

GURU: AI Business Advisor

COMP 490 POC Presentation

Team Future Group :

Rainer Santos, Emmanuel Renderos, Tochukwu Awujo, Jay Patel,
Rajvir Singh

Introduction



Team Future Group :

Rainer – Team Lead • Frontend Dev

Rajvir – Agent Dev • Repo Master

Emmanuel – Security Dev • Researcher

Tochi – Frontend Developer

Jay – Backend Developer

GURU : Project Overview

What is GURU?

- **An AI business advisor built for online entrepreneurs and retailers.**
- **Works like having your own personal AI business guru guiding every step.**
- **Helps users scale operations, grow revenue, and accelerate learning with AI-powered insights.**
- **Makes data-driven business decisions easy, fast, and actionable.**

What GURU Can Do

- **Store and manage inventory in one centralized system**
- **Track sales data and monitor performance in real time**
- **Generate personalized financial reports automatically**
- **Analyze trends, patterns, and profitability across products and categories.**
- **Provide a continuously learning AI advisor tailored to your business operations.**

Problem Statement

Why do you need GURU?

The purpose of the app is to provide an easy way to manage inventory and track user sales. We aimed to integrate AI into our application to enhance the users ability to use the app.

Who is GURU for?

Guru aims to provide help to starting e-commerce shopfronts strategic and personalized guidance towards their fiscal goals, boosting revenue and scale.

How is GURU different?

GURU stands to simplify business analysis and complex decision making with the leverage of AI insights. GURU saves you time, money, and the headache of figuring out how to run your business by yourself.

