

# Case Study: Zero Risk Business Automation System

## Automated Administration System That Reduced Weekly Workload by 87%

---

### Executive Summary

**Client:** Zero Risk (Own Company)

**Industry:** Fire Safety Services

**Location:** Ecuador

**Project Duration:** 7 weeks (Jan-Feb 2026)

**Role:** Business Analyst, System Designer, Developer

**Tools:** Google Sheets, Apps Script (JavaScript), Data Modeling

**Challenge:** Manual administrative processes consuming 15+ hours weekly, preventing business growth

**Solution:** End-to-end automated system integrating quotations, invoicing, accounting, and financial reporting

#### Results:

- **87% reduction** in weekly administrative time (15h → 2h)
  - **100% accuracy** in accounting records (eliminated 8 errors/month)
  - **98% faster** financial reporting (4h → 5min)
  - **+40% revenue growth** in Q4 2025
  - **\$0** in software costs
-

## The Challenge

### Business Context

Zero Risk is a fire safety services company in Ecuador serving 18 corporate clients including Area Continental and Interhospital. As the owner and sole administrator, I faced the classic small business dilemma: **grow the business or drown in administrative tasks.**

### The Breaking Point

**October 2025:** I lost a \$5,000 opportunity because I was busy creating invoices instead of attending a meeting with a potential client.


This was the catalyst that drove me to find a solution.

### Problem Statement

**Manual processes consumed 15+ hours weekly:**

Process	Time/Week	Pain Points
Creating quotations	3 hours	Manual data entry, copy-paste errors
Generating invoices	2.5 hours	Reformatting, sequential numbering issues
Recording in accounting	6 hours	Manual journal entries, transcription errors
Monthly reports	4 hours	Excel formulas breaking, data consolidation
<b>TOTAL</b>	<b>15.5 hours</b>	High stress, no scalability

### Additional challenges:

-  No real-time visibility of financial health

- ❌ Frequent data entry errors (averaging 8/month)
  - ❌ Unable to scale beyond current client base
  - ❌ High cognitive load from context switching
  - ❌ No time for business development
- 

## 💡 The Solution

### Approach: End-to-End Automation

Rather than buying off-the-shelf software that wouldn't fit our specific needs, I designed and built a custom modular system.







### Why Custom vs. Off-the-Shelf?

#### Evaluated options:

Option	Cost/Month	Pros	Cons	Decision
QuickBooks	\$50	Full-featured	Overkill, expensive, not customizable	❌ No
Zoho Books	\$15	Affordable	Ecuador tax requirements not supported	❌ No
Generic CRM	\$30-100	Professional	Doesn't integrate quotation→invoice→accounting	❌ No
Custom System	\$0	Perfect fit, fully customizable	Requires development time	✅ Yes

