

FINANCIAL PLAYBOOK - MASTER PLAYBOOK DOCUMENT

Complete Project Specification & Development Guide

Project Name: Financial Playbook

Owner: Superior Networks LLC

Developer: Manus AI Platform

Created: November 24, 2025

Version: 1.0 (Master Consolidated)

GitHub Repository: <https://github.com/SuperiorNetworks/financial-playbook>

Dev Server: <https://3000-i58gxuxgsw3xb0fmqnp19-1807f537.manusvm.computer>

TABLE OF CONTENTS

1. [Executive Summary](#)
 2. [Value Proposition & Business Case](#)
 3. [Complete Feature List](#)
 4. [Technical Architecture](#)
 5. [Wireframes with QC Numbers](#)
 6. [External Integrations](#)
 7. [Visual Design System](#)
 8. [UI Implementation Plan](#)
 9. [Development Plan](#)
 10. [Security & Compliance](#)
 11. [Deployment Strategy](#)
 12. [Future Roadmap](#)
-

EXECUTIVE SUMMARY

What is Financial Playbook?

Financial Playbook is a single-user web application that validates QuickBooks Online data against raw bank statements, highlights discrepancies with an intelligent visual status system, and provides forward-looking scenario planning to help small business owners make confident financial decisions.

The Core Problem

Small business owners can't fully trust their QuickBooks data due to data entry errors, timing mismatches, and incomplete information. This lack of confidence leads to missed opportunities, poor decisions, wasted time, and constant anxiety about cash flow.

The Solution

Financial Playbook provides three critical layers:

1. **Independent Verification** - Upload bank statements to create a second source of truth
2. **Intelligent Reconciliation** - Visual traffic light system () shows exactly where QuickBooks and reality diverge
3. **Forward-Looking Intelligence** - AI-powered scenario planning simulates the future based on validated data

Key Value Proposition

"Organize the chaos of business by providing a single source of truth that validates QuickBooks data against raw bank statements, highlights discrepancies, and gives owners the insights they need to make confident decisions - saving time and money to improve all aspects of the business."

ROI

- **Time Savings:** 4 hours/week → 15 minutes/week = 3.75 hours saved

- **Annual Value:** \$14,000-15,000 (time + error prevention + better decisions)
 - **Annual Cost:** \$600-1,200 (estimated subscription)
 - **ROI:** 10-25x return on investment
-

VALUE PROPOSITION & BUSINESS CASE

The Problem: Organized Chaos in Small Business Finance

Small business owners face a critical challenge: they can't fully trust their own financial data. Even with QuickBooks Online, the reality is messy:

- **Data entry errors** - Transactions get miscategorized or missed entirely
- **Timing mismatches** - What's in QuickBooks doesn't match what's in the bank
- **Incomplete picture** - QuickBooks shows what happened, not what's about to happen
- **No early warning system** - By the time you see a problem, it's too late
- **Decision paralysis** - Without confidence in the data, owners hesitate to make critical decisions
- **Time drain** - Hours spent reconciling, double-checking, and worrying

The Real Cost

When you can't trust your financial data:

- You miss opportunities because you think you don't have the cash
- You overspend because you think you have more than you do
- You can't plan ahead because you don't know what's coming
- You lose sleep worrying about the unknown
- You waste time that should be spent growing your business

How Financial Playbook Solves This

Layer 1: Independent Verification

Problem: You’re not 100% confident in your QuickBooks data

Solution: Upload bank statements and let AI extract the raw transaction data

- OCR reads bank statements (PDF, images)
- AI extracts every transaction with dates and amounts
- Math validation ensures accuracy
- Creates an independent record of what actually happened

Benefit: You now have two sources of truth to compare

Layer 2: Intelligent Reconciliation

Problem: You don’t know where QuickBooks and reality diverge

Solution: Visual status system shows exactly where things don’t match

-  **GREEN:** QuickBooks matches bank statements perfectly - trust this data
-  **YELLOW:** Minor variance detected - worth a quick review
-  **RED:** Critical mismatch - action required immediately

Benefit: Instant visibility into data quality without manual reconciliation

Layer 3: Forward-Looking Intelligence

Problem: QuickBooks shows the past, but you need to plan the future

Solution: Scenario planning and simulation engine

- Build “what-if” scenarios based on validated historical data
- Simulate day-by-day cash flow for the next 30, 60, 90 days
- AI flags risks before they become problems
- Test different strategies without real-world consequences

Benefit: Make decisions with confidence based on accurate projections

Real-World Use Cases

Use Case 1: The Missing Bill

Scenario: You have a \$5,627 loan payment due on Feb 15, but it's not entered in QuickBooks.

Without Financial Playbook:

- You don't notice until Feb 14
- Scramble to move money around
- Possible late fee or overdraft

With Financial Playbook:

-  RED alert on Feb 10: "Critical: Missing bill - Miamisburg Loan \$5,627.01 due in 5 days"
- Click for details: "No matching bill found in QuickBooks. Add now or adjust playbook."
- You add the bill to QuickBooks immediately
- Crisis averted

Value: \$50-100 saved in late fees + stress avoided

Use Case 2: The Price Variance

Scenario: Your Connectwise bill is usually 1,683.88, but this month it's 1,700.00.

Without Financial Playbook:

- You pay the bill without noticing
- Over a year, small increases add up
- No visibility into vendor pricing trends

With Financial Playbook:

-  YELLOW alert: "Price variance detected - Connectwise bill increased \$16.12 (0.96%)"
- Click for details: "Expected: 1,683.88 | Actual :1,700.00 | View in QuickBooks"

- You investigate and discover an unnecessary add-on
- You remove it and save \$200/year

Value: \$200/year saved + awareness of vendor pricing

Use Case 3: The Overdue Invoice

Scenario: Customer owes you \$5,200 on Invoice #1234, due 45 days ago.

Without Financial Playbook:

- You’re too busy to follow up
- Customer forgets to pay
- You lose cash flow

With Financial Playbook:

-  RED alert: “Critical: Payment missing - Invoice #1234 (\$5,200) was due 3 days ago”
- Click for details: “Expected payment not found in QuickBooks register. Contact customer.”
- You send a polite reminder
- Customer pays within 2 days

Value: \$5,200 collected 40+ days earlier = improved cash flow

Use Case 4: The Growth Decision

Scenario: You’re considering hiring a new technician at \$4,500/month.

Without Financial Playbook:

- You guess based on current balance
- Fear of running out of cash holds you back
- Business growth stalls

With Financial Playbook:

- Create scenario: “Add \$4,500/month expense starting March 1”

- Simulation shows: “90-day runway maintained, no overdraft risk”
- AI recommends: “Proceed with hire. Consider delaying equipment purchase to Q2.”
- You hire with confidence

Value: Business growth + peace of mind

COMPLETE FEATURE LIST

Completed Features (3)

1. **Project Initialization** - Next.js, TypeScript, Tailwind CSS, PostgreSQL, Drizzle ORM
2. **GitHub Repository** - Version control and deployment pipeline
3. **Complete Documentation with QC Numbers** - All wireframes numbered for easy reference

Core Features (90+ in Development Plan)

1. Authentication & User Management

- **Google OAuth 2.0** - Secure login with Google accounts
- **Session Management** - JWT-based authentication
- **User Profile** - Basic profile management
- **Security Headers** - HTTPS, CSP, HSTS enforcement

2. QuickBooks Online Integration

- **OAuth 2.0 Connection** - Secure QuickBooks account linking
- **Real-Time Data Sync** - Webhook-based automatic updates
- **Sync on Login** - Automatic refresh when user logs in
- **Manual Sync Button** - Force refresh anytime with “Sync Now” button
- **Configurable Sync Frequency** - Hourly, 4-hour, 8-hour, or daily background sync

- **Last Synced Timestamp** - Always visible on Accounts Management page
- **READ-ONLY Data Policy** - QuickBooks data cannot be edited in app
- “**View in QuickBooks**” **Links** - Direct links to edit records at source

Data Synced from QuickBooks:

- Bank account balances (checking, savings)
- Credit card balances
- Bills and payables
- Accounts receivable (invoices)
- Transactions and register entries
- Vendor information
- Customer information

3. Accounts Management

- **Bank Account Display** - View all connected accounts from QuickBooks
- **Credit Card Display** - Track credit card balances
- **Bills & Payables** - Upcoming bills from QuickBooks
- **Accounts Receivable** - Outstanding invoices tracking
- **Connection Status** - Real-time QuickBooks sync status
- **Visual Status Indicators** - Green/Yellow/Red for each account
- **QuickBooks Connector Settings** - Configure connection in Settings page

4. AI-Powered Strategy Builder Wizard

- **Document Upload** - Drag-and-drop PDF, PNG, JPG (bank statements, financial docs)
- **OCR & Text Extraction** - Automatic data extraction from documents
- **AI Data Analysis** - Identify recurring income and expenses
- **Math Validation Layer** - All AI calculations verified by traditional programming
- **Transaction Categorization** - Automatic categorization of transactions
- **Recurring Bill Identification** - Detect patterns in expenses

- **Strategy Recommendations** - AI suggests 1-3 tailored playbook strategies
- **Auto-Playbook Creation** - Pre-populated scenarios from uploaded data
- **Discrepancy Flagging** - UI alerts when AI math validation fails

Recommended Document Uploads:

- Last 3-6 months of bank statements
- Last 3-6 months of financial statements
- Recent credit card statements
- Any other financial documents

5. Scenario Planning & Logic Simulator ★

- **Multiple Scenarios** - Create and compare different financial strategies
- **Visual Logic Editor** - Build conditional rules (IF/THEN logic)
- **Condition Builder** - Define triggers (date, balance threshold, transaction type)
- **Action Builder** - Define responses (flag risk, send alert, adjust projection)
- **Day-by-Day Simulation** - Step through time applying business rules
- **Event Processing** - Handle income, bills, and user-defined rules
- **Risk Flagging System** - Automatic detection of overdrafts, cash flow issues
- **Lock-In Capability** - Save and lock successful scenarios
- **What-If Analysis** - Test different financial decisions
- **Scenario Comparison** - Side-by-side comparison of multiple scenarios

Example Logic Rules:

- IF Chase Balance < \$8,000 THEN flag “Reserve Risk”
- ON Feb 15, 2026 THEN PAY Miamisburg Loan \$5,627.01
- IF Owner Draw > \$5,000 in single month THEN flag “High Withdrawal Warning”

6. Advanced Report Generator ★

- **AI-Powered Analysis** - Root cause analysis of financial risks
- **Solution Recommendations** - AI-generated action plans

- **Multi-Page Reports** - Executive summary, detailed registers, charts
- **Math Validation** - All report calculations double-checked by traditional code
- **Report Sections:**
 - Executive Summary
 - Critical Risks Identified
 - Root Cause Analysis
 - Recommended Solutions
 - Running Balance Register
 - Cash Flow Projections
 - Scenario Comparison Charts
- **Multiple Export Formats** - PDF, Word, Markdown, CSV
- **Interactive Charts** - Visual cash flow projections with Chart.js
- **Customizable Templates** - Save report preferences

7. Visual Status System (Traffic Light Design) [NEW]

-  **GREEN Status** - In sync, no issues, QuickBooks matches reality
-  **YELLOW Status** - Warnings (price variance, sync delay >24hrs, duplicates detected)
-  **RED Status** - Critical (missing bills within 5 days, overdue payments within 3 days, overdraft risk)
- **Info Icons** () - Click for detailed explanations
- **Explanation Modals** - Contextual help for every status indicator
- **Configurable Thresholds** - Customize warning/critical levels in Settings
- **Real-Time Updates** - Status changes immediately after sync
- **Applied Everywhere** - Consistent across all pages (Dashboard, Accounts, Scenarios, Reports)
- **Status on:** Account balances, bills, transactions, scenario items, playbook activities

Status Logic Rules:

- **GREEN:** QuickBooks transaction matches playbook expectation within tolerance
- **YELLOW:** Price variance exceeds threshold (default 3%), sync delayed >24 hours, duplicate detected
- **RED:** Expected activity within 5 days not found in QuickBooks, expected payment within 3 days missing from register, projected overdraft

8. Customizable Dashboard Layout [NEW]

- **Drag-and-Drop** - Reposition any widget anywhere on dashboard
- **Resizable Widgets** - Stretch or shrink to fit your needs
- **9 Available Widgets:**
 1. Active Alerts (critical/warning notifications)
 2. Cash Flow Chart (interactive projection)
 3. Quick Stats (balance, burn rate, AR)
 4. Upcoming Bills (next 30 days)
 5. Account Balances (all accounts at a glance)
 6. Recent Activity (latest transactions)
 7. Scenario Status (active scenarios)
 8. Calendar Preview (upcoming events)
 9. QuickBooks Sync Status (connection health)
- **Layout Persistence** - Your layout saves automatically to database
- **Layout Presets**
 - “Executive View” , “Detailed Analysis” , “Monitoring Mode”
- **Lock/Unlock Toggle** - Prevent accidental changes when locked
- **Reset to Default** - One-click restore to original layout
- **Mobile Responsive** - Adapts to tablet and phone screens (stacks vertically)
- **Visual Grid Guides** - See grid during drag operations
- **Snap-to-Grid** - Widgets align automatically
- **Undo/Redo** - Revert layout changes

Technical Implementation:

- React Grid Layout library
- 12-column responsive grid system
- Database-persisted user preferences