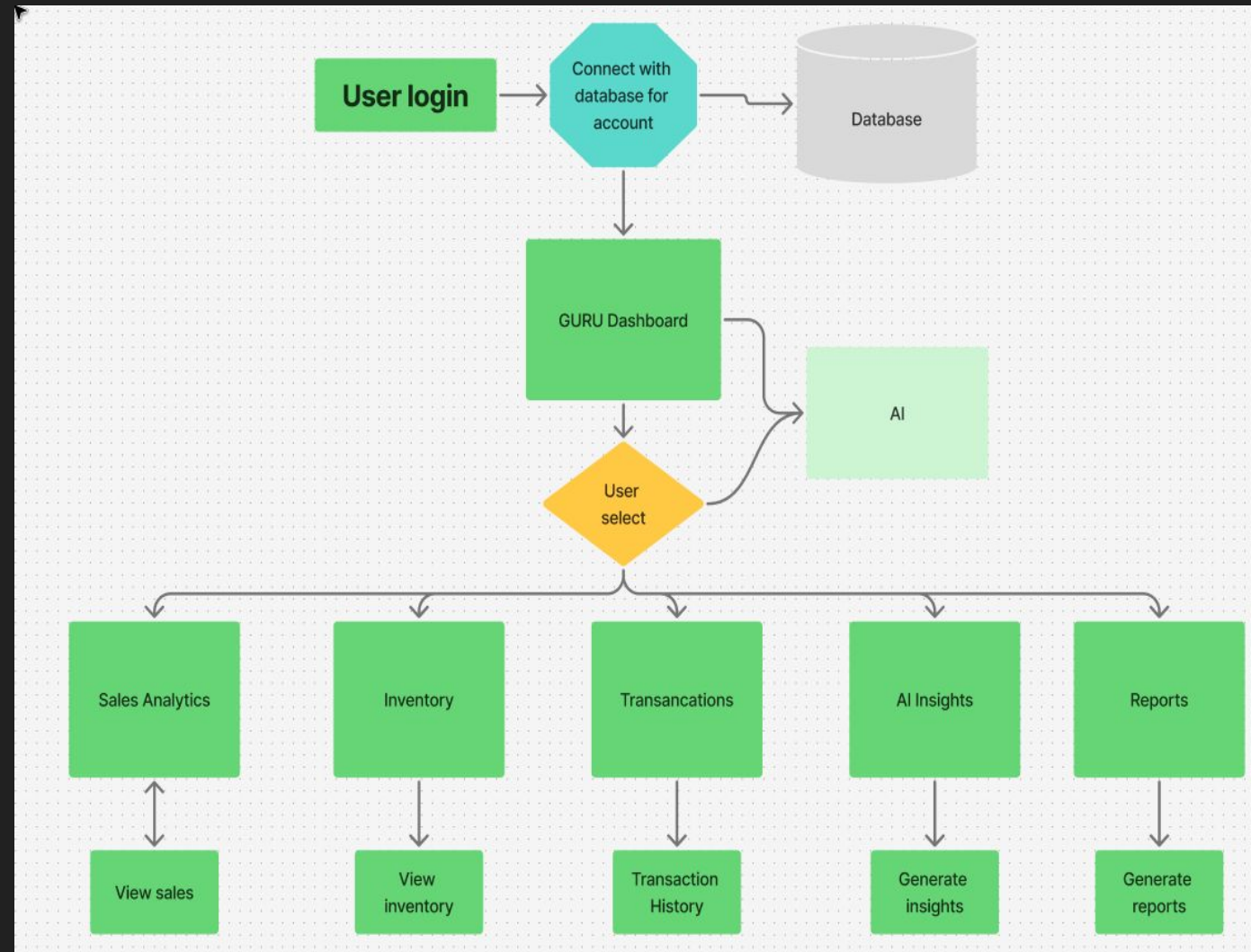
Objective / Goals for PoC

Demonstrate Feasibility

- Showcase the application interface and core functionality.
- Demonstrate user login, authentication, and navigation via the app taskbar.
- Highlight how AI is integrated into the application.

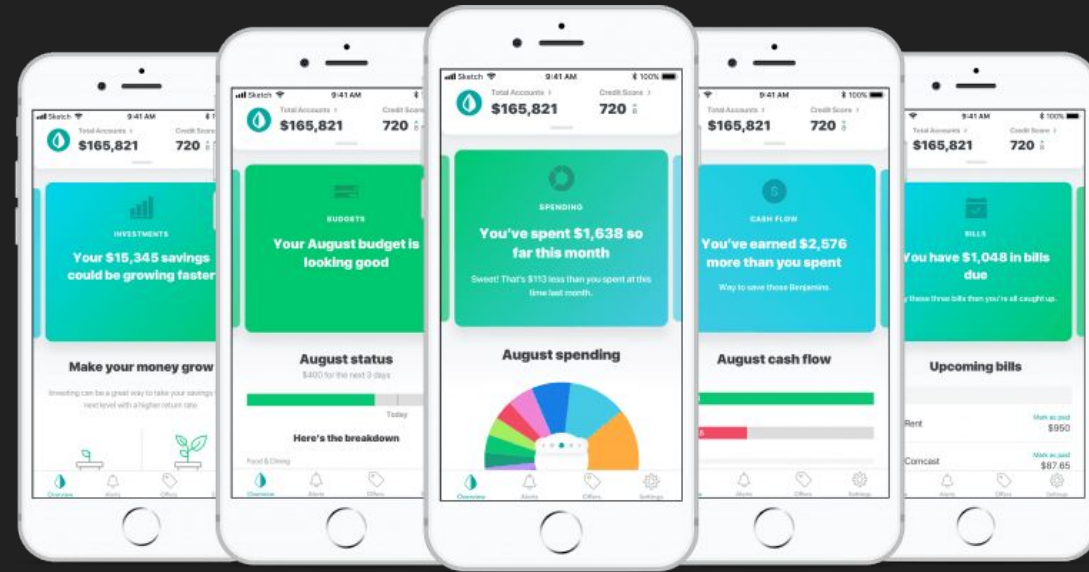
Core Functionalities

- Developed essential features to illustrate key capabilities of the solution.
- Implemented user login and taskbar-based page navigation.
- Enabled users to download a sample AI-generated PDF report.