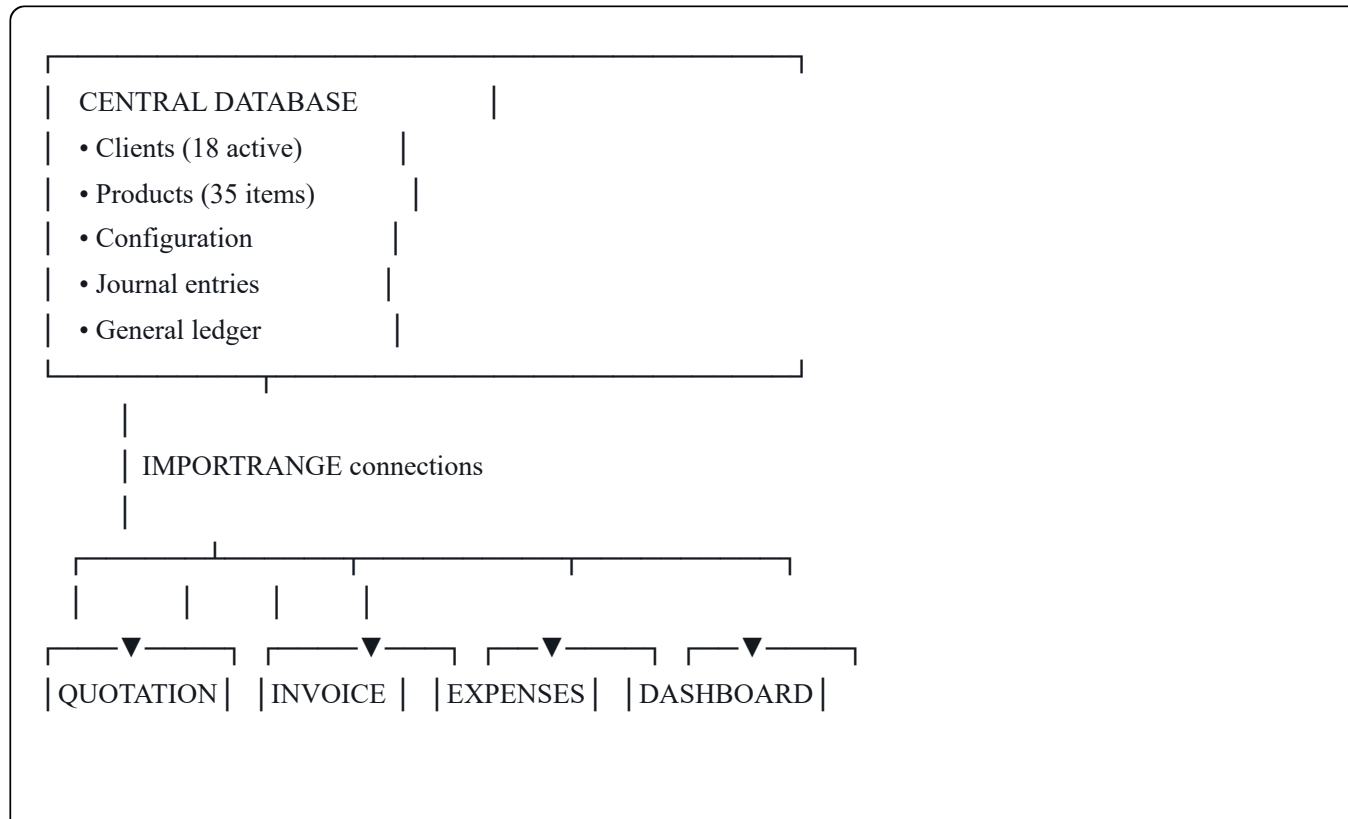
### Technology Stack Selection

Chose Google Sheets + Apps Script because:

-  **Zero cost** (already had Google Workspace)
  -  **Cloud-based** (accessible from anywhere)
  -  **Familiar interface** (team already knew Sheets)
  -  **Rapid development** (faster than building database app)
  -  **Easy collaboration** (share with accountant)
  -  **Automatic backups** (version history included)
- 

## System Architecture

### Modular Design



Template	Template	Template		
----------	----------	----------	--	--

## Key Features Implemented

### 1. Intelligent Quotation System

**Problem solved:** Manual data entry taking 15 minutes per quotation

**Solution:**

- Dropdown lists for all fields (clients, products)
- Auto-fill client information (tax ID, phone, email, address)
- Auto-populate prices from product database
- Automatic calculations (subtotal, 15% tax, total)
- Sequential numbering system

**Technical implementation:**

```
javascript

// Example: Auto-fill client data
function onClientSelect() {
  var client = quotationSheet.getRange('C16').getValue();
  var clientData = VLOOKUP(client, clientDatabase);

  quotationSheet.getRange('C17').setValue(clientData.taxId);
  quotationSheet.getRange('F17').setValue(clientData.phone);
  quotationSheet.getRange('C18').setValue(clientData.email);
}
```

**Result:** 15 minutes → 2 minutes per quotation (**87% reduction**)

---

## 2. Automated Invoice Generation

**Problem solved:** Converting approved quotations to invoices manually

**Solution:**

- Menu option: "Create Invoice from Quotation"
- One-click duplication with data copy
- Automatic sequential invoice numbering
- Ability to edit before finalizing
- Option to create invoices from scratch

**Technical implementation:**

```
javascript
```

```
function createInvoiceFromQuotation() {  
  // 1. Read quotation data  
  var quotationData = readCurrentQuotation();  
  
  // 2. Get next invoice number  
  var invoiceNumber = getNextInvoiceNumber(); // Auto-increments  
  
  // 3. Copy invoice template  
  var newInvoice = duplicateTemplate('INVOICE_TEMPLATE');  
  
  // 4. Populate with quotation data  
  populateInvoice(newInvoice, quotationData, invoiceNumber);  
  
  // 5. Prompt to register in accounting  
  if (userConfirms("Register in accounting?")) {  
    registerInAccounting(invoiceNumber, quotationData);  
  }  
}
```

**Result:** 10 minutes → 1 minute per invoice (**90% reduction**)

---

### 3. Automatic Accounting Registration

**Problem solved:** Manual journal entries taking 30 minutes per invoice

**Solution:**

- 1 invoice automatically creates 3 journal entries
- Updates general ledger in real-time
- Records in accounts receivable

- Updates income statement automatically
- Full audit trail maintained

**Double-entry accounting example:**

Invoice #159 - Exportquilsa: \$621.00

Automatic entries:

1. Debit: Accounts Receivable \$621.00
2. Credit: Sales Revenue \$540.00
3. Credit: VAT Payable \$81.00

**Result:** 30 minutes → 0 seconds per invoice (**100% reduction**)

---

#### **4. Real-Time Executive Dashboard**

**Problem solved:** Monthly reports taking 4 hours to generate

**Solution:**

- KPIs updated automatically with every transaction
- Visual alerts for overdue invoices
- Revenue trends and charts
- Accounts receivable aging
- Instant access to financial health