9. Google Calendar Integration [NEW]

- **Calendar View Page** - Full calendar showing all financial activities
- **Two-Way Sync** - Playbook ↔ Google Calendar
- **Automatic Event Creation** - Bills, payments, tasks appear as calendar events
- **Color-Coded Events** - Green/Yellow/Red matching status system
- **Event Types:**
 - Upcoming bills (with amount)
 - Expected payments (AR)
 - Delegated tasks (future: multi-user)
 - Scenario milestones
 - QuickBooks sync reminders
- **Calendar Reminders** - Alerts for critical activities (RED status items)
- **Category Filtering** - Filter by bills, income, tasks, delegations
- **Google Calendar API** - OAuth 2.0 integration
- **Calendar Settings** - Configure sync frequency and event types

10. Automation System

- **Task Status** - Manual, Automated, or Delegated (delegated deferred to multi-user phase)
- **Recurring Bill Automation** - Auto-detect and track recurring bills
- **Price Variance Alerts** - Email when bill amount changes (default 3% threshold)
- **Automation Settings Modal** - Configure automation rules per bill/transaction
- **Configurable Thresholds** - Customize alert sensitivity
- **Email Notifications** - Automatic alerts for critical events
- **Notification Settings** - Customize alert timings and recipients

- **Default Overdue Alert** - 3 days (configurable in Settings)

Automation Rules:

- Recurring bill detected → Auto-add to playbook
- Price variance > 3% → Send email alert
- Expected transaction missing > 3 days → Send overdue alert
- Sync fails → Immediate notification

11. Comprehensive Audit Trail

- **Immutable Logging** - All actions tracked permanently (append-only)
- **Before/After Snapshots** - See exactly what changed
- **Tracked Actions:**
 - Scenario creation/modification/deletion
 - Logic rule changes
 - QuickBooks sync events
 - User login/logout
 - Report generation
 - Settings changes
 - Document uploads
- **Filterable Reports** - Filter by date range, action type
- **CSV Export** - Download audit logs for compliance
- **Tamper Detection** - Checksums verify log integrity
- **Audit Trail Page** - Dedicated UI with search and filters
- **Database Triggers** - Automatic logging at database level

12. Feature Request System (Feedback Portal) NEW

- **Submit Ideas** - Users can submit feature requests with descriptions
- **Voting System** - Upvote features you want most
- **Status Tracking** - Submitted → Under Review → Planned → In Progress → Completed → Declined

- **Commenting** - Discuss features with other users and admin
- **Public Roadmap** - See what's planned for Q1 2026, Q2 2026, etc.
- **Admin Dashboard** - Review and manage all requests
- **Email Notifications** - Updates when features change status
- **Analytics** - Track trending categories and popular requests
- **My Requests View** - Users track their own submissions
- **Request Merging** - Combine duplicate requests
- **Attachment Support** - Upload mockups or examples with requests

QC Numbers: 1000-1050 range

13. UI Quality Control System

- **3-Digit QC Numbers** - Every UI element numbered (e.g., [101], [102])
- **Development Mode Toggle** - Show/hide QC numbers in Settings
- **Easy Production Removal** - One-click to remove all QC numbers for launch
- **Communication Tool** - Makes UI discussions precise and easy
- **Sequential Numbering:**
 - 100s: Login/Authentication
 - 200s: Global navigation and Dashboard
 - 300s: Accounts Management
 - 400s: Scenario Planning
 - 500s: Scenario Editor/Logic Builder
 - 600s: AI Strategy Builder Wizard
 - 700s: Playbook Report View
 - 800s: Settings & Notifications
 - 900s: Audit Trail Report
 - 1000s: Feature Request System

TECHNICAL ARCHITECTURE

Technology Stack

Frontend

- **Framework:** React 19
- **Routing:** Wouter (lightweight React router)
- **Styling:** Tailwind CSS 4
- **UI Components:** shadcn/ui (accessible, customizable)
- **State Management:** TanStack Query (React Query)
- **API Client:** tRPC React (type-safe)
- **Charts:** Chart.js + react-chartjs-2
- **Drag-and-Drop:** React Grid Layout
- **Markdown Rendering:** Streamdown (for AI-generated content)
- **Icons:** Lucide React

Backend

- **Runtime:** Node.js 22
- **Framework:** Express 4
- **API Layer:** tRPC 11 (end-to-end type safety)
- **Authentication:** Manus OAuth + JWT
- **Database ORM:** Drizzle ORM
- **Validation:** Zod (schema validation)

Database

- **Development:** MySQL/TiDB (Manus-provided)
- **Production:** PostgreSQL (Vercel-hosted)
- **Schema Management:** Drizzle Kit (migrations)
- **Row-Level Security:** Enabled for multi-user (future)

External Services

- **QuickBooks Online API** - Financial data source
- **Google OAuth 2.0** - User authentication
- **Google Calendar API** - Task/event management
- **AI/LLM API** - OpenAI GPT-4 or Grok (document analysis)
- **Email Service** - SendGrid or AWS SES (notifications)
- **Cloud Storage** - S3 (document storage)

Development Tools

- **Package Manager:** pnpm
- **Build Tool:** Vite
- **TypeScript:** 5.x (strict mode)
- **Testing:** Vitest (unit tests)
- **Linting:** ESLint + Prettier
- **Git:** GitHub (version control)
- **CI/CD:** GitHub Actions → Vercel

Hosting & Deployment

- **Platform:** Vercel (serverless)
- **Edge Network:** Vercel Edge Functions
- **Analytics:** Vercel Analytics
- **Error Tracking:** Sentry
- **Monitoring:** Built-in health checks

Database Schema (Key Tables)

Core Tables

```
// Users
users {
    id: int (PK, auto-increment)
    openId: varchar(64) (unique, from OAuth)
    name: text
    email: varchar(320)
    loginMethod: varchar(64)
    role: enum('user', 'admin')
    createdAt: timestamp
    updatedAt: timestamp
    lastSignedIn: timestamp
}

// QuickBooks Connections
quickbooks_connections {
    id: int (PK)
    userId: int (FK → users.id)
    accessToken: text (encrypted)
    refreshToken: text (encrypted)
    realmId: varchar(64)
    lastSyncedAt: timestamp
    syncFrequency: enum('hourly', '4hours', '8hours', 'daily')
    isActive: boolean
}

// Accounts (from QuickBooks)
accounts {
    id: int (PK)
    userId: int (FK)
    qbAccountId: varchar(64) (QuickBooks ID)
    name: varchar(255)
    type: enum('checking', 'savings', 'credit_card', 'other')
    balance: decimal(15,2)
    currency: varchar(3)
    lastSyncedAt: timestamp
}

// Transactions (from QuickBooks)
transactions {
    id: int (PK)
    userId: int (FK)
```

```
    accountId: int (FK)
    qbTransactionId: varchar(64)
    date: date
    amount: decimal(15,2)
    description: text
    category: varchar(255)
    type: enum('income', 'expense', 'transfer')
    lastSyncedAt: timestamp
}

// Bills (from QuickBooks)
bills {
    id: int (PK)
    userId: int (FK)
    qbBillId: varchar(64)
    vendorName: varchar(255)
    amount: decimal(15,2)
    dueDate: date
    isPaid: boolean
    paidDate: date
    automationStatus: enum('manual', 'automated')
    expectedAmount: decimal(15,2)
    varianceThreshold: decimal(5,2) (default 3.0)
}

// Scenarios
scenarios {
    id: int (PK)
    userId: int (FK)
    name: varchar(255)
    description: text
    startDate: date
    endDate: date
    isLocked: boolean
    createdAt: timestamp
    updatedAt: timestamp
}

// Logic Rules
logic_rules {
    id: int (PK)
    scenarioId: int (FK)
    conditionType: enum('date', 'balance', 'transaction')
    conditionValue: text (JSON)
    actionType: enum('flag_risk', 'send_alert', 'adjust_projection')
    actionValue: text (JSON)
```

```
    priority: int
}

// Simulation Results
simulation_results {
    id: int (PK)
    scenarioId: int (FK)
    simulationDate: date
    projectedBalance: decimal(15,2)
    riskLevel: enum('none', 'low', 'medium', 'high', 'critical')
    flaggedIssues: text (JSON)
}

// Uploaded Documents
uploaded_documents {
    id: int (PK)
    userId: int (FK)
    filename: varchar(255)
    fileKey: varchar(512) (S3 key)
    fileUrl: text (S3 URL)
    mimeType: varchar(100)
    fileSize: int
    uploadedAt: timestamp
    processedAt: timestamp
    extractedData: text (JSON)
}

// Audit Logs
audit_logs {
    id: int (PK)
    userId: int (FK)
    action: varchar(255)
    entityType: varchar(100)
    entityId: int
    beforeData: text (JSON)
    afterData: text (JSON)
    ipAddress: varchar(45)
    userAgent: text
    createdAt: timestamp
    checksum: varchar(64) (tamper detection)
}

// Dashboard Layouts
dashboard_layouts {
    id: int (PK)
    userId: int (FK)
```

```
layoutName: varchar(255)
layoutConfig: text (JSON)
isActive: boolean
createdAt: timestamp
updatedAt: timestamp
}

// Feature Requests
feature_requests {
    id: int (PK)
    userId: int (FK)
    title: varchar(255)
    description: text
    category: varchar(100)
    status: enum('submitted', 'under_review', 'planned', 'in_progress',
'completed', 'declined')
    votes: int (default 0)
    createdAt: timestamp
    updatedAt: timestamp
}

// Feature Request Comments
feature_request_comments {
    id: int (PK)
    featureRequestId: int (FK)
    userId: int (FK)
    comment: text
    createdAt: timestamp
}
```

API Architecture (tRPC)

```
// server/routers.ts structure
appRouter = {
  auth: {
    me: publicProcedure.query()
    logout: publicProcedure.mutation()
  },
  quickbooks: {
    connect: protectedProcedure.mutation()
    disconnect: protectedProcedure.mutation()
    sync: protectedProcedure.mutation()
    getConnectionStatus: protectedProcedure.query()
    updateSyncFrequency: protectedProcedure.mutation()
  },
  accounts: {
    list: protectedProcedure.query()
    getById: protectedProcedure.query()
    getTransactions: protectedProcedure.query()
  },
  bills: {
    list: protectedProcedure.query()
    updateAutomation: protectedProcedure.mutation()
  },
  scenarios: {
    list: protectedProcedure.query()
    create: protectedProcedure.mutation()
    update: protectedProcedure.mutation()
    delete: protectedProcedure.mutation()
    lock: protectedProcedure.mutation()
  },
  logic: {
    listRules: protectedProcedure.query()
    createRule: protectedProcedure.mutation()
    updateRule: protectedProcedure.mutation()
    deleteRule: protectedProcedure.mutation()
  },
  simulation: {
    run: protectedProcedure.mutation()
  }
}
```

```
    getResults: protectedProcedure.query()
  },

  reports: {
    generate: protectedProcedure.mutation()
    export: protectedProcedure.mutation()
  },

  documents: {
    upload: protectedProcedure.mutation()
    list: protectedProcedure.query()
    process: protectedProcedure.mutation()
  },

  strategyBuilder: {
    analyzeDocuments: protectedProcedure.mutation()
    getRecommendations: protectedProcedure.query()
    createFromRecommendation: protectedProcedure.mutation()
  },

  calendar: {
    connect: protectedProcedure.mutation()
    sync: protectedProcedure.mutation()
    getEvents: protectedProcedure.query()
  },

  dashboard: {
    getLayout: protectedProcedure.query()
    saveLayout: protectedProcedure.mutation()
    resetLayout: protectedProcedure.mutation()
  },

  audit: {
    list: protectedProcedure.query()
    export: protectedProcedure.mutation()
  },

  featureRequests: {
    list: protectedProcedure.query()
    create: protectedProcedure.mutation()
    vote: protectedProcedure.mutation()
    comment: protectedProcedure.mutation()
    // Admin only:
    updateStatus: adminProcedure.mutation()
    merge: adminProcedure.mutation()
  },
}
```

```
    system: {
      notifyOwner: protectedProcedure.mutation()
    }
}
```

Security Architecture

Authentication Flow

1. User clicks “Sign in with Google” [QC: 109]
2. Redirect to Google OAuth consent screen
3. Google redirects back with authorization code
4. Backend exchanges code for access token
5. Backend creates/updates user in database
6. Backend generates JWT session token
7. JWT stored in httpOnly cookie
8. All subsequent requests include cookie automatically

Authorization

- **Public Procedures:** Login, registration (no auth required)
- **Protected Procedures:** Require valid JWT in cookie
- **Admin Procedures:** Require valid JWT + role= ‘admin’
- **Row-Level Security:** All queries filtered by userId (future: database-level)

Data Security

- **Encryption at Rest:** QuickBooks tokens encrypted with AES-256
- **Encryption in Transit:** HTTPS only (enforced by Vercel)
- **SQL Injection Prevention:** Parameterized queries via Drizzle ORM
- **XSS Prevention:** React auto-escaping + Content Security Policy
- **CSRF Protection:** SameSite cookies + CSRF tokens
- **Rate Limiting:** API endpoint throttling (100 req/min per user)

Secrets Management

- **Environment Variables:** All secrets in .env (never committed)
 - **Vercel Secrets:** Production secrets managed via Vercel dashboard
 - **Key Rotation:** QuickBooks tokens refreshed automatically
 - **Access Logs:** All sensitive operations logged to audit trail
-

WIREFRAMES WITH QC NUMBERS

Wireframe 1: Login / Authentication Page

The wireframe shows a login page with the following components:

- [101] FINANCIAL PLAYBOOK SYSTEM (Application title)
- [102] Email: [103] [_____] (Email input field)
- [104] Password: [105] [_____] (Password input field)
- [106] [LOGIN] (Login button)
- [107] [Forgot Password?] (Forgot Password link)
- [108] [Create New Account] (Create New Account link)
- [109] [Sign in with Google] (Sign in with Google button)

Components:

- [101] Application title/branding
- [102-103] Email input field
- [104-105] Password input field

- [106] Login button
 - [107] Password recovery link
 - [108] New account registration link
 - [109] Google OAuth button 
-

Wireframe 2: Dashboard / Home Page (Customizable Layout)

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
 [Accounts] [205] [Scenarios] [206] [Reports] [221] [Calendar] |
 [207] [User: Dwain]
 [208] [Logout] |

[209] ! ACTIVE ALERTS (Widget - Draggable/Resizable)

[210]  Overdraft Risk: Chase Account - Projected 12/15/2025	 [More Info]
[211]  Large AR Lag: Invoice #1234 - 45 days overdue	 [More Info]

[212] CASH FLOW CHART (Widget - Draggable/Resizable)

[213] [Interactive Chart with Green/Yellow/Red zones]
Balance over Time with Event Markers

[214] QUICK STATS (Widget - Draggable/Resizable)

[215] Total Balance \$45,230 	[216] Monthly Burn -\$5,100 	[217] AR Outstanding \$12,400 
---	--	--

[218] [+ New Scenario] [219] [View All Scenarios] [220] [Generate Report]

[222] [ Customize Dashboard] [223] [ Lock Layout]

Components:

- [201] Application logo
- [202] Application title
- [203-206] Main navigation menu items
- [221] Calendar navigation item [NEW]
- [207] User profile display
- [208] Logout button
- [209] Alerts widget (draggable/resizable) [NEW]
- [210-211] Individual alert items with status indicators [NEW]
- [212] Cash flow chart widget (draggable/resizable) [NEW]
- [213] Interactive chart with color zones [NEW]
- [214] Quick stats widget (draggable/resizable) [NEW]
- [215-217] Stat cards with status indicators [NEW]
- [218-220] Action buttons
- [222] Customize dashboard button [NEW]
- [223] Lock/unlock layout toggle [NEW]

Note: All financial data is READ-ONLY from QuickBooks. Status indicators (  ) show sync status and data quality.

