MHO B2B E-commerce Platform

A modern, scalable B2B e-commerce platform with real-time Zoho Inventory integration, built with Next.js 16, Prisma, and Docker.



Features

Modern UI/UX

- Beautiful Product Catalog Grid/List views with smooth animations
- Shadcn/Radix UI Modern component library with accessibility
- Responsive Design Mobile-first approach
- Stock Badges Color-coded inventory status (In Stock/Low Stock/Out of Stock)
- Product Modals Quick view with detailed information

S Zoho Integration

- Real-time Sync Automatic product synchronization from Zoho Inventory
- Zoho Books Integration Automatic sales orders and invoice generation
- Branch Auto-registration Branches automatically synced as customers/contacts
- Invoice PDF Generation Automatic PDF creation and viewing
- OAuth 2.0 Secure authentication with token refresh
- Comprehensive Data 50+ product fields synced (pricing, stock, images, etc.)
- Multi-organization Support for multiple Zoho organizations
- Service Account Background sync service for scheduled updates

B2B Features

- Shopping Cart Branch-specific carts with guest support
- Order Management Complete order lifecycle from cart to invoice
- Invoice System Automatic invoice generation with PDF viewing
- Quotations Create, manage, and convert quotations to orders
- Multi-branch Support Manage multiple business locations
- Organization Management Company and branch hierarchy
- Role-based Access Admin, Manager, Customer roles
- **Branch Switching** Easy switching between locations
- Recent Activity Dashboard Real-time activity feed for orders, invoices, and branches

Analytics & Reporting

- Comprehensive Analytics Dashboard Real-time insights into spending and orders
- Financial Metrics Total spent amount, average order value, cost per employee
- Interactive Charts Revenue over time, category breakdown, top products
- Time Range Filters Last 7 days, 30 days, 90 days, 1 year, or custom range
- Branch Filtering Filter analytics by specific branch

- Export Capabilities Export analytics to PDF or CSV
- Latest Orders & Quotations Quick view of recent activity
- Status Breakdown Visual breakdown of orders by status
- AED Currency All financial data displayed in UAE Dirhams

Authentication & Security

- **NextAuth.js** Secure authentication with session management
- Redis Sessions Fast, scalable session storage
- Protected Routes Role-based access control
- Secure API JWT-based authentication

Performance

- **Docker Compose** Complete containerized environment
- PostgreSQL Robust relational database
- Redis Caching Fast session and data caching
- Image Optimization Sharp for image compression
- Optimistic Loading Skeleton states for better UX

Architecture

```
MHO/
                             # Next.js frontend & API
 — apps/web/
    — app/
                             # Next.js 16 App Router
            — арі/
            — quotations/ # Quotation management
               - zoho/  # Zoho integration
- orgs/  # Organization management
oducts/  # Product catalog page
rt/  # Shopping cart page
            orgs/
          - products/
           - cart/
                            # Checkout page
# Orders list & details
# Quotations list & management
# Analytics dashboard
          - orders/
           - checkout/
          — quotations/
          — analytics/
        onboarding/
                            # Admin dashboard
# User onboarding flow
                             # React components
       - components/
                                    # Shadcn UI components
          − ui/
        └─ analytics/
                            # Analytics char
# Utilities & helpers
                                    # Analytics chart components
      - lib/
          — auth.ts
                              # NextAuth configuration
           – prisma.ts
                             # Database client
           - redis.ts
                              # Redis client
                            # Zoho API client
           - zoho.ts
```

Quick Start

Prerequisites

- Docker & Docker Compose
- Node.js 20+ (for local development)
- pnpm
- Zoho Inventory account

1. Clone the Repository

```
git clone https://github.com/Lanioque/MHO-B2B-Ecommerce.git cd MHO-B2B-Ecommerce
```

2. Environment Setup

```
cp apps/web/env.example apps/web/.env
```

Edit apps/web/.env with your credentials:

```
# Database
DATABASE_URL=postgresql://postgres:postgres@localhost:5432/ecommerce_dev

# Redis
REDIS_URL=redis://localhost:6379

# NextAuth
NEXTAUTH_URL=http://localhost:3000
NEXTAUTH_SECRET=your-secret-key-here

# Zoho (Get from Zoho API Console)
ZOHO_CLIENT_ID=your_client_id
ZOHO_CLIENT_SECRET=your_client_secret
ZOHO_ORGANIZATION_ID=your_org_id
ZOHO_REGION=eu # or us, in, au
ZOHO_SCOPE=ZohoInventory.fullaccess.all,ZohoBooks.contacts.CREATE,ZohoBooks.invoices.
```

```
READ
ZOHO_BOOKS_ORGANIZATION_ID=your_books_org_id
```

3. Start with Docker

```
docker-compose up -d
```

The application will be available at:

• Web App: http://localhost:3000

• Prisma Studio: http://localhost:5555

• PostgreSQL: localhost:5432

• Redis: localhost:6379

4. Initial Setup

- 1. Visit http://localhost:3000/register to create an account
- 2. Complete the onboarding process
- 3. Go to http://localhost:3000/test-zoho to connect Zoho
- 4. Click "Sync to Database" to import products

Nevelopment \$\%\text{Development}\$

Local Development (without Docker)

```
# Install dependencies
pnpm install

# Start PostgreSQL & Redis
docker-compose up postgres redis -d

# Run migrations
cd apps/web
pnpm prisma migrate dev

# Start dev server
pnpm dev
```

Database Management

```
# Generate Prisma client
pnpm prisma generate

# Create migration
pnpm prisma migrate dev --name your_migration_name
```

```
# Open Prisma Studio

pnpm prisma studio

# Seed database

pnpm db:seed
```

Building for Production

```
# Build all services
docker-compose build

# Start in production mode
docker-compose up -d
```

Tech Stack

Frontend

- **Next.js 16** React framework with App Router
- React 19 Latest React with React Compiler
- **TypeScript** Type safety
- Tailwind CSS 4 Utility-first CSS
- Shadcn UI Component library
- Radix UI Headless UI primitives
- Lucide Icons Beautiful icon set

Backend

- Next.js API Routes RESTful API
- NextAuth.js Authentication
- Prisma Type-safe ORM
- PostgreSQL Primary database
- Redis Session & cache store
- Zod Runtime validation

Analytics & Charts

- Recharts Interactive charts (line, bar, pie)
- date-fns Date manipulation and formatting
- jsPDF & jsPDF-AutoTable PDF generation for exports
- Papaparse CSV export functionality

DevOps

- **Docker** Containerization
- **Docker Compose** Multi-container orchestration
- pnpm Fast package manager

• ESLint & Prettier - Code quality

♥ API Endpoints

Authentication

- POST /api/auth/register User registration
- POST /api/auth/signout Sign out

Products

- GET /api/products List products (paginated)
- GET /api/products/:id Get product details
- POST /api/products Create product (admin)

Shopping Cart

- GET /api/cart Get user's cart (with branch filter)
- POST /api/cart Add items to cart
- DELETE /api/cart Clear cart
- PATCH /api/cart/items/:itemId Update cart item quantity
- DELETE /api/cart/items/:itemId Remove item from cart

Orders

- GET /api/orders List orders (with filters)
- POST /api/orders Create order from cart
- GET /api/orders/:id Get order details
- PATCH /api/orders/:id Update order status

Invoices

- GET /api/invoices/:id Get invoice details
- GET /api/invoices/:id/pdf Download invoice PDF

Analytics

- GET /api/analytics Get organization analytics (with date range and branch filters)
- GET /api/analytics/export Export analytics to CSV or PDF

Quotations

- GET /api/quotations List quotations (with filters)
- POST /api/quotations Create quotation
- GET /api/quotations/:id Get quotation details
- PATCH /api/quotations/:id Update quotation status or convert to order
- DELETE /api/quotations/:id Delete quotation

Zoho Integration

• GET /api/zoho/oauth/start - Start OAuth flow

- GET /api/zoho/oauth/callback OAuth callback
- GET /api/zoho/products Fetch from Zoho (live)
- POST /api/zoho/sync Sync to database

Organizations

- GET /api/orgs List organizations
- POST /api/orgs Create organization
- GET /api/branches List branches

fig Environment Variables

See apps/web/env.example for all required variables.

Database Schema

Key models:

- User Authentication & profile
- Organization Company entities
- Branch Business locations (with Zoho contact sync)
- Membership User-org relationships with roles
- Product Product catalog (global)
- Cart Shopping carts (branch-specific, guest support)
- CartItem Cart line items
- Order Customer orders (with status tracking)
- OrderItem Order line items
- **Invoice** Generated invoices (linked to orders)
- Quotation Quotations with status lifecycle (DRAFT, SENT, APPROVED, REJECTED, EXPIRED, CONVERTED)
- QuotationItem Quotation line items
- Customer Customer records
- **ZohoConnection** OAuth tokens & settings

All Features Done

✓ Core Features

Substitution & Authorization

- NextAuth.js with Redis sessions
- Role-based access control (OWNER, ADMIN, STAFF, CUSTOMER)
- Secure password hashing
- Session management

- Multi-organization support
- o Branch creation and management
- Employee management per branch

- Address management (billing & shipping)
- Branch-specific operations

✓ Product Catalog

- Product listing with pagination
- Detailed product information (50+ fields)
- Product categories
- Stock status tracking (In Stock/Low Stock/Out of Stock)
- Product images support
- Dietary information and tags
- Product search and filtering