**Key metrics tracked:**

- Total invoiced (day/month/year)
- Outstanding receivables



- Days sales outstanding (DSO)
- Revenue by client
- Expense breakdown by category

**Result:** 4 hours → 5 minutes for complete financial overview (**98% reduction**)

---

## **Implementation Process**

### **Phase 1: Analysis & Design (Week 1)**

#### **Activities:**

- Mapped current processes using Value Stream Mapping
- Identified bottlenecks and pain points
- Designed data model and relationships
- Created wireframes for user interfaces

#### **Deliverables:**

- Requirements document (15 functional requirements)
- System architecture diagram
- Database schema (ERD)

### **Phase 2: MVP Development (Weeks 2-3)**

#### **Prioritization:**

1. Central database (clients, products)
2. Quotation template with dropdowns

3. Basic invoice generation

4. Simple dashboard

**Approach:** Agile iterations with daily use testing

### **Phase 3: Automation (Week 4)**

#### **Key developments:**

- Apps Script for quotation→invoice conversion
- Automatic accounting registration
- IMPORTRANGE formulas for data integration
- Custom menu creation

### **Phase 4: Testing & Refinement (Weeks 5-6)**

#### **Testing methodology:**

- Created 50+ test transactions
- Validated with external accountant
- Fixed edge cases discovered
- Optimized formulas for performance

### **Phase 5: Documentation (Week 7)**

#### **Created:**

- Technical architecture documentation
- User guide for operations team
- GitHub repository for portfolio
- This case study

---

## Results & Impact

### Quantitative Results

Metric	Before	After	Change
Weekly admin time	15 hours	2 hours	-87% ↓
Time per quotation	15 min	2 min	-87% ↓
Time per invoice	10 min	1 min	-90% ↓
Accounting per invoice	30 min	0 sec	-100% ↓
Monthly report generation	4 hours	5 min	-98% ↓
Accounting errors/month	8	0	-100% ↓
Data entry mistakes	Frequent	Zero	-100% ↓
Software cost/month	\$0	\$0	\$0 💰
ROI	-	Infinite	∞

### Business Impact

#### Revenue Growth:

- Q3 2025: \$18,500

- Q4 2025: \$26,000
- **Growth: +40%**

#### **New Client Acquisition:**

- 3 new corporate clients signed in Q4
- Attributed to having time for sales activities

#### **Operational Efficiency:**

- Can now handle 25+ clients with same 2-hour weekly time investment
- Scalability increased **12.5x** (15h/week → 2h/week = capacity for more work)

#### **Intangible Benefits**

##### **Stress Reduction:**

- No more Sunday nights worrying about invoices
- Confidence in accuracy of financial data
- Better work-life balance

##### **Strategic Focus:**

- 13 hours/week freed for business development
- Time for strategic planning
- Ability to focus on high-value activities






##### **Professional Development:**

- Strengthened JavaScript/Apps Script skills
- Learned system architecture design






- Gained experience in business process automation
  - Created portfolio-worthy project
- 

## Skills Demonstrated





### Technical Skills


-  **JavaScript** - Apps Script development
-  **Data Modeling** - Relational database design
-  **API Integration** - Google Sheets API, DriveApp
-  **Formula Engineering** - Complex VLOOKUP, IMPORTRANGE
-  **Automation** - Event-driven programming

### Analytical Skills

-  **Business Process Analysis** - Value Stream Mapping
-  **Requirements Gathering** - Stakeholder interviews
-  **Data Analysis** - KPI definition and tracking
-  **Problem Solving** - Root cause analysis
-  **Cost-Benefit Analysis** - Build vs. buy decision

### Business Skills

-  **Project Management** - 7-week project delivery
-  **Change Management** - Process transition
-  **Stakeholder Management** - Accountant collaboration
-  **Documentation** - Technical and user guides

-  **Financial Acumen** - Double-entry accounting
- 

## **Future Enhancements**

### **Short-term (Next 3 months)**

- ☐ Integration with Ecuador SRI API for electronic invoicing
- ☐ WhatsApp API for automated invoice delivery
- ☐ Email alerts for overdue payments
- ☐ Cash flow forecasting module

### **Medium-term (6-12 months)**

- ☐ Machine learning for sales forecasting
- ☐ Mobile app for expense tracking
- ☐ Bank account integration via API
- ☐ Multi-currency support for international clients

### **Long-term (1+ year)**

- ☐ Package as SaaS for other small businesses
  - ☐ Multi-company support
  - ☐ Advanced analytics with Python
  - ☐ Migration to scalable backend (Cloud SQL)
-