Wireframe 3: Accounts Management Page

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] | [207] [User: Dwain]
[208] [Logout]

[301] ACCOUNTS MANAGEMENT

[302] QuickBooks Connection: Connected
[303] Last Synced: 2025-11-24 10:45 AM
[304] [Sync Now] [305] [Configure Sync]

[306] BANK ACCOUNTS (READ-ONLY - from QuickBooks)

[307] Chase Business Checking
Balance: \$39,255.53 [308] [View in QuickBooks]
 In sync - Last verified 5 minutes ago

[309] PNC Savings
Balance: \$15,200.00 [310] [View in QuickBooks]
 In sync - Last verified 5 minutes ago

[311] CREDIT CARDS (READ-ONLY - from QuickBooks)

[312] Chase Business Credit Card
Balance: -\$3,450.00 [313] [View in QuickBooks]
 Warning: Approaching credit limit (85%)

[314] UPCOMING BILLS (READ-ONLY - from QuickBooks)

[315] Connectwise - Due 12/01/2025
Amount: \$1,700.00 [316] [View in QuickBooks]
 Price variance: Expected \$1,683.88 (+\$16.12)
[317] [Set Automation]

[318] Miamisburg Loan - Due 02/15/2026
Amount: \$5,627.01 [319] [View in QuickBooks]

| | | [i] In sync - Matches expected amount | | |

| | | [320] [⚙ Set Automation] | | |

| | | [321] ACCOUNTS RECEIVABLE (READ-ONLY - from QuickBooks) | | |

| | | [322] Invoice #1234 - Customer ABC Corp | | |

| | | Amount: \$5,200.00 [red circle] [323] [View in QuickBooks] | | |

| | | [i] Critical: 45 days overdue - Expected payment missing | | |

| | | [324] Invoice #1235 - Customer XYZ Inc | | |

| | | Amount: \$3,100.00 [green circle] [325] [View in QuickBooks] | | |

| | | [i] Due in 15 days - On track | | |

| | | [326] [📊 View Account History] [327] [📈 Generate Account Report] | | |

Components:

- [301] Page title
- [302] QuickBooks connection status with indicator [NEW]
- [303] Last synced timestamp [NEW]
- [304] Manual sync button [NEW]
- [305] Configure sync settings button [NEW]
- [306-310] Bank accounts section (READ-ONLY) [NEW]
- [311-313] Credit cards section (READ-ONLY) [NEW]
- [314-320] Upcoming bills section (READ-ONLY) [NEW]
- [317, 320] Automation settings buttons [NEW]
- [321-325] Accounts receivable section (READ-ONLY) [NEW]
- [326-327] Action buttons

Key Features:

- All QuickBooks data is READ-ONLY
- “View in QuickBooks” links for editing at source

- Status indicators () on every item
 - Info icons (i) with detailed explanations
 - Automation configuration per bill
-

Wireframe 4: Scenario Planning Page

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] |
[207] [User: Dwain]
[208] [Logout]

[401] SCENARIO PLANNING

[402] [+ Create New Scenario] [403] [Import from Strategy Builder]

[404] MY SCENARIOS

[405] Q4 2025 Operations (Active) ●

Created: 11/01/2025 | Last Modified: 11/24/2025

Status: In sync with QuickBooks

[406] [Edit] [407] [Run Simulation] [408] [Lock]

[409] Hiring Scenario - Add Technician ○

Created: 11/15/2025 | Last Modified: 11/20/2025

Status: Needs review - Price variance detected

[410] [Edit] [411] [Run Simulation] [412] [Delete]

[413] Conservative Cash Flow (Locked) 🔒

Created: 10/01/2025 | Locked: 11/01/2025

Status: Archived - Historical reference

[414] [View Only] [415] [Unlock] [416] [Clone]

[417] SCENARIO COMPARISON

[418] Select scenarios to compare:

[419] Q4 2025 Operations

[420] Hiring Scenario - Add Technician

[421] Conservative Cash Flow

[422] [Compare Selected Scenarios]

[423] [Side-by-side chart showing projected balances]

| [424] [Generate Comparison Report] [425] [Export All Scenarios] |

Components:

- [401] Page title
 - [402] Create new scenario button
 - [403] Import from AI Strategy Builder [NEW]
 - [404] Scenarios list section
 - [405-416] Individual scenario cards with status indicators [NEW]
 - [417] Scenario comparison section [NEW]
 - [418-422] Comparison selector and button [NEW]
 - [423] Comparison chart [NEW]
 - [424-425] Action buttons
-

Wireframe 5: Scenario Editor / Logic Builder

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] |
[207] [User: Dwain]
[208] [Logout]

[501] SCENARIO EDITOR: "Q4 2025 Operations"

[502] [Save] [Run Simulation] [504] [Back to Scenarios]

[505] SCENARIO DETAILS

[506] Name: [507] Q4 2025 Operations
[508] Start Date: [509] 11/01/2025
[510] End Date: [511] 12/31/2025
[512] Description: [513] [Text area]

[514] LOGIC RULES

[515] [+ Add New Rule]

[516] Rule 1: Reserve Risk Alert

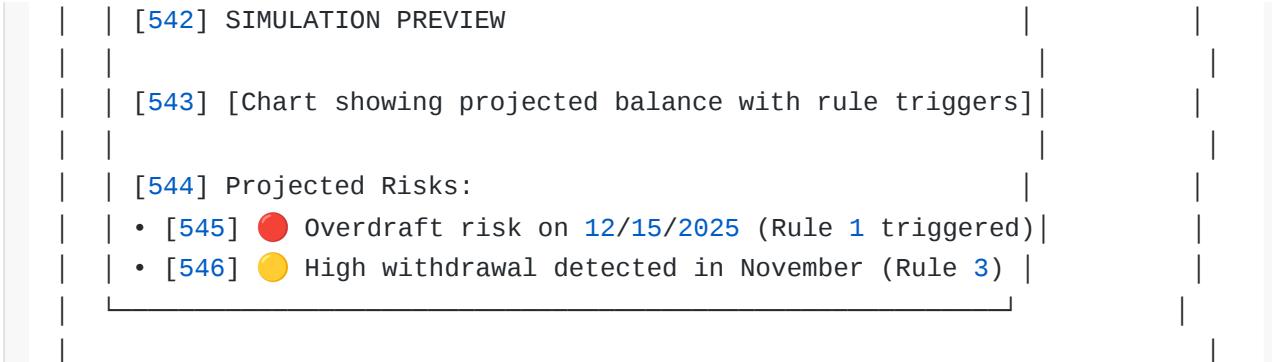
IF [517] Chase Balance [518] [<] [519] [\$8,000]
THEN [520] Flag Risk: "Reserve Risk"
[521] [Edit] [522] [Delete] [523] [▲] [524] [▼]

[525] Rule 2: Miamisburg Loan Payment

ON [526] 02/15/2026
THEN [527] Pay Bill: "Miamisburg Loan" \$5,627.01
[528] [Edit] [529] [Delete] [530] [▲] [531] [▼]

[532] Rule 3: High Withdrawal Warning

IF [533] Owner Draw [534] [>] [535] [\$5,000]
IN [536] Single Month
THEN [537] Send Alert: "High Withdrawal"
[538] [Edit] [539] [Delete] [540] [▲] [541] [▼]



Components:

- [501] Page title with scenario name
- [502-504] Action buttons
- [505-513] Scenario details section
- [514] Logic rules section header
- [515] Add new rule button
- [516-541] Individual logic rules with visual rule builder NEW
- [542-546] Simulation preview section NEW

Key Features:

- Visual rule builder (IF/THEN logic)
- Drag-and-drop rule reordering
- Real-time simulation preview
- Risk detection and flagging

Wireframe 6: AI Strategy Builder Wizard

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] |
[207] [User: Dwain]
[208] [Logout]

[601] AI STRATEGY BUILDER WIZARD

[602] Step 1 of 3: Upload Documents
[603] ●—○—○

[604] UPLOAD FINANCIAL DOCUMENTS

[605] Drag and drop files here, or click to browse

[606] Recommended documents:

- Last 3-6 months of bank statements
- Last 3-6 months of financial statements
- Recent credit card statements

[607] Supported formats: PDF, PNG, JPG

[608] UPLOADED FILES:

[609] ✓ Chase_Statement_Oct_2025.pdf (2.3 MB)
[610] [Remove]
[611] ✓ Chase_Statement_Nov_2025.pdf (2.1 MB)
[612] [Remove]
[613] ✓ Financial_Statement_Q4.pdf (1.8 MB)
[614] [Remove]

[615] [BACK] Cancel] [616] [Next: Analyze Documents ➔]

Step 1 Components:

- [601] Wizard title
- [602-603] Progress indicator

- [604-614] Document upload section with drag-and-drop 
- [615-616] Navigation buttons

[617] AI STRATEGY BUILDER WIZARD

[618] Step 2 of 3: Review Analysis

[619] ○————●————○

[620] AI ANALYSIS RESULTS

[621] ✓ Documents processed successfully

[622] ✓ Math validation passed **for** all extracted data

[623] EXTRACTED DATA SUMMARY:

[624] Average Monthly Income: **\$18,500.00** 

 Based on 3 months of data

[625] Average Monthly Expenses: **\$13,200.00** 

 Based on 3 months of data

[626] Net Monthly Cash Flow: **+\$5,300.00** 

 Positive trend detected

[627] RECURRING BILLS IDENTIFIED:

- [628] Connectwise: **\$1,683.88/month** (1st of month)
- [629] Office Rent: **\$2,500.00/month** (15th of month)
- [630] Utilities: **~\$450.00/month** (varies)
- [631] Insurance: **\$1,200.00/quarter** (1st of quarter)

[632]  Edit Extracted Data]

[633] [ Back] [634] [Next: Choose Strategy 

Step 2 Components:

- [617-619] Progress indicator (step 2)

- [620-632] AI analysis results with math validation confirmation NEW
 - [633-634] Navigation buttons
-

[635] AI STRATEGY BUILDER WIZARD

[636] Step 3 of 3: Choose Strategy

[637] ○—○—●

[638] RECOMMENDED STRATEGIES

[639] ★ RECOMMENDED: Cash Flow Optimization

Based on your positive cash flow trend, this strategy focuses on maximizing savings and building reserves.

Key Features:

- Automated bill tracking
- Reserve goal: \$25,000 by Q2 2026
- Alerts for unusual expenses

[640] [Select This Strategy]

[641] Growth & Expansion Strategy

Your cash flow supports growth. This strategy models hiring, equipment purchases, and expansion.

Key Features:

- Scenario for adding 1-2 employees
- Equipment budget: \$15,000
- Maintains 60-day runway

[642] [Select This Strategy]

[643] Debt Reduction Focus

Prioritize paying down existing debt while maintaining operational cash flow.

Key Features:

- Extra \$2,000/month to Miamisburg Loan
- Payoff accelerated by 18 months
- Interest savings: \$4,200

[644] [Select This Strategy]

[645] [Create Custom Strategy Instead]

[646] [BACK] Back] [647] [Create Playbook ✓]

Step 3 Components:

- [635-637] Progress indicator (step 3)
- [638-645] AI-recommended strategies with detailed descriptions NEW
- [646-647] Navigation buttons

Key Features:

- 3-step wizard process
 - Document upload with drag-and-drop
 - AI-powered data extraction with math validation
 - Recurring bill identification
 - 3 tailored strategy recommendations
 - Automatic playbook creation
-

Wireframe 7: Playbook Report View

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] |
[207] [User: Dwain]
[208] [Logout]

[701] PLAYBOOK REPORT: "Q4 2025 Operations"

[702] Generated: 11/24/2025 10:45 AM

[703] [Export PDF] [704] [Export Word] [705] [Export CSV]

[706] EXECUTIVE SUMMARY

[707] This playbook simulates Q4 2025 operations based on current QuickBooks data and projected activities. The analysis identifies 2 critical risks and 3 warnings that require attention.