Shopping Cart System

- Branch-specific carts
- Guest cart support (session-based)
- Authenticated user carts
- Quantity management
- o Real-time cart updates
- Cart drawer UI component

• **Order Management**

- Order creation from cart
- Order status tracking (PENDING, AWAITING_PAYMENT, PAID, FAILED, CANCELLED, REFUNDED)
- Order history
- Order details view
- Order items tracking

Invoice System

- Automatic invoice generation
- Invoice PDF viewing
- Invoice status tracking (DRAFT, ISSUED, SENT, PAID, OVERDUE, VOID)
- Linked to orders

Quotation Management

- Quotation creation
- Quotation status lifecycle (DRAFT, SENT, APPROVED, REJECTED, EXPIRED, CONVERTED)
- Convert quotations to orders
- Quotation history
- Expiry date management

Analytics & Reporting

- Comprehensive analytics dashboard
- Revenue over time charts
- Category breakdown (bar charts)
- Top products visualization

- Status pie charts (orders, quotations)
- o Time range filters (7 days, 30 days, 90 days, 1 year, custom)
- Branch filtering
- Export to PDF/CSV
- Latest orders and quotations tables
- o Financial metrics (total spent, average order value, cost per employee)
- AED currency display

☑ Zoho Integration

• **☑** OAuth 2.0 Integration

- Secure OAuth flow
- Token refresh mechanism
- Multi-organization support
- Region support (EU, US, IN, AU)

Zoho Inventory Sync

- Real-time product synchronization
- o 50+ product fields synced
- Stock level updates
- Product image sync
- Scheduled sync service
- Manual sync trigger

Zoho Books Integration

- Automatic sales order creation
- Automatic invoice generation
- Branch auto-registration as contacts/customers
- Invoice PDF generation
- o Customer management sync

✓ User Interface

Modern UI Components

- Shadcn/Radix UI components
- Responsive design (mobile-first)
- Dark mode support (implicit via theme)
- Beautiful animations
- Skeleton loading states
- Toast notifications

Pages & Routes

- Product catalog page
- Shopping cart page
- Checkout page

- Order listing and details
- Quotation management
- Analytics dashboard
- Admin dashboard
- Organization management
- Branch management
- User onboarding flow
- Login/Register pages

✓ Technical Features

Database & ORM

- PostgreSQL database
- o Prisma ORM with type safety
- Database migrations
- Seed scripts

Caching & Performance

- Redis session storage
- Redis caching
- Image optimization with Sharp
- o Optimistic UI updates

API & Backend

- RESTful API routes
- Request validation (Zod)
- Error handling middleware
- Type-safe API responses

✓ DevOps & Deployment

- Docker containerization
- Docker Compose setup
- o Environment configuration
- Production build setup

User Stories

As a Customer (Branch Manager)

- 1. **As a customer**, I want to browse products in a catalog so that I can see available items with prices and stock status.
- 2. As a customer, I want to add products to my cart so that I can prepare an order.
- 3. **As a customer**, I want to manage cart items (quantity, remove) so that I can adjust my order before checkout.
- 4. As a customer, I want to place orders from my cart so that I can purchase items for my branch.
- 5. **As a customer**, I want to view my order history so that I can track my past purchases.

6. **As a customer**, I want to view order details including status so that I know when my order will be delivered.

- 7. **As a customer**, I want to view invoices so that I can keep records for accounting.
- 8. **As a customer**, I want to create quotations so that I can prepare quotes before placing orders.
- 9. **As a customer**, I want to convert quotations to orders so that I can quickly place orders from approved quotes.
- 10. As a customer, I want to switch between branches so that I can manage orders for different locations.

As an Admin/Staff

- 1. As an admin, I want to manage organizations so that I can set up new client companies.
- 2. **As an admin**, I want to manage branches so that I can configure multiple locations per organization.
- 3. As an admin, I want to view all orders across the organization so that I can monitor business activity.
- 4. As an admin, I want to update order status so that customers know the progress of their orders.
- 5. **As an admin**, I want to view analytics dashboard so that I can understand business performance.
- 6. As an admin, I want to export analytics data so that I can create reports for stakeholders.
- 7. **As an admin**, I want to filter analytics by branch and date range so that I can analyze specific periods and locations.
- 8. **As an admin**, I want to manage quotations so that I can track sales opportunities.
- 9. As an admin, I want to view employee information so that I can manage workforce.
- 10. **As an admin**, I want to access role-based features so that I can perform administrative tasks.

As a System Administrator

- 1. **As a system admin**, I want to connect Zoho Inventory so that products are automatically synced.
- 2. As a system admin, I want products to sync automatically so that inventory stays up-to-date.
- 3. **As a system admin**, I want orders to be created in Zoho Books so that accounting is automated.
- 4. As a system admin, I want invoices to be generated automatically so that billing is streamlined.
- 5. As a system admin, I want branches to be registered in Zoho Books so that they appear as customers.
- 6. As a system admin, I want to manage Zoho OAuth tokens so that integration stays connected.

