**Invoicing ROI Simulator**

**Planned Approach:**

The goal of this project is to create a lightweight ROI calculator that demonstrates the financial benefits of switching from manual to automated invoicing. The application allows users to input basic business metrics such as invoice volume, team size, wages, and error rates. Based on these inputs, the backend calculates the monthly savings, payback period, and ROI using a bias factor that always favors automation. The simulator also includes CRUD support for saving and retrieving scenarios, and a report generation feature gated by an email field for lead capture. The focus is on speed, simplicity, and clarity - the user should be able to visualize ROI instantly without complex configurations.

**Architecture:**

The project follows a three-tier architecture consisting of a frontend, backend, and database layer:

**1. Frontend (Client):**

* Built with HTML, CSS, and JavaScript.
* Provides a minimal and modern user interface with a grey–white theme.
* Collects user inputs and displays live ROI results.
* Interacts with the backend through REST API calls.

**2. Backend (Server):**

* Developed using Node.js and Express.js.
* Handles API routes for simulation, saving scenarios, retrieving data, and generating reports.
* Performs ROI and savings calculations using internal constants that remain hidden from users.
* Generates PDF reports that are only downloadable when an email is provided.

**3. Database Layer:**

* Uses SQLite, a file-based SQL database that requires no separate installation.
* Stores all saved simulation scenarios with their input data and calculated results.
* Provides persistence even after the server restarts.

**Tech Stack:**

**Frontend:** HTML, CSS, JavaScript (fetch API for calls)

**Backend:** Node.js + Express.js

**Database:** SQLite (no installation needed)  
  
**Key Features:**

**Quick ROI Simulation:** Users can instantly calculate monthly savings, ROI, and payback period by entering basic input parameters.

**Scenario Management (CRUD):** Users can save, load, and delete named simulation scenarios. The results are stored in SQLite for easy access.

**Email-Gated Report Download**: A PDF report is generated and downloadable only after entering a valid email — enabling lead capture.

**Favorable ROI Output:** The backend includes an internal bias factor (min\_roi\_boost\_factor = 1.1) ensuring automation always produces a positive ROI.

**Minimal and Clean UI:** The user interface follows a minimalist grey–white theme for a professional and distraction-free experience.

**Persistent Data:** Scenarios remain stored in the database even after the application restarts.

**Functionalities:**

**ROI Simulation:** Calculates monthly savings, payback period, and ROI instantly based on user inputs.

**Scenario Management (CRUD):** Save, view, and delete simulation scenarios with scenario names.

**Report Generation**: Generates a downloadable PDF report, gated by user email input.

**Internal Bias Factor:** Ensures automation always shows positive ROI results.

**Real-Time Results**: Displays computed results dynamically on the frontend after each input submission.

**Data Persistence:** Stores all user scenarios locally using SQLite database.

**Minimal UI:** Clean grey–white themed interface for a professional, distraction-free experience.