[708] Overall Risk Level: 🟡 MODERATE

[709] Key Findings:

- [710] 🔴 Overdraft risk projected on 12/15/2025
- [711] 🔴 Large AR lag: Invoice #1234 (45 days)
- [712]🟡 Price variance: Connectwise bill (+\$16.12)

[713] CRITICAL RISKS IDENTIFIED

[714] Risk 1: Projected Overdraft on 12/15/2025

[715] ROOT CAUSE ANALYSIS (AI-Generated, Math-Validated):

The overdraft is caused by a combination of:

1. Large owner draw on 12/10 (\$5,000)
2. Miamisburg Loan payment on 12/15 (\$5,627.01)
3. Delayed AR payment from Invoice #1234 (\$5,200)

Current projected balance on 12/15: -\$1,372.01

✓ Math validation: Calculation verified by traditional code

[716] RECOMMENDED SOLUTIONS:

- 1. Reduce owner draw on 12/10 to \$2,000 (saves \$3,000)
- 2. Follow up on Invoice #1234 immediately
- 3. Delay Miamisburg Loan payment to 12/20 (if possible)

[717] [Apply Solution 1] [718] [Apply Solution 2]

[719] RUNNING BALANCE REGISTER

[720] [Table showing daily balance with events]

Date	Event	Amount	Balance
11/24/2025	Starting Balance	--	\$39,255.53
12/01/2025	Connectwise Bill	-\$1,700.00	\$37,555.53
12/05/2025	Customer Payment	+\$3,100.00	\$40,655.53
12/10/2025	Owner Draw	-\$5,000.00	\$35,655.53
12/15/2025	Miamisburg Loan	-\$5,627.01	-\$1,372.01
...

[721] CASH FLOW PROJECTION CHART

[722] [Interactive chart with green/yellow/red zones]

[723] Legend:

- Green Zone: Balance > \$15,000 (Safe)
- Yellow Zone: Balance \$5,000-\$15,000 (Caution)
- Red Zone: Balance < \$5,000 (Critical)

[724] [BACK] Back to Scenarios] [725] [EMAIL Email Report] [726] [PRINT Print]

Components:

- [701-705] Report header with export options
- [706-712] Executive summary section NEW
- [713-718] Critical risks with AI analysis and solutions NEW
- [719-720] Running balance register table NEW
- [721-723] Cash flow projection chart with color zones NEW

- [724-726] Action buttons

Key Features:

- AI-powered root cause analysis
 - Math validation confirmation on all calculations
 - AI-generated solution recommendations
 - Detailed register with daily projections
 - Interactive charts with risk zones
 - Multiple export formats
-

Wireframe 8: Settings & Notifications

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] | [207] [User: Dwain]
[208] [Logout]

[801] SETTINGS

[802] QUICKBOOKS CONNECTOR

[803] Status:  Connected

[804] Connected Account: Superior Networks LLC

[805] Last Synced: 11/24/2025 10:45 AM

[806] Sync Frequency: [807] [Dropdown: Hourly ▼]

Options: Hourly, Every 4 hours, Every 8 hours, Daily

[808] Sync on Login: [809] [Enabled]

[810] [Sync Now] [811] [Disconnect QuickBooks]

[812] [Reconnect QuickBooks]

[813] NOTIFICATION SETTINGS

[814] Email Notifications: [815] [Enabled]

[816] Email Address: [817] [dwain@superiornetworks.com]

[818] Alert Thresholds:

[819] Price Variance Threshold: [820] [3.0%]

(Alert when bill amount changes by more than this)

[821] Overdue Alert Delay: [822] [3 days]

(Alert when expected transaction is missing)

[823] Critical Risk Alert: [824] [5 days]

(Alert when upcoming activity **not** in QuickBooks)

[825] Notification Types:

- [826] Critical Risks (RED status)
- [827] Warnings (YELLOW status)
- [828] Info Updates (GREEN status)
- [829] QuickBooks Sync Failures
- [830] Weekly Summary Reports

[831] GOOGLE CALENDAR INTEGRATION

- [832] Status:  Connected
- [833] Connected Account: dwain@superiornetworks.com
- [834] Sync Events: [835] Enabled
- [836] Event Types to Sync:
 - [837] Upcoming Bills
 - [838] Expected Payments (AR)
 - [839] Critical Alerts (RED status)
 - [840] All Warnings (YELLOW status)
- [841] [ Disconnect Calendar] [842] [ Reconnect]

[843] DEVELOPMENT MODE

- [844] Show QC Numbers: [845] Disabled
 - (Enable to show [###] numbers on all UI elements)
- [846] [ Remove All QC Numbers for Production]

- [847] [ Save Settings] [848] [ Reset to Defaults]

Components:

- [801] Page title
- [802-812] QuickBooks connector settings NEW
- [813-830] Notification settings with configurable thresholds NEW
- [831-842] Google Calendar integration settings NEW
- [843-846] Development mode toggle for QC numbers NEW

- [847-848] Action buttons

Key Features:

- Configurable sync frequency (hourly, 4hr, 8hr, daily)
 - Sync on login toggle
 - Customizable alert thresholds (price variance, overdue delay, critical risk timing)
 - Selective notification types
 - Google Calendar event sync configuration
 - QC number visibility toggle
 - One-click production cleanup
-

Wireframe 9: Audit Trail Report

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] | [207] [User: Dwain]
[208] [Logout]

[901] AUDIT TRAIL

[902] [Export CSV] [903] [Advanced Filters]

[904] FILTERS

[905] Date Range: [906] [11/01/2025] to [907] [11/24/2025]
[908] Action Type: [909] [All Actions ▾]
[910] User: [911] [All Users ▾] (future: multi-user)
[912] [Apply Filters] [913] [Clear Filters]

[914] AUDIT LOG ENTRIES

[915] 11/24/2025 10:45 AM | User: Dwain
Action: Scenario Modified
Entity: "Q4 2025 Operations"
Changes: Updated end date from 12/31/2025 to 01/15/2026
[916] [View Details]

[917] 11/24/2025 10:30 AM | User: Dwain
Action: QuickBooks Sync
Entity: All Accounts
Changes: 15 transactions synced, 2 new bills added
[918] [View Details]

[919] 11/24/2025 09:15 AM | User: Dwain
Action: Report Generated
Entity: "Q4 2025 Operations Report"
Changes: PDF exported
[920] [View Details]

... (more entries) ...

Components:

- [901-903] Page header with export and filter options
- [904-913] Filter section (date range, action type, user)
- [914-920] Audit log entries list with details
- [921-923] Pagination controls

Key Features:

- Immutable log of all actions
 - Before/after snapshots (visible in details)
 - Filterable by date, action type, user
 - CSV export for compliance
 - Tamper detection via checksums
-

Wireframe 10: Calendar View

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] [221] [Calendar] |
[207] [User: Dwain]
[208] [Logout]

[950] FINANCIAL CALENDAR

[951] [<◀ Previous Month] [952] December 2025 [953] [Next Month ▶]
[954] [Today] [955] [Sync with Google Calendar]

[956] FILTERS

[957] Bills Payments Alerts Tasks

[961] CALENDAR GRID

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1 Bill \$1,700	2	3 Bill \$450	4	5
6	7	8	9	10 Owner Draw \$5,000	11	12
13	14	15 Loan \$5,627 ⚠ Risk	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

[962] UPCOMING EVENTS (Next 7 Days)

- | | | |
|-------------|--|---|
| [963] 12/01 | | Connectwise Bill - \$1,700.00 |
| [964] 12/03 | | Utilities Bill - \$450.00 |
| [965] 12/10 | | Owner Draw - \$5,000.00 (High withdrawal) |
| [966] 12/15 | | Miamisburg Loan - \$5,627.01 (Overdraft risk) |

Components:

- [950-955] Calendar header with navigation and sync
- [956-960] Event type filters
- [961] Calendar grid with color-coded events NEW
- [962-966] Upcoming events list with status indicators NEW

Key Features:

- Month/week/day views
 - Color-coded events matching status system ()
 - Two-way sync with Google Calendar
 - Filter by event type
 - Click events for details
-

Wireframe 11: Feature Request System

[201] [LOGO] [202] Financial Playbook [203] [Dashboard] [204]
[Accounts] [205] [Scenarios] [206] [Reports] [224] [Feature Requests] |
[207] [User: Dwain]
[208] [Logout]

[1000] FEATURE REQUESTS

[1001] [+ Submit New Request] [1002] [View Roadmap] [1003] [My Requests]

[1004] FILTERS & SORT
[1005] Status: [1006] [All ▾]
[1007] Category: [1008] [All ▾]
[1009] Sort by: [1010] [Most Votes ▾]

[1011] FEATURE REQUEST LIST

[1012] 45 | Multi-Currency Support
Status: Planned (Q2 2026)
Category: Integrations
Submitted by: User123 on 11/15/2025
[1013] [View Details] [1014] [Upvote]

[1015] 32 | Mobile App (iOS/Android)
Status: Under Review
Category: Platform
Submitted by: User456 on 11/10/2025
[1016] [View Details] [1017] [Upvote]

[1018] 28 | Dark Mode Theme
Status: In Progress
Category: UI/UX
Submitted by: User789 on 11/05/2025
[1019] [View Details] [1020] [Upvote]

... (more requests) ...

Components:

- [1000-1003] Page header with action buttons
- [1004-1010] Filters and sorting
- [1011-1020] Feature request list with voting NEW

Key Features:

- Submit new feature ideas
 - Vote on existing requests
 - Status tracking (Submitted, Under Review, Planned, In Progress, Completed, Declined)
 - Public roadmap view
 - Commenting and discussion
 - Admin dashboard for management
-

EXTERNAL INTEGRATIONS

1. QuickBooks Online API

Purpose: Primary financial data source

Authentication: OAuth 2.0

Scope: Full read access to accounts, transactions, bills, invoices

Key Endpoints:

- /v3/company/{realmId}/account - List all accounts
- /v3/company/{realmId}/query - Query transactions
- /v3/company/{realmId}/bill - List bills
- /v3/company/{realmId}/invoice - List invoices
- /v3/company/{realmId}/vendor - List vendors

- `/v3/company/{realmId}/customer` - List customers

Webhooks:

- Transaction created/updated
- Bill created/updated/paid
- Invoice created/updated/paid
- Account balance changed

Data Refresh:

- On login (automatic)
- Manual sync button
- Background sync (configurable: hourly, 4hr, 8hr, daily)
- Webhook-triggered (real-time)

Rate Limits:

- 500 requests per minute
- 10,000 requests per day

Error Handling:

- Token refresh on 401 errors
- Exponential backoff on rate limits
- User notification on sync failures

2. Google OAuth 2.0

Purpose: User authentication

Scope: `openid, email, profile`

Flow:

1. User clicks “Sign in with Google”
2. Redirect to Google consent screen
3. Google redirects back with authorization code

4. Backend exchanges code for access token
5. Fetch user profile from Google
6. Create/update user in database
7. Generate JWT session token

Token Management:

- Access tokens valid for 1 hour
 - Refresh tokens stored encrypted
 - Automatic refresh before expiration
-

3. Google Calendar API

Purpose: Task and event management

Authentication: OAuth 2.0

Scope: calendar.events

Key Endpoints:

- `/calendar/v3/calendars/primary/events` - List events
- `/calendar/v3/calendars/primary/events` (POST) - Create event
- `/calendar/v3/calendars/primary/events/{eventId}` (PUT) - Update event
- `/calendar/v3/calendars/primary/events/{eventId}` (DELETE) - Delete event

Event Types Created:

- Upcoming bills (with amount and due date)
- Expected payments (AR invoices)
- Critical alerts (RED status items)
- Scenario milestones

Color Coding:

- Green events: Normal, on-track
- Yellow events: Warnings
- Red events: Critical, action required

Sync Frequency:

- Real-time on playbook changes
 - Background sync every 4 hours
 - Manual sync button
-

4. AI/LLM API (OpenAI GPT-4 or Grok)

Purpose: Document analysis, recommendations, root cause analysis

Authentication: API key (server-side only)

Use Cases:

Document Analysis

- OCR text extraction from PDFs and images
- Transaction identification and categorization
- Recurring bill detection
- Income/expense pattern analysis

Math Validation:

- All AI-extracted numbers verified by traditional code
- Discrepancies flagged for user review
- Validation results logged to audit trail

Strategy Recommendations

- Analyze financial trends
- Suggest 1-3 tailored playbook strategies
- Provide rationale for each recommendation

Root Cause Analysis

- Analyze simulation results
- Identify why risks occurred

- Generate natural language explanations

Solution Generation

- Suggest actionable solutions to identified risks
- Prioritize solutions by impact
- Provide step-by-step implementation guidance

API Configuration:

```
const response = await invokeLLM({
  messages: [
    { role: "system", content: "You are a financial analyst..." },
    { role: "user", content: "Analyze this bank statement..." }
  ],
  response_format: {
    type: "json_schema",
    json_schema: {
      name: "transaction_analysis",
      strict: true,
      schema: {
        type: "object",
        properties: {
          transactions: { type: "array", items: { ... } },
          recurring_bills: { type: "array", items: { ... } },
          average_monthly_income: { type: "number" },
          average_monthly_expenses: { type: "number" }
        }
      }
    }
  }
});
```

5. Email Service (SendGrid or AWS SES)

Purpose: Notification delivery

Authentication: API key (server-side only)

Email Types:

Alert Emails

- Critical risks (RED status)
- Warnings (YELLOW status)
- QuickBooks sync failures
- Overdue transactions

Template:

Subject:  Critical Alert: Overdraft Risk Detected

Hi Dwain,

Financial Playbook has detected a critical risk that requires your attention:

Overdraft Risk on 12/15/2025

Projected balance: -\$1,372.01

Root Cause:

- Large owner draw on 12/10 (\$5,000)
- Miamisburg Loan payment on 12/15 (\$5,627.01)
- Delayed AR payment from Invoice #1234 (\$5,200)

Recommended Solutions:

1. Reduce owner draw on 12/10 to \$2,000 (saves \$3,000)
2. Follow up on Invoice #1234 immediately
3. Delay Miamisburg Loan payment to 12/20 (if possible)

[View Full Report] [Login to Financial Playbook]

This is an automated alert from Financial Playbook.