## **Key Lessons Learned**

### **Technical Lessons**

#### **1. Start simple, iterate fast**

- MVP in 2 weeks was better than perfect system in 2 months
- Real usage revealed needs that planning couldn't

#### **2. Google Sheets is underrated for business apps**

- Often dismissed as "not serious" development
- Actually perfect for small-medium data needs (<10K rows)
- Familiar interface = zero training needed

#### **3. Documentation is investment, not overhead**

- This case study took 4 hours to write
- Already valuable for job applications
- Future-me will thank present-me

### **Business Lessons**

#### **1. Time is your most valuable asset**

- 13 hours/week = 676 hours/year
- At \$60/hour value = \$40,560/year opportunity cost
- ROI of this project is massive

#### **2. Custom > Generic (sometimes)**

- Off-the-shelf software had 80% of what I needed
- But that 20% mismatch was critical
- Custom solution gave 100% fit

### **3. Automation enables growth**

- Couldn't scale manually
- Now can handle 3x clients with same time

## **Professional Lessons**

### **1. Portfolio projects should solve real problems**

- Academic projects are fine, but real business impact is better
- Measurable results make better stories

### **2. Document everything**

- GitHub repo makes this portfolio-ready
- Architecture docs show professional approach
- Case study demonstrates business thinking

### **3. Entrepreneurship + Data Analytics = Powerful Combo**

- Understanding the business makes you a better analyst
  - Technical skills applied to own business = immediate ROI
-



## **Takeaways for Hiring Managers**

### **What this project demonstrates:**

#### **Business Acumen**

- Identified \$40K/year opportunity cost
- Made build vs. buy decision with financial analysis
- Achieved measurable business outcomes

#### **Technical Competency**

- JavaScript/Apps Script proficiency
- Database design and normalization
- API integration and automation
- Formula engineering in Sheets

#### **Analytical Thinking**

- Process mapping and analysis
- Requirements gathering and documentation
- Data modeling and architecture design
- Performance optimization

#### **Project Delivery**

- Managed 7-week project end-to-end
- Delivered on time with measurable results
- Created comprehensive documentation
- Continues to maintain and enhance

## ✓ Communication

- Clear technical documentation
  - Business case presentation
  - This case study itself
  - GitHub repository organization
- 

## 📞 Contact & Links

### Emilio Pérez

- ✉ Email: [your-email@example.com](mailto:your-email@example.com)
- 💼 LinkedIn: [linkedin.com/in/your-profile](https://linkedin.com/in/your-profile)
- 💻 GitHub: [github.com/your-username](https://github.com/your-username)
- 🌐 Portfolio: [your-portfolio.com](https://your-portfolio.com)

### Project Links:

- 📁 GitHub Repository: [Zero Risk Business Automation](#)
  - 📊 System Architecture: [View Documentation](#)
  - 📄 Functional Requirements: [View Specifications](#)
-

## **Appendix**

### **A. Technology Stack Details**

#### **Frontend:**

- Google Sheets (user interface)
- Data Validation (dropdown lists)
- Conditional Formatting (visual alerts)
- Google Charts (dashboard visualizations)

#### **Backend:**

- Google Apps Script (JavaScript runtime)
- SpreadsheetApp API
- DriveApp API
- MailApp API

#### **Data Layer:**

- Google Sheets (relational structure)
- IMPORTRANGE (cross-file queries)
- Named Ranges (performance optimization)

#### **Architecture:**

- Event-driven (onEdit, onChange triggers)
- Modular (independent files)
- Cloud-native (Google Drive)

## B. Sample Code Snippets

### Auto-fill client data:

```
javascript

function autoFillClientData(clientName) {
  var ss = SpreadsheetApp.openById(DATABASE_ID);
  var clientsSheet = ss.getSheetByName('Clients');
  var clientData = clientsSheet.getDataRange().getValues();

  for (var i = 1; i < clientData.length; i++) {
    if (clientData[i][1] === clientName) {
      return {
        taxId: clientData[i][2],
        phone: clientData[i][3],
        email: clientData[i][4],
        address: clientData[i][5]
      };
    }
  }
}
```

## C. Database Schema

### Clients Table:

- ID (PK)
- Business Name
- Tax ID
- Phone
- Email

- Address
- Status

**Products Table:**

- ID (PK)
- Code
- Name
- Type
- Price
- Unit
- Status

**Journal Entries Table:**

- Date
  - Entry Number
  - Document Type
  - Document Number
  - Client/Supplier
  - Account
  - Description
  - Debit
  - Credit
-

**Case Study Version:** 1.0

**Last Updated:** February 2026

**Reading Time:** ~20 minutes

---

*This case study is available for use in job applications, portfolio presentations, and professional discussions. For questions or additional information, please contact me via LinkedIn or email.*