

Project Summary & Plan of Action: Cloud-Based Multi-Branch Bakery ERP in VB.NET

1. Project Overview

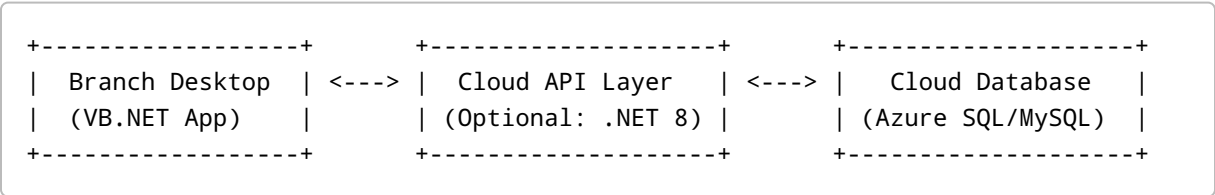
- **Application Type:** Full-stack desktop application in **VB.NET**
- **Key Modules:**
 - Warehousing / Stock Room
 - Manufacturing
 - Retail (POS)
 - Accounting
- **Multi-branch Support:** Yes
- **Backend:** Cloud-based (no local backend)

2. Cloud Backend Recommendation

Option	Pros	Cons	Estimated Cost (July 2025)
Azure SQL Database	Seamless Visual Studio integration, scalable, secure	Monthly cost applies	\~R90-R150/month (Basic Tier)
PlanetScale (MySQL)	Zero maintenance, serverless, free tier	Free tier limits, newer tech	Free to \~R180/month
Amazon RDS / DigitalOcean	Scalable, predictable costs	Slightly more setup	\~R280-R550/month

Recommendation: Start with **Azure SQL Database (Basic Tier)** for smooth integration with Visual Studio and .NET.

3. Suggested Architecture



- Option A: Direct connection to Cloud DB (faster to build)
- Option B: Connect via API (better security and future scalability)

4. Modular Build Approach (Recommended)

Build the system in the following modular order:

1. Warehousing / Stock Room Module:

2. Manage raw materials, inventory, and stock levels
3. Generate **Purchase Orders (POs)** for suppliers or Manufacturing

4. Manufacturing Module:

5. Track production, batch management, and work-in-progress
6. Generate **Transfer Orders** for Retail (finished goods)

7. Retail (POS) Module:

8. Sales, invoicing, stock decrement from Manufacturing
9. Cash management and shift closure

10. Accounting Module (In-App Custom Build):

11. Capture all financial transactions from other modules
12. Generate financial reports, manage journals, perform reconciliations

Key Interactions:

- Warehousing → Manufacturing (Stock Requests/POs)
 - Manufacturing → Retail (Goods Transfer)
 - Retail → Accounting (Sales & Revenue)
 - All → Accounting (Journals & Ledgers)
-

5. Accounting Module Design (Custom In-App)

Recommendation: Build your own accounting module **inside the app** with features inspired by **Pastel** for full control, simplicity, and cost savings.

Key Features:

- **General Ledger:** Chart of accounts, double-entry journal system
- **Stock Management:** FIFO, LIFO, weighted average, real-time updates
- **Accounts Payable (Creditors):** Supplier management, invoices, payments, aged analysis

- **Accounts Receivable (Debtors):** Customer accounts, invoicing, receipts, aged analysis
- **Bank Reconciliation:** CSV statement import, match & reconcile
- **Financial Reporting:** Trial Balance, Income Statement, Balance Sheet
- **Tax/VAT Compliance:** Automated calculations (optional)

User-Friendly Design:

- Tabbed navigation for ledgers, banking, reporting
- Always-visible balances (like Pastel)
- Printable reports (Crystal or FastReport)

6. Pastel Accounting Integration (Optional)

Method	How it Works	Tools Required	Cost Estimate (ZAR)
CSV/Excel	Export transactions, import into Pastel	Simple, manual	Free
ODBC	Direct read/write to Pastel DB	Pastel + ODBC driver	Included in Pastel license
Pastel SDK/API	Automated integration	SDK license	\~R8,000+ one-time + annual fee

Recommendation: Start with **CSV/Excel** integration.

7. Tools & Costs Summary

Component	Estimated Cost (ZAR)
Azure SQL Database	\~R90–R150/month
Azure Web/API Hosting	\~R90/month (optional)
Visual Studio Community	Free
Crystal Reports Runtime	Free
Crystal Reports Full	\~R6,000 once-off (optional)
FastReport.NET	\~R3,500 once-off (optional)
Power BI Pro (optional)	\~R180/user/month
Pastel Partner (optional)	\~R5,000+ once-off
Pastel SDK (optional)	\~R8,000+ once-off + fees

8. Estimated Total Monthly Cost (Minimum Setup)

Item	Monthly Cost (ZAR)
Azure SQL Database	R150
Azure Web/API (optional)	R90
Visual Studio	Free
Crystal Reports (runtime)	Free
Total Monthly Cost	R240/month

(Optional tools like Power BI, full Crystal Reports, or Pastel would add extra costs if used.)

9. Plan of Action

Phase	Task	Deliverable
Phase 1	Set up Azure SQL Database & design database schema	Cloud database ready
Phase 2	Build Warehousing / Stock Module	Purchase Order generation, stock management
Phase 3	Build Manufacturing Module	Production tracking, transfer to retail
Phase 4	Build Retail (POS) Module	Sales, invoicing, daily reports
Phase 5	Build Accounting Module (custom)	Financials, ledgers, reports
Phase 6	Implement Security, Role Management, and Reporting	User access controls & reports
Phase 7	Testing, Deployment, and Training	Live system with documentation

End of Summary & Action Plan