To adjust notification settings, visit Settings > Notifications.

Weekly Summary Emails

- Overview of financial health
- New alerts since last week
- Upcoming bills and payments

- QuickBooks sync status

Feature Request Updates

- Status changes (Planned, In Progress, Completed)
- Admin responses to requests
- Merged duplicate requests

Configuration:

- Configurable in Settings > Notifications
 - Selective notification types
 - Customizable thresholds
-

6. Cloud Storage (S3)

Purpose: Secure document storage

Authentication: AWS credentials (server-side only)

Stored Files:

- Uploaded bank statements
- Uploaded financial documents
- Generated reports (PDF, Word, CSV)
- User profile images (future)

Security:

- Files encrypted at rest (AES-256)
- Non-enumerable file keys (random suffixes)
- Presigned URLs for temporary access
- Metadata stored in database

File Naming Convention:

{userId}-documents/{filename}-{randomSuffix}.{ext}

Example: 123-documents/Chase_Statement_Oct_2025-a3f9b2e4.pdf

Upload Flow:

1. Frontend uploads file to backend endpoint
 2. Backend validates file type and size
 3. Backend uploads to S3 via `storagePut()`
 4. Backend saves metadata to database
 5. Backend returns S3 URL to frontend
-

VISUAL DESIGN SYSTEM

Traffic Light Status System

The application uses a consistent color-coded status system throughout all pages to provide instant visual feedback on data quality and financial health.

Status Levels

GREEN - In Sync / Healthy

- **Meaning:** QuickBooks and Playbook are perfectly aligned, no issues detected
- **Visual:** Subtle green halo or background glow
- **Examples:**
 - Account balance matches QuickBooks exactly
 - Bill amount matches expected amount
 - Transaction synced within last hour
 - No risks detected in scenario

YELLOW - Warning / Attention Needed

- **Meaning:** Minor issue detected that should be reviewed

- **Visual:** Yellow halo with warning icon (
- **Examples:**
 - Price variance exceeds threshold (default 3%)
 - Sync delayed more than 24 hours
 - Duplicate transaction detected
 - Approaching credit limit (>80%)

RED - Critical / Action Required

- **Meaning:** Critical issue requiring immediate attention
- **Visual:** Red halo with alert icon () , optional pulse animation
- **Examples:**
 - Expected activity missing from QuickBooks (5 days out)
 - Expected payment not in register (3 days overdue)
 - Projected overdraft detected
 - Account disconnected

Info Icons ()

Every status indicator includes a clickable info icon that displays a detailed explanation modal.

Modal Contents:

- **Status Level:** Green/Yellow/Red with icon
- **Issue Description:** Plain language explanation
- **Root Cause:** Why this status was assigned
- **Recommended Action:** What to do next
- **Related Data:** Relevant numbers and dates
- **Quick Actions:** Buttons to resolve (e.g., “View in QuickBooks” , “Adjust Playbook”)

Example Modal:

YELLOW WARNING

Price Variance Detected

Your Connectwise bill for December is \$1,700.00, which is \$16.12 (0.96%) higher than the expected amount of \$1,683.88.

This variance exceeds your configured threshold of 3.0%.

Recommended Action:

Review the bill in QuickBooks to identify the cause of the increase. It may be a legitimate price adjustment or an error.

[\[View in QuickBooks\]](#) [\[Adjust Expected Amount\]](#) [\[Dismiss\]](#)

Configurable Thresholds

All status thresholds can be customized in Settings > Notifications:

Threshold	Default	Description
Price Variance	3.0%	Alert when bill amount changes by more than this percentage
Overdue Alert Delay	3 days	Alert when expected transaction is missing for this many days
Critical Risk Alert	5 days	Alert when upcoming activity not in QuickBooks within this timeframe
Sync Delay Warning	24 hours	Yellow status if sync hasn't occurred in this timeframe
Credit Limit Warning	80%	Yellow status when credit card balance exceeds this percentage

Application Across Pages

Dashboard:

- Account balance indicators
- Alert cards (color-coded)

- Quick stats cards
- Cash flow chart zones

Accounts Management:

- Bank account status
- Credit card status
- Bill status
- Invoice status

Scenario Planning:

- Scenario health indicators
- Simulation result status
- Risk level indicators

Reports:

- Overall risk level badge
- Individual risk items
- Chart zones (green/yellow/red)
- Register entries

Calendar:

- Event color coding
 - Date background colors
 - Upcoming events list
-

Color Palette

Primary Colors

- **Green:** #22C55E (Success, in-sync, healthy)
- **Yellow:** #EAB308 (Warning, attention needed)
- **Red:** #EF4444 (Critical, action required)

- **Blue:** #3B82F6 (Primary actions, links)

Neutral Colors

- **Gray 900:** #111827 (Primary text)
- **Gray 700:** #374151 (Secondary text)
- **Gray 500:** #6B7280 (Tertiary text, borders)
- **Gray 200:** #E5E7EB (Dividers)
- **Gray 100:** #F3F4F6 (Backgrounds)
- **White:** #FFFFFF (Cards, panels)

Status Halo Effects

```

/* Green halo */
.status-green {
  background: rgba(34, 197, 94, 0.1);
  border: 1px solid rgba(34, 197, 94, 0.3);
  box-shadow: 0 0 10px rgba(34, 197, 94, 0.2);
}

/* Yellow halo */
.status-yellow {
  background: rgba(234, 179, 8, 0.1);
  border: 1px solid rgba(234, 179, 8, 0.3);
  box-shadow: 0 0 10px rgba(234, 179, 8, 0.2);
}

/* Red halo with pulse */
.status-red {
  background: rgba(239, 68, 68, 0.1);
  border: 1px solid rgba(239, 68, 68, 0.3);
  box-shadow: 0 0 10px rgba(239, 68, 68, 0.2);
  animation: pulse-red 2s ease-in-out infinite;
}

@keyframes pulse-red {
  0%, 100% { box-shadow: 0 0 10px rgba(239, 68, 68, 0.2); }
  50% { box-shadow: 0 0 20px rgba(239, 68, 68, 0.4); }
}

```

Typography

Font Family:

- Primary: Inter (Google Fonts)
- Monospace: JetBrains Mono (for numbers, code)

Font Sizes:

- Heading 1: 2.25rem (36px) - Page titles
- Heading 2: 1.875rem (30px) - Section headers
- Heading 3: 1.5rem (24px) - Subsection headers
- Body: 1rem (16px) - Regular text
- Small: 0.875rem (14px) - Secondary text, labels
- Tiny: 0.75rem (12px) - Captions, timestamps

Font Weights:

- Bold: 700 - Headings, emphasis
 - Semibold: 600 - Subheadings, labels
 - Medium: 500 - Buttons, links
 - Regular: 400 - Body text
-

Spacing System

Base Unit: 4px

Name	Value	Usage
xs	4px	Tight spacing, icon gaps
sm	8px	Small gaps, button padding
md	16px	Standard gaps, card padding
lg	24px	Section spacing
xl	32px	Page margins
2xl	48px	Major section dividers

Component Styles

Buttons

- **Primary:** Blue background, white text, rounded corners
- **Secondary:** White background, blue border, blue text
- **Danger:** Red background, white text
- **Ghost:** Transparent background, gray text, hover effect

Cards

- White background
- Subtle shadow: `0 1px 3px rgba(0,0,0,0.1)`
- Rounded corners: 8px
- Padding: 16px (md)

Inputs

- Border: 1px solid gray-300
- Rounded corners: 6px
- Padding: 8px 12px
- Focus: Blue border, subtle glow

Status Badges

- Pill shape (fully rounded)
 - Small text (0.875rem)
 - Semibold weight
 - Color-coded background (green/yellow/red)
-
-

UI IMPLEMENTATION PLAN

My Final Designer Selections

After analyzing both lists and my capabilities, here are the **4 designers/agencies** I will use as primary references:

Primary Foundation (95% confidence)

1. **Octet** - Design system architecture and component consistency
2. **Ran Liu** - Clean hierarchy and layout structure

Financial Specifics (90% confidence)

1. **Ana Vadillo** - Financial data tables and dense information display

Dashboard & Visualization (88% confidence)

1. **Fuselab Creative** - Customizable dashboard and data visualization

Polish Layer (selective, 92% confidence)

1. **Lazarev** - Professional fintech aesthetic and empty states
-

UI Element Breakdown by Designer

Here's exactly which designer I'll reference for each specific UI element in Financial Playbook:

Login & Authentication Page [QC: 101-109]

Primary Reference: Lazarev

Secondary Reference: Ran Liu

Why:

- Lazarev: Professional fintech credibility, clean onboarding flows
- Ran Liu: Simple, uncluttered layout

Specific Elements:

- [101] Logo placement → Lazarev (centered, prominent)
- [102] “Sign in with Google” button → Lazarev (large, clear CTA)
- [103] Welcome message → Ran Liu (clean typography, minimal)
- [104] Background → Lazarev (subtle gradient or solid professional color)
- [105] Footer links → Ran Liu (simple, understated)

Implementation:

- Clean, centered layout
- Single prominent CTA button
- Minimal distractions
- Professional color palette (blues/grays)
- Simple fade-in animation

Global Navigation & Header [QC: 201-208]

Primary Reference: Octet

Secondary Reference: Ran Liu

Why:

- Octet: Consistent navigation patterns across complex apps
- Ran Liu: Clean, system-like layout grids

Specific Elements:

- [201] App logo → Octet (consistent placement, clickable home)
- [202] Main navigation menu → Octet (clear hierarchy, consistent spacing)
- [203] User profile dropdown → Octet (standard pattern, accessible)
- [204] Notifications bell → Octet (badge for count, dropdown)
- [205] QuickBooks sync status → **Ana Vadillo** (financial data indicator)
- [206] Last synced timestamp → Ana Vadillo (subtle, informative)
- [207] Search bar → Ran Liu (clean, minimal, expandable)
- [208] Settings gear icon → Octet (standard placement)

Implementation:

- Persistent top navigation bar
 - Left: Logo + main nav items
 - Right: Search, notifications, sync status, profile
 - Sticky on scroll
 - Consistent across all pages
 - shadcn/ui Navigation Menu component
-

Dashboard (Customizable) [QC: 209-220]

Primary Reference: Fuselab Creative

Secondary Reference: Ran Liu

Why:

- Fuselab: Dashboard and widget specialization
- Ran Liu: Clean hierarchy (one primary chart per view)

Specific Elements:

- [209] Dashboard title → Ran Liu (clear, prominent)
- [210] “Customize Layout” button → Fuselab (widget management)
- [211] Widget grid container → Fuselab (drag-and-drop grid)
- [212] Active Alerts widget → **Ana Vadillo** (financial alerts with severity)
- [213] Cash Flow Chart widget → Fuselab (primary chart, prominent)
- [214] Quick Stats widget → Ran Liu (clean cards with numbers)
- [215] Upcoming Bills widget → Ana Vadillo (financial table, compact)
- [216] Account Balances widget → Ana Vadillo (financial data, clear hierarchy)
- [217] Recent Activity widget → Ana Vadillo (transaction list, scannable)
- [218] Scenario Status widget → Ran Liu (status cards, simple)
- [219] Calendar Preview widget → Fuselab (compact calendar with events)
- [220] QuickBooks Sync Status widget → Ana Vadillo (sync indicator with details)

Implementation:

- React Grid Layout for drag-and-drop
- 12-column responsive grid
- Each widget = Card component from shadcn/ui
- Resize handles on widgets
- “Lock Layout” toggle
- Layout persistence in database
- Mobile: Stacks vertically

Visual Status System (Traffic Light):

-  GREEN glow → Subtle box-shadow with green hue
 -  YELLOW glow → Box-shadow with yellow/amber hue + warning icon
 -  RED glow → Box-shadow with red hue + alert icon + optional pulse
 - Info bubble () → Tooltip on hover, modal on click
-

Accounts Management Page [QC: 301-329]

Primary Reference: Ana Vadillo

Secondary Reference: Ran Liu

Why:

- Ana Vadillo: Financial data tables, dense information, clear hierarchy
- Ran Liu: Clean layout structure

Specific Elements:

Header Section:

- [301] Page title “Accounts” → Ran Liu (clear hierarchy)
- [302] “Sync Now” button → Ana Vadillo (financial action, prominent)
- [303] Last synced timestamp → Ana Vadillo (subtle, informative)
- [304] Filter dropdown → Ana Vadillo (financial data filtering)

Bank Accounts Section:

- [305] Section title “Bank Accounts” → Ran Liu (clear section break)
- [306] Account list table → **Ana Vadillo** (finance-grade table)
 - [307] Account name column → Ana Vadillo (bold, primary)
 - [308] Account type column → Ana Vadillo (secondary info)
 - [309] Current balance column → Ana Vadillo (prominent, right-aligned)
 - [310] Status indicator → **Custom** (traffic light glow)
 - [311] “View in QuickBooks” link → Ana Vadillo (external link pattern)

Bills Section:

- [312] Section title “Bills” → Ran Liu
- [313] Bills table → Ana Vadillo
 - [314] Bill name column → Ana Vadillo
 - [315] Vendor column → Ana Vadillo
 - [316] Amount column → Ana Vadillo (right-aligned, bold)
 - [317] Due date column → Ana Vadillo (date formatting)

- [318] Status indicator → Custom (traffic light)
- [319] “View in QuickBooks” link → Ana Vadillo

Accounts Receivable Section:

- [320] Section title “Accounts Receivable” → Ran Liu
- [321] AR table → Ana Vadillo
 - [322] Invoice number → Ana Vadillo
 - [323] Customer → Ana Vadillo
 - [324] Amount → Ana Vadillo
 - [325] Due date → Ana Vadillo
 - [326] Status → Custom (traffic light)
 - [327] “View in QuickBooks” link → Ana Vadillo

Empty States:

- [328] No accounts message → **Lazarev** (friendly empty state)
- [329] “Connect QuickBooks” CTA → Lazarev (clear action)

Implementation:

- shadcn/ui Table component
 - Sortable columns
 - Filter by status (green/yellow/red)
 - Search functionality
 - Responsive: Cards on mobile
 - Read-only (no edit/delete buttons)
 - “View in QuickBooks” opens new tab
-

🎯 Scenario Planning Page [QC: 401-425]

Primary Reference: Ran Liu

Secondary Reference: Lazarev

Why:

- Ran Liu: Clean hierarchy for complex workflows
- Lazarev: Scenario management patterns from their SaaS work

Specific Elements:

Header:

- [401] Page title “Scenario Planning” → Ran Liu
- [402] “Create New Scenario” button → Lazarev (primary CTA)
- [403] View toggle (Grid/List) → Ran Liu (simple toggle)

Scenario Cards:

- [404] Scenario card container → Lazarev (card-based layout)
 - [405] Scenario name → Ran Liu (bold, clear)
 - [406] Status badge (Active/Draft/Locked) → Octet (consistent badges)
 - [407] Created date → Ran Liu (subtle metadata)
 - [408] Quick stats (Cash flow, Risk level) → Ran Liu (compact metrics)
 - [409] Status indicator → Custom (traffic light glow on card border)
 - [410] “Edit” button → Octet (secondary action)
 - [411] “View Report” button → Octet (primary action)
 - [412] “Lock Scenario” toggle → Octet (toggle switch)
 - [413] “Duplicate” button → Octet (tertiary action)
 - [414] “Delete” button → Octet (destructive action, confirm modal)

Comparison View:

- [415] “Compare Scenarios” button → Lazarev
- [416] Scenario selector (multi-select) → Octet
- [417] Side-by-side comparison table → **Ana Vadillo** (financial comparison)
 - [418] Metric rows → Ana Vadillo
 - [419] Scenario columns → Ana Vadillo
 - [420] Difference indicators → Custom (green/red delta)

Empty State:

- [421] No scenarios message → Lazarev
- [422] “Start with AI Strategy Builder” CTA → Lazarev

Filters:

- [423] Filter by status → Octet
- [424] Sort by (Date, Name, Risk) → Octet
- [425] Search scenarios → Ran Liu

Implementation:

- Card grid layout (3 columns desktop, 1 mobile)
 - shadcn/ui Card, Badge, Button components
 - Comparison view = shadcn/ui Table
 - Delete confirmation = shadcn/ui Alert Dialog
 - Drag to reorder (optional)
-

Scenario Editor / Logic Builder [QC: 501-526]

Primary Reference: Octet

Secondary Reference: Ran Liu

Why:

- Octet: Complex flows, interaction patterns
- Ran Liu: Clean hierarchy for dense interfaces

Specific Elements:

Header:

- [501] Scenario name (editable) → Octet (inline edit)
- [502] Save button → Octet (primary action)
- [503] “Back to Scenarios” link → Octet (breadcrumb)
- [504] Status badge → Octet

Left Sidebar (Inputs):

- [505] “Starting Conditions” section → Ran Liu (clear section)
 - [506] Account balances (from QuickBooks) → Ana Vadillo (read-only, financial)
 - [507] Date range selector → Octet (date picker)
 - [508] “Refresh from QuickBooks” button → Ana Vadillo

Center Panel (Logic Rules):

- [509] “Business Rules” section → Octet
 - [510] Rule list → Octet (sortable, draggable)
 - [511] “Add Rule” button → Octet
 - [512] Rule card → Octet
 - [513] IF condition dropdown → Octet (select)
 - [514] Comparison operator → Octet (select: <, >, =, etc.)
 - [515] Value input → Octet (number/text input)
 - [516] THEN action dropdown → Octet (select)
 - [517] Action details → Octet (conditional inputs)
 - [518] “Delete Rule” button → Octet (icon button)
 - [519] Status indicator → Custom (validation: green = valid, red = error)

Right Sidebar (Preview):

- [520] “Cash Flow Preview” section → Fuselab
 - [521] Mini chart → Fuselab (line chart, simplified)
 - [522] Risk indicators → Custom (traffic light badges)
 - [523] “Run Full Simulation” button → Lazarev (CTA)

Bottom Panel (Timeline):

- [524] “Event Timeline” → Fuselab (horizontal timeline)
 - [525] Event markers → Fuselab (dots on timeline)
 - [526] Event details on hover → Fuselab (tooltip)

Implementation:

- Three-column layout (sidebar-main-sidebar)

- Rule builder = Form with shadcn/ui Select, Input components
 - Drag-and-drop rules with react-beautiful-dnd
 - Real-time validation
 - Preview chart = Chart.js line chart
 - Timeline = custom component with CSS
-

AI Strategy Builder Wizard [QC: 601-651]

Primary Reference: Lazarev

Secondary Reference: Octet

Why:

- Lazarev: Onboarding flows, wizard patterns
- Octet: Form patterns, multi-step flows

Specific Elements:

Step 1: Upload Documents [601-610]:

- [601] Wizard progress indicator → Lazarev (step dots)
- [602] Step title “Upload Financial Documents” → Lazarev
- [603] Step description → Lazarev (helpful copy)
- [604] File upload dropzone → Octet (drag-and-drop area)
- [605] File list → Octet (uploaded files with remove button)
- [606] Supported formats note → Ran Liu (subtle helper text)
- [607] “Next” button → Lazarev (primary CTA)
- [608] “Cancel” button → Octet (secondary)
- [609] Upload progress bar → Octet (linear progress)
- [610] Error message → Octet (alert component)

Step 2: AI Analysis [611-620]:

- [611] Step title “Analyzing Documents” → Lazarev
- [612] Loading animation → Lazarev (spinner or skeleton)

- [613] Progress text “Extracting transactions...” → Lazarev
- [614] Analysis results card → Ran Liu (clean summary)
 - [615] Detected income → Ana Vadillo (financial metric)
 - [616] Detected expenses → Ana Vadillo
 - [617] Average monthly cash flow → Ana Vadillo
 - [618] Number of transactions → Ana Vadillo
- [619] “Looks good” button → Lazarev
- [620] “Re-analyze” button → Octet (secondary)

Step 3: Verification [621-635]:

- [621] Step title “Verify Extracted Data” → Lazarev
- [622] Verification table → **Ana Vadillo** (financial data table)
 - [623] Transaction date → Ana Vadillo
 - [624] Description → Ana Vadillo
 - [625] Amount → Ana Vadillo (bold, right-aligned)
 - [626] Category → Ana Vadillo
 - [627] Confidence score → Custom (green/yellow/red indicator)
 - [628] Edit icon → Octet (inline edit)
- [629] “Approve All” button → Lazarev
- [630] “Edit Selected” button → Octet
- [631] Filter by confidence → Octet
- [632] Search transactions → Ran Liu
- [633] Pagination → Octet
- [634] “Next” button → Lazarev
- [635] “Back” button → Octet

Step 4: AI Recommendations [636-645]:

- [636] Step title “Recommended Strategies” → Lazarev
- [637] Recommendation cards → Lazarev (card-based selection)
 - [638] Strategy name → Lazarev (bold, clear)

- [639] Strategy description → Lazarev (concise copy)
- [640] Key benefits list → Lazarev (bullet points)
- [641] Risk level badge → Custom (green/yellow/red)
- [642] “Select This Strategy” radio → Octet
- [643] “Create Scenario” button → Lazarev (primary CTA)
- [644] “Customize” button → Octet (secondary)
- [645] “Back” button → Octet

Step 5: Completion [646-651]:

- [646] Success icon → Lazarev (checkmark animation)
- [647] Success message → Lazarev (celebratory copy)
- [648] Scenario summary → Ran Liu (clean card)
- [649] “View Scenario” button → Lazarev (primary CTA)
- [650] “Create Another” button → Octet (secondary)
- [651] “Go to Dashboard” link → Octet (tertiary)

Implementation:

- Multi-step wizard with shadcn/ui Stepper pattern
 - File upload with react-dropzone
 - AI analysis with loading states
 - Table with shadcn/ui Table component
 - Cards with shadcn/ui Card component
 - Form validation with Zod
 - Success animation with simple CSS
-

Playbook Report View [QC: 701-744]

Primary Reference: Ana Vadillo

Secondary Reference: Fuselab Creative

Why:

- Ana Vadillo: Financial reporting, dense data display
- Fuselab: Data visualization, charts

Specific Elements:

Header:

- [701] Report title → Ran Liu (clear, prominent)
- [702] Scenario name → Ran Liu (subtitle)
- [703] Date range → Ran Liu (metadata)
- [704] “Export PDF” button → Octet (secondary action)
- [705] “Share” button → Octet (secondary action)
- [706] “Back to Scenarios” link → Octet (breadcrumb)

Executive Summary Section:

- [707] Section title “Executive Summary” → Ran Liu
- [708] Summary cards → Ran Liu (metric cards)
 - [709] Starting cash → Ana Vadillo (financial metric)
 - [710] Ending cash → Ana Vadillo
 - [711] Net change → Ana Vadillo (with delta indicator)
 - [712] Risk level → Custom (traffic light badge)

Cash Flow Chart Section:

- [713] Section title “Cash Flow Projection” → Ran Liu
- [714] Chart container → **Fuselab** (primary visualization)
 - [715] Line chart → Fuselab (Chart.js line chart)
 - [716] Legend → Fuselab (chart legend)
 - [717] Axis labels → Fuselab (clear labeling)
 - [718] Tooltip on hover → Fuselab (data point details)
- [719] Chart controls → Fuselab (zoom, pan, reset)

Risk Analysis Section:

- [720] Section title “Risk Analysis” → Ran Liu

- [721] Risk table → Ana Vadillo (financial risk table)
 - [722] Risk description → Ana Vadillo
 - [723] Severity → Custom (traffic light indicator)
 - [724] Date detected → Ana Vadillo
 - [725] Recommended action → Ana Vadillo
 - [726] Status → Custom (resolved/active badge)

Transaction Register Section:

- [727] Section title “Detailed Register” → Ran Liu
- [728] Register table → **Ana Vadillo** (finance-grade table)
 - [729] Date → Ana Vadillo
 - [730] Description → Ana Vadillo
 - [731] Category → Ana Vadillo
 - [732] Debit → Ana Vadillo (red, right-aligned)
 - [733] Credit → Ana Vadillo (green, right-aligned)
 - [734] Balance → Ana Vadillo (bold, right-aligned)
 - [735] Source → Ana Vadillo (QuickBooks/Manual badge)
 - [736] Status → Custom (traffic light)
- [737] Filter by category → Octet
- [738] Search transactions → Ran Liu
- [739] Pagination → Octet

AI Insights Section:

- [740] Section title “AI Insights & Recommendations” → Lazarev
- [741] Insight cards → Lazarev (card-based layout)
 - [742] Insight text → Lazarev (readable copy)
 - [743] Confidence score → Custom (percentage with color)
 - [744] “Apply Recommendation” button → Lazarev (CTA)

Implementation:

- Long-form report layout

- Print-friendly CSS
 - Charts with Chart.js
 - Tables with shadcn/ui Table
 - Export PDF with html2pdf.js or similar
 - Responsive: Stacks on mobile
 - Sticky table of contents sidebar (desktop)
-

Settings & Notifications [QC: 801-823]

Primary Reference: Octet

Secondary Reference: Ran Liu

Why:

- Octet: Settings patterns, form-heavy pages
- Ran Liu: Clean, organized layout

Specific Elements:

Sidebar Navigation:

- [801] Settings sidebar → Octet (vertical tab navigation)
 - [802] General tab → Octet
 - [803] QuickBooks Connector tab → Octet
 - [804] Notifications tab → Octet
 - [805] Audit Trail tab → Octet
 - [806] QC Numbers tab → Octet (dev mode)

General Settings:

- [807] Section title “General Settings” → Ran Liu
- [808] App title input → Octet (text input)
- [809] Theme toggle (Light/Dark) → Octet (switch)
- [810] Language selector → Octet (select dropdown)
- [811] “Save Changes” button → Octet (primary action)

QuickBooks Connector:

- [812] Section title “QuickBooks Connector” → Ran Liu
- [813] Connection status → **Ana Vadillo** (financial integration status)
- [814] “Connect QuickBooks” button → Lazarev (primary CTA if not connected)
- [815] “Disconnect” button → Octet (destructive action if connected)
- [816] Sync frequency selector → Octet (radio group: 1h, 4h, 8h, 24h)
- [817] “Sync on Login” toggle → Octet (switch)
- [818] Last synced timestamp → Ana Vadillo

Notifications:

- [819] Section title “Notification Settings” → Ran Liu
- [820] Notification preferences → Octet (checkbox list)
 - Email on new risk detected
 - Email on overdue task
 - Email on sync failure
 - Email on scenario completion
- [821] Overdue threshold input → Octet (number input with unit)
- [822] “Save Preferences” button → Octet