Key Features / Functionality in GURU

- User is able to sign up with an email and/or login with an existing account
- User Authentication is implemented via supabase
- Sales Analytics Dashboard to analyze business progress and efficiency
- Inventory Management page to quickly sort and optimize load
- AI Report generation for revenue and sales.
- AI GURU Chatbot functionality with personalized 'Insights'



Architecture Overview

- The system is designed with a modular architecture consisting of- **Client/UI**, **API Gateway**, **Service Layer**, and **Database/Storage** components.
- The **Client/UI** handles user interaction, sending requests through the **API Gateway**, which manages routing, security, and communication between modules.
- The **Service Layer** contains the core business logic, ensuring efficient processing of system operations.
- The **Database/Storage module** manages persistent data storage and retrieval, ensuring reliable access to application data.
- **Streamlit** combined with **Supabase** is used for secure backend user authentication and session management.
- We also integrated **Gemini AI** to handle intelligent features, such as automated insights, predictions, or enhanced user support.

Data Flow Representation

Client → API Gateway → Service Layer → Database

Technology Stack Legend :

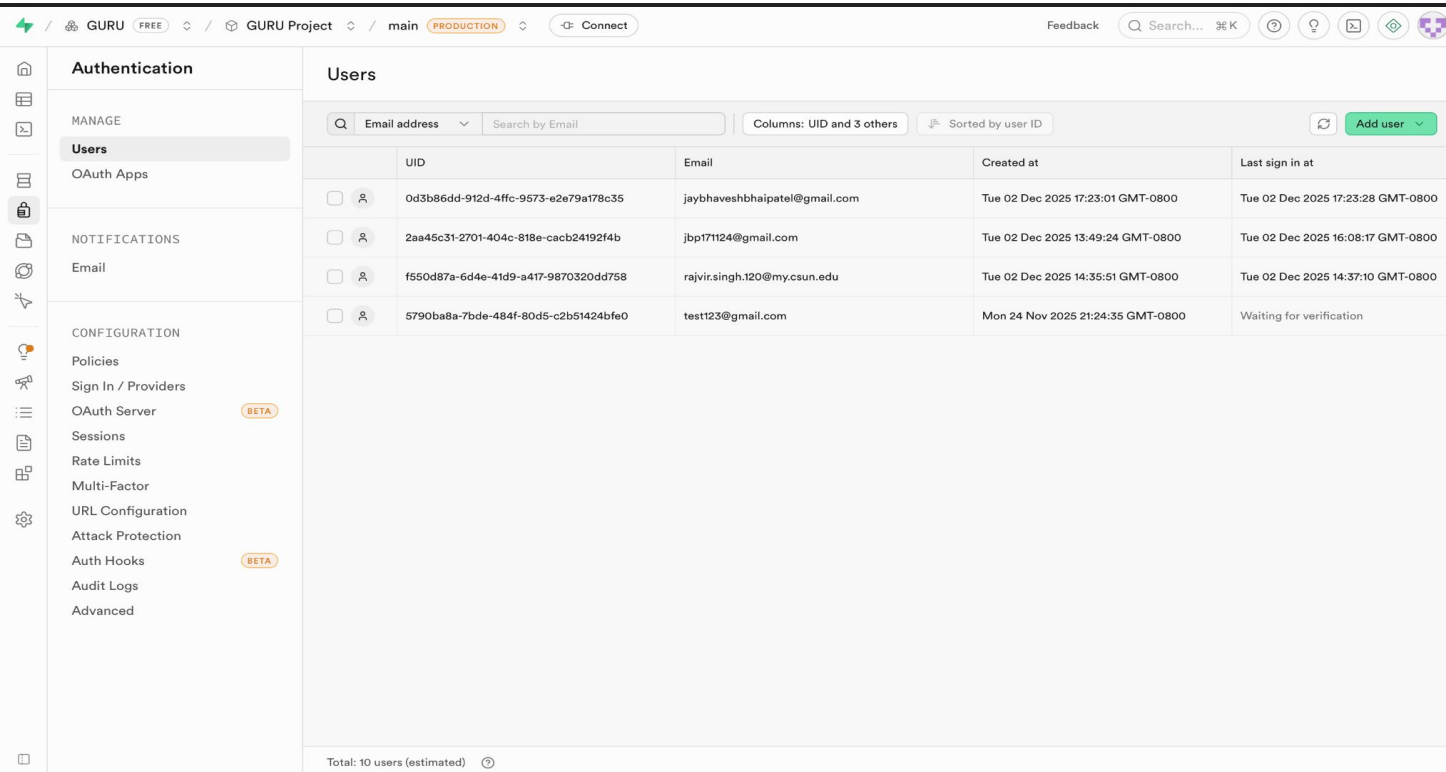
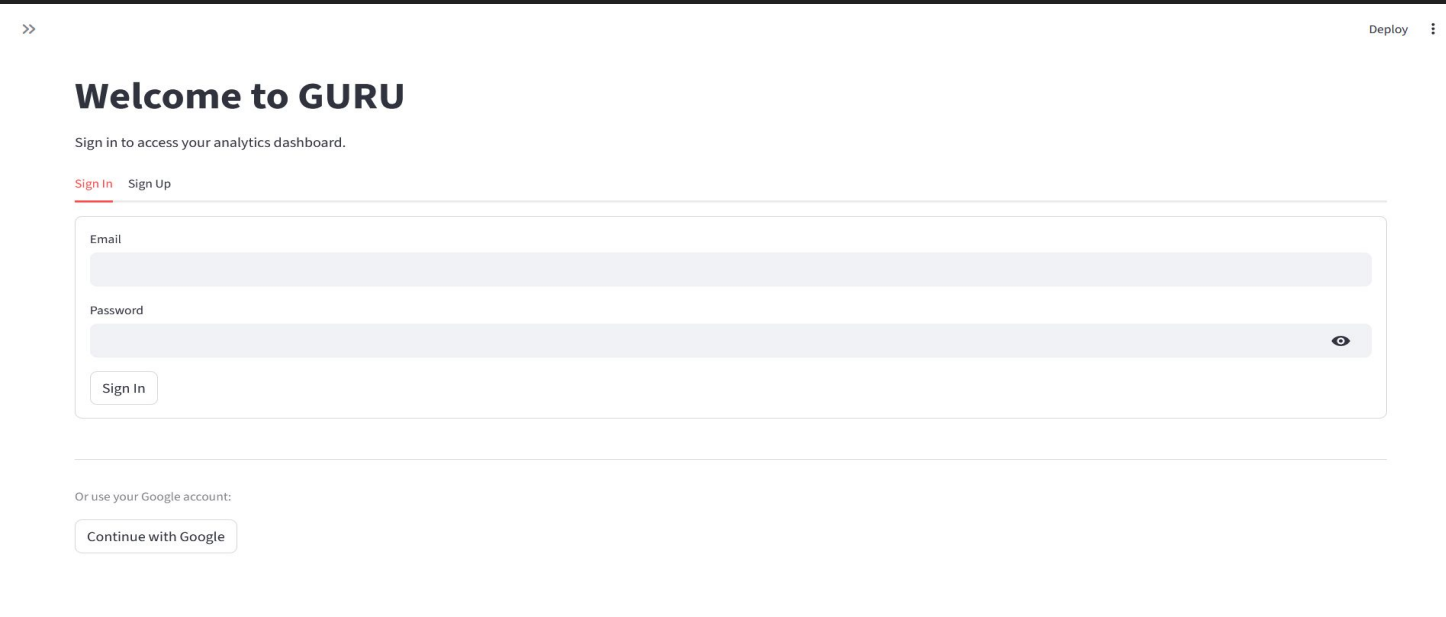
Streamlit – Frontend UI and app framework

