Insert Journal Entry Header

INSERT INTO JournalEntries (EntryDate, Description, Reference, CreatedBy)

VALUES ('2025-08-01', 'Purchased laptop for office use', 'INV-001', 'admin');

-- Get the ID of the inserted journal entry

DECLARE @JournalEntryId INT = SCOPE\_IDENTITY();

-- Insert Journal Lines (Double Entry)

INSERT INTO JournalEntryLines (JournalEntryId, AccountId, Debit, Credit, Narration)

VALUES

(@JournalEntryId, (SELECT Id FROM Accounts WHERE Code = '1500'), 1000.00, 0.00, 'Laptop purchase'), -- Equipment

(@JournalEntryId, (SELECT Id FROM Accounts WHERE Code = '1000'), 0.00, 1000.00, 'Cash paid'); -- Cash