QC Numbers (Dev Mode):

- [823] “Show QC Numbers” toggle → Octet (switch)

Implementation:

- Two-column layout (sidebar + content)
 - shadcn/ui Tabs for navigation
 - Forms with shadcn/ui Form components
 - Toggle switches with shadcn/ui Switch
 - Save confirmation toast
 - Validation with Zod
-

Audit Trail Report [QC: 901-920]

Primary Reference: Ana Vadillo

Secondary Reference: Fuselab Creative

Why:

- Ana Vadillo: Dense data tables, financial logging
- Fuselab: Timeline visualization

Specific Elements:

Header:

- [901] Page title “Audit Trail” → Ran Liu
- [902] Date range filter → Octet (date range picker)
- [903] User filter → Octet (select dropdown)
- [904] Action type filter → Octet (multi-select)
- [905] “Export CSV” button → Octet (secondary action)
- [906] “Clear Filters” button → Octet (tertiary)

Audit Log Table:

- [907] Audit table → **Ana Vadillo** (detailed log table)
 - [908] Timestamp → Ana Vadillo (precise date/time)
 - [909] User → Ana Vadillo (name + role)
 - [910] Action → Ana Vadillo (bold, clear)
 - [911] Entity → Ana Vadillo (what was changed)
 - [912] Before value → Ana Vadillo (strikethrough if changed)
 - [913] After value → Ana Vadillo (highlighted if changed)
 - [914] IP address → Ana Vadillo (security metadata)
 - [915] “View Details” button → Octet (expands row)

Timeline View:

- [916] “Switch to Timeline” toggle → Octet
- [917] Timeline container → **Fuselab** (vertical timeline)

- [918] Event markers → Fuselab (dots on timeline)
- [919] Event cards → Fuselab (card with details)
- [920] Time axis → Fuselab (date labels)

Implementation:

- shadcn/ui Table with expandable rows
 - Filter bar with shadcn/ui components
 - Timeline with custom CSS (vertical line + cards)
 - Pagination for large datasets
 - Export to CSV with papaparse or similar
 - Search functionality
-

Feature Request System [QC: 1001-1050]

Primary Reference: Lazarev

Secondary Reference: Octet

Why:

- Lazarev: User engagement patterns, feedback flows
- Octet: Form patterns, voting UI

Specific Elements:

Feature Request List:

- [1001] Page title “Feature Requests” → Ran Liu
- [1002] “Submit New Request” button → Lazarev (primary CTA)
- [1003] Filter by status → Octet (tabs: All, Submitted, Planned, In Progress, Completed)
- [1004] Sort by → Octet (dropdown: Votes, Date, Status)
- [1005] Search requests → Ran Liu

Request Cards:

- [1006] Request card → Lazarev (card-based layout)

- [1007] Request title → Lazarev (bold, clickable)
- [1008] Status badge → Octet (color-coded)
- [1009] Vote count → Octet (number with up arrow)
- [1010] “Upvote” button → Octet (toggle button)
- [1011] Category tag → Octet (badge)
- [1012] Submitted by → Ran Liu (user name + date)
- [1013] Comment count → Octet (icon + number)
- [1014] “View Details” link → Lazarev

Request Detail Modal:

- [1015] Modal container → Octet (dialog)
 - [1016] Request title → Lazarev
 - [1017] Status badge → Octet
 - [1018] Full description → Lazarev (formatted text)
 - [1019] Attachments → Octet (file list)
 - [1020] Vote button → Octet (prominent)
 - [1021] Comment section → Octet (threaded comments)
 - [1022] “Add Comment” form → Octet (textarea + button)
 - [1023] “Close” button → Octet

Submit Request Form:

- [1024] Form modal → Octet
 - [1025] Title input → Octet (text input)
 - [1026] Description textarea → Octet (rich text)
 - [1027] Category selector → Octet (select)
 - [1028] Attachment upload → Octet (file upload)
 - [1029] “Submit Request” button → Lazarev (primary CTA)
 - [1030] “Cancel” button → Octet

Admin Dashboard (Owner Only):

- [1031] “Admin View” toggle → Octet

- [1032] Admin table → Ana Vadillo (detailed table)
 - [1033] Request title → Ana Vadillo
 - [1034] Votes → Ana Vadillo (sortable)
 - [1035] Status dropdown → Octet (change status)
 - [1036] Assigned to → Octet (user selector)
 - [1037] Priority → Octet (select: Low, Medium, High)
 - [1038] “Merge” button → Octet (merge duplicates)
 - [1039] “Delete” button → Octet (destructive)

Roadmap View:

- [1040] “View Roadmap” link → Lazarev
- [1041] Roadmap timeline → Fuselab (horizontal quarters)
 - [1042] Q1 2026 column → Fuselab
 - [1043] Q2 2026 column → Fuselab
 - [1044] Q3 2026 column → Fuselab
 - [1045] Q4 2026 column → Fuselab
 - [1046] Feature cards in columns → Lazarev (draggable)
- [1047] “Public Roadmap” toggle → Octet (show/hide from users)

Analytics (Admin):

- [1048] Analytics dashboard → Fuselab
 - [1049] Top requested categories chart → Fuselab (bar chart)
 - [1050] Request trend chart → Fuselab (line chart)

Implementation:

- Card grid layout
- shadcn/ui Dialog for modals
- shadcn/ui Form for submission
- Voting with optimistic updates
- Comments with threaded replies
- Admin table with inline editing

- Roadmap with drag-and-drop (react-beautiful-dnd)
 - Charts with Chart.js
-

Calendar View [QC: 1051-1070]

Primary Reference: Fuselab Creative

Secondary Reference: Octet

Why:

- Fuselab: Calendar and timeline visualization
- Octet: Event management patterns

Specific Elements:

Header:

- [1051] Page title “Calendar” → Ran Liu
- [1052] View toggle (Month/Week/Day) → Octet (button group)
- [1053] Today button → Octet (quick navigation)
- [1054] Date navigation (prev/next) → Octet (arrow buttons)
- [1055] Current date display → Ran Liu (prominent)
- [1056] “Sync with Google Calendar” button → Lazarev (integration CTA)

Calendar Grid:

- [1057] Calendar container → **Fuselab** (calendar grid)
 - [1058] Day cells → Fuselab (grid cells)
 - [1059] Event markers → Fuselab (colored dots or bars)
 - [1060] Event titles → Fuselab (truncated text)
 - [1061] Status colors → Custom (green/yellow/red matching traffic light)
 - [1062] “More events” link → Fuselab (if overflow)

Event Detail Popover:

- [1063] Event popover → Octet (popover component)
 - [1064] Event title → Lazarev

- [1065] Event type badge → Octet (Bill, Payment, Task, etc.)
- [1066] Date/time → Ran Liu
- [1067] Status indicator → Custom (traffic light)
- [1068] Description → Lazarev
- [1069] “View in QuickBooks” link → Ana Vadillo (if applicable)
- [1070] “Mark Complete” button → Octet (if task)

Implementation:

- Calendar with react-big-calendar or custom component
 - Event colors match traffic light system
 - Click event to open popover
 - Google Calendar two-way sync via API
 - Responsive: List view on mobile
 - Filter by event type
-

Design System Foundation

Typography (Ran Liu + Octet)

Font Stack:

```
font-family: 'Inter', -apple-system, BlinkMacSystemFont, 'Segoe UI', sans-serif;
```

Scale:

- Display: 2.5rem (40px) - Page titles
- Heading 1: 2rem (32px) - Section titles
- Heading 2: 1.5rem (24px) - Subsection titles
- Heading 3: 1.25rem (20px) - Card titles
- Body: 1rem (16px) - Default text

- Small: 0.875rem (14px) - Metadata, captions
- Tiny: 0.75rem (12px) - Labels, badges

Weights:

- Bold (700) - Primary headings, important numbers
- Semibold (600) - Secondary headings, table headers
- Medium (500) - Buttons, labels
- Regular (400) - Body text

Color Palette (Ana Vadillo + Ran Liu)

Base Colors:

```

/* Neutrals (Ran Liu - calm palette) */
--background: 0 0% 100%;           /* White */
--foreground: 222 47% 11%;         /* Near black */
--muted: 210 40% 96%;            /* Light gray */
--muted-foreground: 215 16% 47%;  /* Medium gray */
--border: 214 32% 91%;          /* Border gray */

/* Primary (Ana Vadillo - financial blue) */
--primary: 221 83% 53%;          /* Professional blue */
--primary-foreground: 210 40% 98%; /* White text on blue */

/* Status Colors (Traffic Light System) */
--success: 142 76% 36%;          /* Green - In sync */
--warning: 38 92% 50%;           /* Amber - Warning */
--destructive: 0 84% 60%;         /* Red - Critical */

/* Financial Colors (Ana Vadillo) */
--credit: 142 76% 36%;           /* Green for income/credits */
--debit: 0 84% 60%;              /* Red for expenses/debits */

```

Traffic Light Glows:

```

/* Green glow */
box-shadow: 0 0 0 3px rgba(34, 197, 94, 0.2);

/* Yellow glow */
box-shadow: 0 0 0 3px rgba(251, 191, 36, 0.3);

/* Red glow (with pulse) */
box-shadow: 0 0 0 3px rgba(239, 68, 68, 0.3);
animation: pulse 2s cubic-bezier(0.4, 0, 0.6, 1) infinite;

```

Spacing (Octet + Ran Liu)

Scale (Tailwind default):

- 0.5 = 0.125rem (2px)
- 1 = 0.25rem (4px)
- 2 = 0.5rem (8px)
- 3 = 0.75rem (12px)
- 4 = 1rem (16px)
- 6 = 1.5rem (24px)
- 8 = 2rem (32px)
- 12 = 3rem (48px)
- 16 = 4rem (64px)

Component Spacing:

- Card padding: 6 (24px)
- Section spacing: 12 (48px)
- Element spacing: 4 (16px)
- Button padding: 4 horizontal, 2 vertical

Components (shadcn/ui + Custom)

From shadcn/ui:

- Button (with variants: default, secondary, outline, ghost, destructive)
- Card (with header, content, footer)
- Table (with sorting, filtering)
- Form (with validation)
- Dialog / Modal
- Dropdown Menu
- Select
- Input
- Textarea
- Switch / Toggle
- Badge
- Alert
- Tooltip
- Tabs
- Popover
- Sheet (sidebar)

Custom Components:

- Traffic Light Status Indicator
 - Widget Container (for dashboard)
 - Chart Wrapper (for data viz)
 - Timeline Component
 - Calendar Component
 - File Upload Dropzone
-



UI Flow Summary

User Journey with Designer References

1. Login (Lazarev)

- Clean, professional entry point
- Single Google OAuth button
- Minimal distractions

2. Dashboard (Fuselab + Ran Liu)

- Customizable widget grid
- One primary chart (cash flow)
- Traffic light status indicators throughout
- Quick access to all features

3. Accounts Management (Ana Vadillo)

- Finance-grade tables
- Clear hierarchy
- Read-only QuickBooks data
- “View in QuickBooks” links

4. Scenario Planning (Ran Liu + Lazarev)

- Card-based scenario selection
- Clear status indicators
- Comparison view for analysis

5. Scenario Editor (Octet + Ran Liu)

- Complex logic builder
- Clean, organized interface
- Real-time validation
- Preview panel

6. AI Strategy Builder (Lazarev + Octet)

- Guided wizard flow
- Clear progress indication
- Verification step
- AI recommendations

7. Playbook Report (Ana Vadillo + Fuselab)

- Comprehensive financial report
- Charts and tables
- AI insights
- Export functionality

8. Settings (Octet + Ran Liu)

- Organized tab navigation
- Clear form patterns
- QuickBooks connector management

9. Audit Trail (Ana Vadillo + Fuselab)

- Detailed log table
- Timeline visualization
- Filter and search

10. Feature Requests (Lazarev + Octet)

- Community feedback portal
- Voting system
- Roadmap view

11. Calendar (Fuselab + Octet)

- Event visualization
 - Google Calendar sync
 - Status-coded events
-

Implementation Confidence by Page

Page	Primary Designer	Confidence	Notes
Login	Lazarev	95%	Simple, clean pattern
Navigation	Octet	95%	Standard component-based
Dashboard	Fuselab + Ran Liu	90%	React Grid Layout proven
Accounts	Ana Vadillo	92%	Tables are my strength
Scenario Planning	Ran Liu + Lazarev	93%	Card layouts straightforward
Scenario Editor	Octet + Ran Liu	88%	Complex but doable
AI Wizard	Lazarev + Octet	90%	Multi-step form pattern
Report	Ana Vadillo + Fuselab	90%	Tables + charts
Settings	Octet + Ran Liu	95%	Standard settings patterns
Audit Trail	Ana Vadillo + Fuselab	92%	Table + timeline
Feature Requests	Lazarev + Octet	90%	Card + voting pattern
Calendar	Fuselab + Octet	85%	Calendar library available

Overall Confidence: 91%

Development Prompt

When I begin Phase 1 development, I will use this exact prompt internally:

"Build the Financial Playbook application using the following UI design references:

Foundation: Octet's component-based design system approach + Ran Liu's clean, hierarchical layouts with calm color palettes

Financial Specifics: Ana Vadillo's finance-grade tables with strong typographic hierarchy and carefully constrained colors for financial data display

Dashboard & Visualization: Fuselab Creative's configurable widget-based dashboards and data visualization patterns

Polish & Flows: Lazarev's professional fintech aesthetic, onboarding flows, and empty states (applied selectively)

Technical Stack: React 19, Tailwind CSS 4, shadcn/ui components, Chart.js for visualization, React Grid Layout for dashboard

Visual Language: Traffic light status system (green/yellow/red glows) applied consistently across all pages. Clean, professional, data-dense without clutter. Component-based architecture for consistency and scalability."