Supabase – User authentication and database backend

Python – Main programming language

Pandas – For data processing and analysis

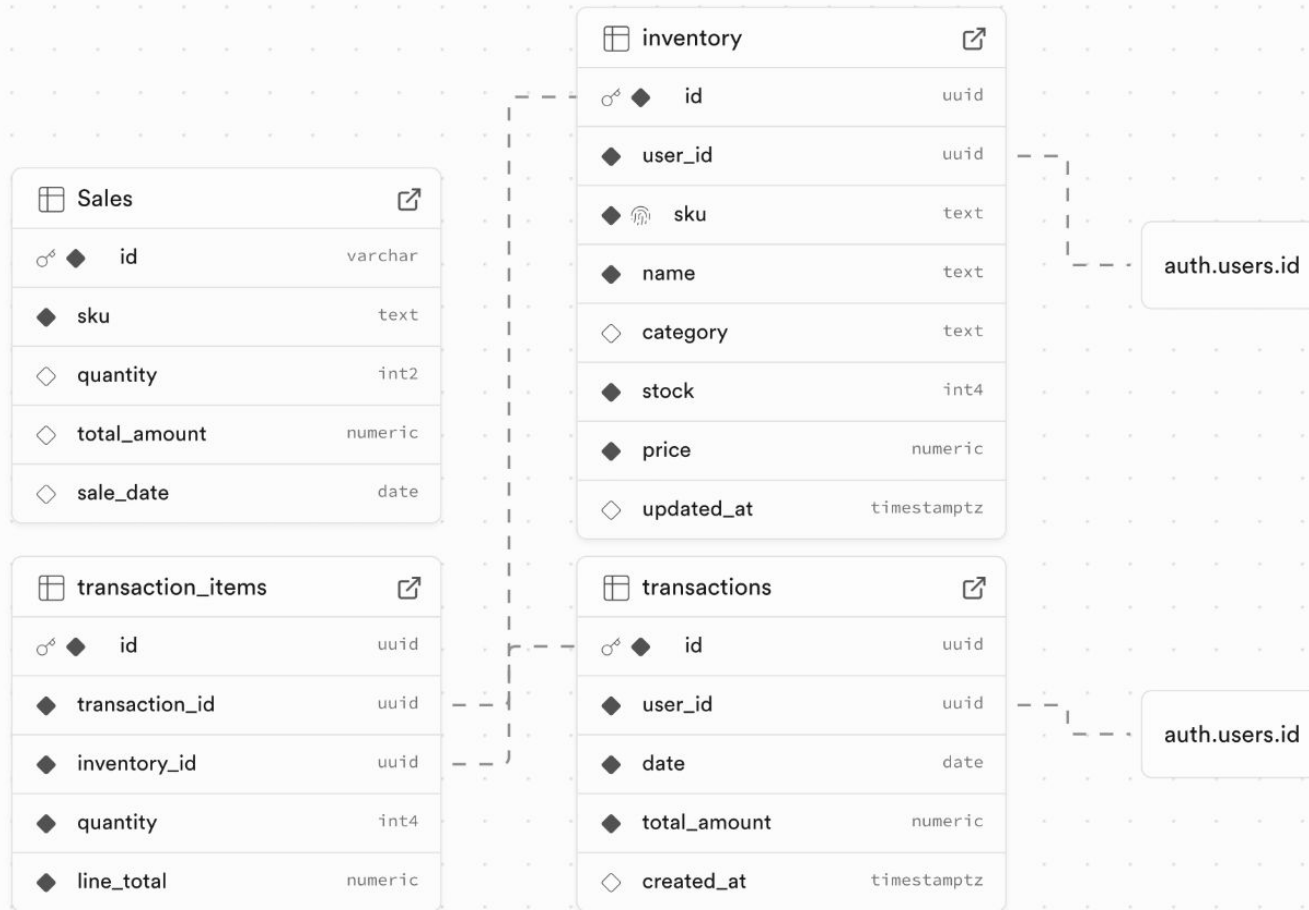
Gemini AI – AI-driven analysis and enhancements



Connected Architecture

Login Page calls Supabase API to verify and authenticate users.

Database Architecture



Primary key # Identity Unique Nullable Non-Nullable

- **User-Linked System**

All data (inventory, transactions, sales) is tied to a specific authenticated user, ensuring secure and personalized data management.