Final Selections Summary

My 4 Primary Designer/Agency References:

1. **Octet** (95% confidence) - Design system foundation, component consistency
2. **Ran Liu** (95% confidence) - Clean hierarchy, layout structure
3. **Ana Vadillo** (92% confidence) - Financial tables, dense data display
4. **Fuselab Creative** (90% confidence) - Dashboard, widgets, data visualization

Polish Layer:

1. **Lazarev** (92% confidence, selective) - Professional fintech aesthetic, empty states, onboarding

Overall Implementation Confidence: 91%

No complexities beyond my capabilities with this combination.

Next Steps

1.  Designer selections finalized
2.  Begin Phase 1: Set up design system foundation
3.  Build component library based on Octet + Ran Liu

4.  Implement dashboard with Fuselab patterns
5.  Create financial tables with Ana Vadillo patterns
6.  Apply Lazarev polish selectively
7.  Implement traffic light status system
8.  Complete all 11 pages
9.  Test, refine, deploy

Ready to begin Phase 1 development with these exact UI references.

DEVELOPMENT PLAN

Phase 1: Project Setup & Basic Authentication (1 week)

Goals:

- Initialize project with all dependencies
- Implement Google OAuth 2.0
- Create basic user profile page
- Set up security headers

Tasks:

- Initialize Next.js project with TypeScript and Tailwind CSS
- Set up PostgreSQL database and Drizzle ORM
- Create GitHub repository
- Implement Google OAuth 2.0 authentication
- Create user registration and login flows
- Build basic user profile page
- Implement HTTPS and security headers (CSP, HSTS)
- Set up development mode flag for UI QC numbers

Deliverables:

- Working authentication system

- User can log in with Google
 - Basic profile page
 - Secure HTTPS connection
-

Phase 2: QuickBooks Integration & Financial Data Engine (2 weeks)

Goals:

- Connect to QuickBooks Online API
- Implement real-time data sync
- Build Accounts Management UI
- Configure sync frequency settings

Tasks:

- Extend database schema for accounts, transactions, bills
- Create quickbooks_connections table with encrypted tokens
- Implement QuickBooks OAuth 2.0 connection flow
- Build QuickBooks data sync service
- Implement QuickBooks webhook receiver for real-time updates
- Build Accounts Management UI with connection status
- Add “Last Synced” timestamp and “Sync Now” button
- Implement configurable sync frequency in Settings
- Implement automatic sync on login
- Build Dashboard with account summaries and quick stats

Deliverables:

- QuickBooks connection working
 - Real-time data sync
 - Accounts Management page with READ-ONLY data
 - Dashboard with financial overview
-

Phase 3: AI Strategy Builder with Math Validation (2-3 weeks)

Goals:

- Build document upload and processing pipeline
- Implement AI-powered data extraction
- Add math validation layer
- Create strategy recommendation engine

Tasks:

- Extend database schema for uploaded_documents, scenarios
- Build Strategy Builder Wizard UI (3-step process)
- Implement file upload interface with drag-and-drop
- Implement document processing pipeline (OCR, text extraction)
- Build AI-powered data extraction service
- Build separate validation service for AI math verification**
- Implement transaction categorization
- Implement recurring bill identification
- Build AI-powered strategy recommendation engine
- Add discrepancy flagging UI for failed validations

Deliverables:

- Working document upload system
- AI extracts financial data from statements
- All AI calculations verified by traditional code
- 3 tailored strategy recommendations
- Automatic playbook creation

Phase 4: Logic Simulator & Validated Report Generator (2-3 weeks)

Goals:

- Build scenario logic editor
- Implement day-by-day simulation engine
- Create AI-powered report generator
- Validate all report calculations

Tasks:

- Extend database schema for simulation_results
- Build Scenario Logic Editor UI with visual rule builder
- Implement condition and action definition system
- Build day-by-day simulation engine
- Implement event processing (income, bills, user-defined rules)
- Implement risk flagging system
- Build Report Generator with AI analysis
- Implement AI Math Validation for all report calculations**
- Build AI-powered root cause analysis
- Build AI-generated solution recommendations
- Implement report assembly (Executive Summary, Running Balances, etc.)
- Build chart generation for reports
- Implement report export (PDF, Word, Markdown, CSV)

Deliverables:

- Visual logic editor for building rules
 - Working simulation engine
 - Comprehensive reports with AI analysis
 - All calculations validated
-

Phase 5: Automation, Audit Trail & Final UI (1-2 weeks)

Goals:

- Implement automation system

- Build audit trail
- Complete all UI pages with QC numbers
- Add visual status system
- Implement customizable dashboard
- Integrate Google Calendar

Tasks:

- Extend database schema for audit_logs
- Update bills/transactions tables with automation fields
- Build simplified automation system for recurring bills
- Implement price variance alerts (default 3%)
- Build Automation Settings UI modal
- Implement notification engine for email alerts
- Build Notification Settings UI in Settings page
- Implement audit trail logging middleware
- Build Audit Trail Report UI with filters
- Implement CSV export for audit logs
- Complete all UI pages with QC numbering system
- Assign unique 3-digit QC numbers to all UI elements
- Implement development mode toggle for QC visibility
- Implement Visual Status System (Green/Yellow/Red)**
- Build Customizable Dashboard Layout (drag-and-drop)**
- Integrate Google Calendar API**
- Build Calendar View page**
- Build Feature Request System**
- Ensure responsive design across all devices
- Implement comprehensive error handling
- Add loading states for all async operations
- Conduct accessibility audit

Deliverables:

- Automation system working
 - Comprehensive audit trail
 - All UI pages complete with QC numbers
 - Visual status system applied everywhere
 - Customizable dashboard
 - Google Calendar integration
 - Feature Request System
 - Responsive, accessible UI
-

Phase 6: Security Review, Testing & Deployment (1 week)

Goals:

- Conduct security audit
- Test all features
- Deploy to production
- Create documentation

Tasks:

- Conduct comprehensive security audit
- Test for SQL injection vulnerabilities
- Test for XSS vulnerabilities
- Verify all API endpoints are protected
- Test rate limiting
- Review all environment variable usage
- Set up production environment on Vercel
- Configure environment variables in Vercel
- Set up PostgreSQL production database
- Configure custom domain (if applicable)

- Set up GitHub Actions workflow for automated testing
- Configure automatic deployment to Vercel
- Implement Vercel Analytics
- Set up Sentry for error tracking
- Create user documentation
- Create admin guide
- Create troubleshooting guide
- Conduct final user acceptance testing
- Create mechanism to remove QC numbers for production
- Deploy to production
- Conduct handover session

Deliverables:

- Fully tested application
 - Production deployment on Vercel
 - Complete documentation
 - Monitoring and analytics in place
-

SECURITY & COMPLIANCE

Authentication & Authorization

- **OAuth 2.0** for Google authentication
- **JWT tokens** for session management (httpOnly cookies)
- **CSRF protection** via SameSite cookies
- **Rate limiting** on all API endpoints (100 req/min per user)
- **Role-based access control** (user, admin)

Data Security

- **Encryption at rest:** AES-256 for QuickBooks tokens
- **Encryption in transit:** HTTPS only (enforced by Vercel)
- **SQL injection prevention:** Parameterized queries via Drizzle ORM
- **XSS prevention:** React auto-escaping + Content Security Policy
- **Row-level security:** All queries filtered by userId

API Security

- **Environment variables** for all secrets (never committed)
- **Server-side API calls only** (no client-side API keys)
- **Webhook signature validation** for QuickBooks webhooks
- **Token refresh** automatic before expiration

Audit Trail

- **Immutable logs** (append-only)
- **Before/after snapshots** for all changes
- **Database triggers** for automatic logging
- **Checksums** for tamper detection

Compliance

- **GDPR-ready:** User data export and deletion
 - **SOC 2 considerations:** Audit trail, encryption, access controls
 - **PCI DSS:** No credit card data stored (QuickBooks handles payments)
-

DEPLOYMENT STRATEGY

Hosting Platform: Vercel

Why Vercel:

- Seamless Next.js integration
- Serverless functions for backend
- Built-in CI/CD pipeline
- Edge network for global performance
- Automatic HTTPS
- Easy environment variable management

Deployment Process

Development:

1. Code changes pushed to GitHub
2. Manus platform syncs to GitHub repo
3. Local testing in sandbox environment

Staging:

1. Push to `develop` branch
2. GitHub Actions runs tests
3. Auto-deploy to Vercel staging environment
4. Manual QA testing

Production:

1. Merge `develop` to `main` branch
2. GitHub Actions runs full test suite
3. Auto-deploy to Vercel production
4. Smoke tests run automatically
5. Rollback available if issues detected

Environment Variables

Development (`.env.local`):

```
DATABASE_URL=mysql://localhost:3306/financial_playbook_dev
JWT_SECRET=dev_secret_key
QUICKBOOKS_CLIENT_ID=dev_client_id
QUICKBOOKS_CLIENT_SECRET=dev_client_secret
GOOGLE_CLIENT_ID=dev_google_id
GOOGLE_CLIENT_SECRET=dev_google_secret
OPENAI_API_KEY=dev_openai_key
SENDGRID_API_KEY=dev_sendgrid_key
AWS_ACCESS_KEY_ID=dev_aws_key
AWS_SECRET_ACCESS_KEY=dev_aws_secret
S3_BUCKET_NAME=dev-financial-playbook
```

Production (Vercel Dashboard):

- All secrets managed via Vercel dashboard
- Encrypted at rest
- Never exposed to client
- Automatic injection at runtime

Database

Development: MySQL/TiDB (Manus-provided)

Production: PostgreSQL (Vercel Postgres)

Migration Strategy:

1. Export schema from development
2. Convert MySQL syntax to PostgreSQL
3. Test migrations in staging
4. Run migrations in production
5. Verify data integrity

Monitoring

Vercel Analytics:

- Page views and user sessions

- Performance metrics (Core Web Vitals)
- Geographic distribution

Sentry Error Tracking:

- Real-time error alerts
- Stack traces and context
- User impact analysis
- Release tracking

Custom Health Checks:

- QuickBooks API connectivity
- Database connectivity
- Email service status
- S3 storage availability

Backup Strategy

Database Backups:

- Automatic daily backups (Vercel Postgres)
- Point-in-time recovery (7 days)
- Manual backups before major changes

Document Backups:

- S3 versioning enabled
 - Cross-region replication (optional)
 - 30-day retention policy
-

FUTURE ROADMAP

Phase 7: Multi-User Expansion (Deferred)

Features:

- User management system
- Delegated access for tax professionals
- Delegate profiles with permissions (QuickBooks access, ACH access, credit card access)
- AI delegation for task assignment
- Email notifications for delegated tasks
- Overdue task alerts (default 3 days, configurable)
- Multi-user audit trail
- Row-level security enforcement

Timeline: Q2-Q3 2026 (after single-user validation)

Phase 8: Advanced Features (Future)

Mobile App:

- iOS and Android native apps
- Push notifications for alerts
- Biometric authentication
- Offline mode with sync

Advanced Reporting:

- Custom report templates
- Scheduled report generation
- Report sharing (read-only links)
- White-label reports for tax pros

Additional Integrations:

- Plaid API for additional bank connections
- Stripe API for payment processing
- Tax software integration (TurboTax, H&R Block)
- Cryptocurrency portfolio tracking

AI Enhancements:

- AI financial advisor chatbot
- Predictive cash flow modeling
- Anomaly detection
- Automated categorization learning

Collaboration:

- Team workspaces
 - Shared scenarios
 - Comments and discussions
 - Activity feed
-

APPENDIX

QC Number Index

Range	Page/Section
100-109	Login/Authentication
200-229	Global Navigation & Dashboard
300-329	Accounts Management
400-425	Scenario Planning
500-541	Scenario Editor/Logic Builder
600-651	AI Strategy Builder Wizard
700-744	Playbook Report View
800-848	Settings & Notifications
900-923	Audit Trail Report
950-966	Calendar View
1000-1050	Feature Request System

Technology References

- **React 19:** <https://react.dev>
- **Next.js:** <https://nextjs.org>
- **Tailwind CSS 4:** <https://tailwindcss.com>
- **tRPC 11:** <https://trpc.io>
- **Drizzle ORM:** <https://orm.drizzle.team>
- **QuickBooks API:** <https://developer.intuit.com>
- **Google OAuth:** <https://developers.google.com/identity/protocols/oauth2>
- **Google Calendar API:** <https://developers.google.com/calendar>
- **OpenAI API:** <https://platform.openai.com>

- **Vercel:** <https://vercel.com>

Glossary

- **Playbook:** A financial scenario with logic rules and projections
- **Scenario:** A what-if analysis of future financial outcomes
- **Logic Rule:** An IF/THEN condition that triggers actions in simulations
- **Simulation:** Day-by-day projection of cash flow based on rules
- **QC Number:** Quality Control number assigned to UI elements for development
- **Status Indicator:** Color-coded visual feedback (Green/Yellow/Red)
- **Math Validation:** Traditional code verification of AI-generated calculations
- **Sync:** Process of updating local data from QuickBooks Online
- **Webhook:** Real-time notification from QuickBooks when data changes
- **Audit Trail:** Immutable log of all user actions and system events

END OF MASTER PLAYBOOK DOCUMENT

Document Version: 1.0

Total Pages: ~100+

Last Updated: November 24, 2025

Status: Complete - Ready for Development

Owner: Superior Networks LLC

Developer: Manus AI Platform

DOCUMENT CHANGE LOG

Date	Version	Changes
11/24/2025	1.0	Initial master playbook created, consolidating all specifications, wireframes, and discussions