- **Central Inventory Table**

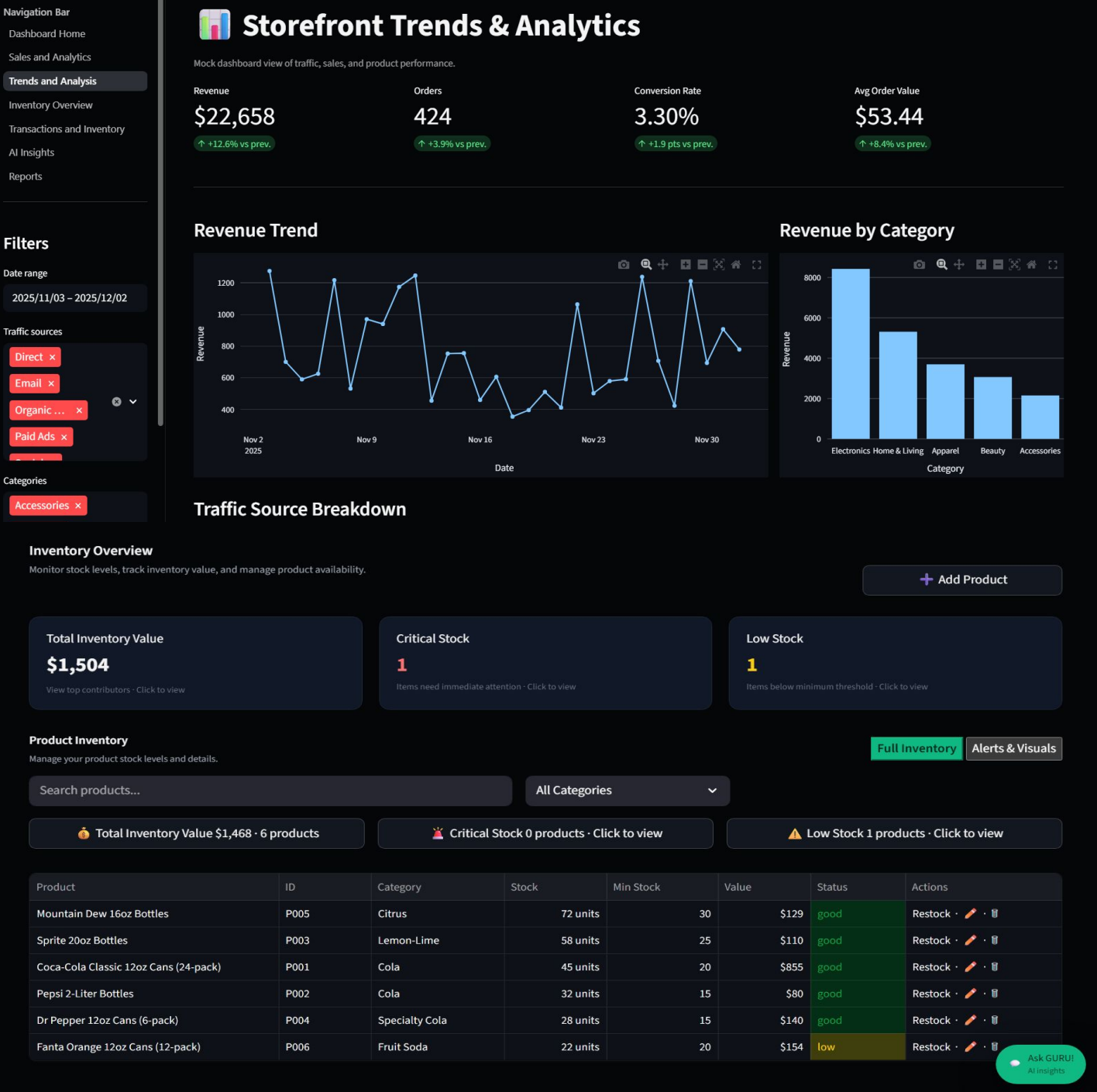
Products are stored in the **inventory** table, containing SKU, name, category, price, and stock for accurate inventory tracking.

- **Transaction Logging**

Every sale is recorded in the **transactions** table with the date and total amount, serving as the main record of business activity.

- **Detailed Transaction Items**

The **transaction_items** table links products to each transaction, showing exactly which items were sold, in what quantity, and at what subtotal.



Design & Mockups

Visual Design Overview

- Our design focuses on a **minimalist and simple interface**

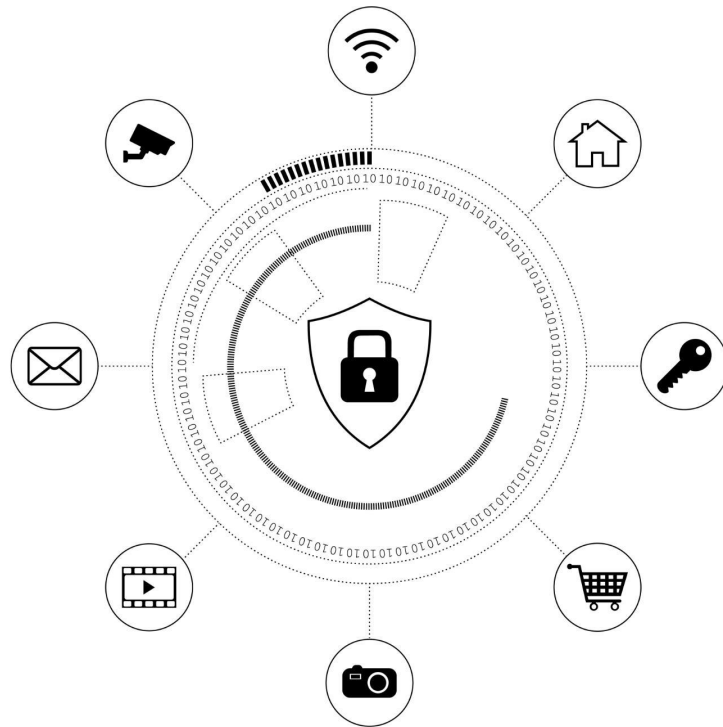
User Experience Considerations

- We based the UI on easy navigation with **minimal clicks** to avoid repetitive actions
- Color choices and contrast are kept simple and readable to support accessibility and reduce eye strain.
- Forms and dashboards are kept uncluttered to make the experience straightforward even for new users.

User-Centered Design Focus

- Each screen was built to help users complete tasks emphasizing speed and optimal viewing.
- The interface flows naturally, so users always know where they are and what to do next.

Legal, Security & Compliance Considerations



- We collect only the data needed to power automatic restocking and refill recommendations.

- We do not sell or share personally identifiable information with third parties for marketing.

Data privacy measures

- Collect only what is necessary to run the app.
- Use anonymized or aggregated data for analytics so individual users cannot be identified.
- Encrypt all stored sensitive data like payment information and addresses to keep your information safe.
- Limit internal visibility so only authorized staff can access specific data needed for their role.
- Give users full control to view, update, or delete their profile and preferences.

Licensing & IP Compliance

Since our app uses third party software, AI models, and datasets. We will manage these assets to stay legally compliant and protect intellectual property

We make sure to use only frameworks and libraries with clear, compatible licenses, such as MIT or Apache 2.0, follow license requirements, including attribution and any usage restrictions.

As well as respect terms of service for vendor APIs and do not attempt to extract or resell their models or data.

Authentication & Authorization

To protect user accounts and internal tools we have very strong authentication. We use short lived session tokens with automatic expiration and refresh. We also support email and password with secureness.

We have to restrict access to sensitive functions such as refunds, order overrides, and model configuration to admin roles only and validate permissions on every API request to ensure users only access their own orders and data.

Lastly we enforce strong password rules and rate limit login attempts as well as monitor for suspicious login behavior and notify users about unusual activity.



Demo Presentation

Feedback & Iteration

UI Styling Improvements :

- Decrease UI clutter as screens felt crowded and visually overwhelming
- Inconsistencies spacing, color hierarchy, and component sizing
- Smoother and Cleaner Mobile UI

Performance Issues:

- Loading times were slower than expected, especially on data-heavy dashboards.
- Incorporate AI agent to reports and insights without slowing down performance
- Faster insights and more immediate visual feedback from interactions.

Next Steps & Future Work

Planned Enhancements

- Styling of UI/UX for mobile for easier user interaction
- Personalized AI Agent memory and inference
- Connecting our app to Shopify via Shopify REST APIs to access the appropriate data.
- Inclusive functionality within E-Commerce Shopfronts
- Host our complete application using Vercel.