As a Guest User

- 1. **As a guest**, I want to browse products so that I can see what's available before registering.
- 2. As a quest, I want to add items to a cart so that I can prepare an order before creating an account.

& Roadmap

Phase 1: Core Enhancements (Q1 2025)

Payment Processing

- Integration with payment gateway (Stripe/PayPal/Telr)
- Payment method management
- Automatic payment status updates
- Payment history tracking
- Refund processing

• Email Notifications

- Order confirmation emails
- o Invoice delivery emails
- o Order status update notifications
- Quotation sent/approved emails
- Low stock alerts

Advanced Search & Filters

- Full-text product search
- Advanced filtering (price range, category, brand, etc.)
- Product sorting options
- Saved search queries
- Search autocomplete

Phase 2: Business Intelligence (Q2 2025)

Enhanced Analytics

- Predictive analytics
- Sales forecasting
- Inventory trend analysis
- Customer behavior insights
- Custom dashboard widgets

Reporting System

- Custom report builder
- Scheduled report generation
- Email report delivery
- Export to Excel/XLSX
- Report templates

Product Recommendations

- Al-powered product suggestions
- o Frequently bought together
- Recently viewed products
- o Personalized recommendations
- Trending products

Phase 3: Multi-currency & Internationalization (Q3 2025)

Multi-currency Support

- Currency selection per organization
- Real-time currency conversion
- Multi-currency pricing display
- Currency exchange rate updates
- Historical exchange rates

• Internationalization (i18n)

- Multi-language support
- Language selection
- Localized date/time formats
- RTL language support
- Translated product descriptions

Phase 4: API & Integrations (Q3-Q4 2025)

API Documentation

- Swagger/OpenAPI documentation
- API versioning
- Interactive API explorer
- Code samples and SDKs
- Rate limiting documentation

Additional Integrations

- Salesforce integration
- QuickBooks integration
- Xero integration
- Shopify integration
- ERP system connectors

Webhook System

- Webhook configuration UI
- Event subscriptions
- Webhook retry mechanism
- Webhook logs and monitoring

Phase 5: Mobile & Advanced Features (Q4 2025)

Mobile Application

- o iOS native app
- Android native app
- Push notifications
- Offline mode support
- o Mobile-specific UI optimizations

Bulk Operations

- o Bulk order creation
- Bulk product import/export
- Bulk quotation generation
- CSV import/export

Workflow Automation

Custom order approval workflows

- Automated quotation expiration
- Inventory reorder points
- Automated invoice generation rules

Phase 6: Advanced Features (2026)

Subscription Management

- Recurring orders
- Subscription plans
- Auto-renewal
- Subscription modifications

Loyalty Program

- Points system
- Rewards management
- Discount tiers
- Referral program

Advanced Inventory

- Multi-location inventory
- Warehouse management
- Batch/lot tracking
- Serial number tracking

Collaboration Features

- Team collaboration on orders
- Comments on orders/quotations
- Activity feed
- Document attachments

Future Considerations

- Advanced AI features (chatbot, natural language queries)
- Voice ordering capabilities

System Architecture Diagram

```
graph TB

subgraph "Client Layer"

WEB[Web Browser

Next.js App]

MOB[Mobile App

Future]

end

subgraph "Application Layer"

subgraph "Next.js Application"
```

```
FE[Frontend
React Components]
            API[API Routes
REST Endpoints]
            AUTH[NextAuth.js
Authentication]
        end
    end
    subgraph "Service Layer"
        AUTH_SVC[Auth Service]
        CART_SVC[Cart Service]
        ORDER_SVC[Order Service]
        QUOTE_SVC[Quotation Service]
        INV_SVC[Invoice Service]
        ANALYTICS_SVC[Analytics Service]
        ORG_SVC[Organization Service]
        BRANCH SVC[Branch Service]
        PROD_SVC[Product Service]
        ZOHO_SVC[Zoho Sync Service]
        EXPORT_SVC[Export Service]
    end
    subgraph "Repository Layer"
        USER_REPO[User Repository]
        ORG_REPO[Organization Repository]
        BRANCH_REPO[Branch Repository]
        PROD_REPO[Product Repository]
        CART_REPO[Cart Repository]
        ORDER_REPO[Order Repository]
        QUOTE REPO[Quotation Repository]
        INV_REPO[Invoice Repository]
    end
    subgraph "Data Layer"
        PG[(PostgreSQL
Primary Database)]
        REDIS[(Redis
Sessions & Cache)]
    end
    subgraph "External Services"
        ZOHO_INV[Zoho Inventory
Product Sync]
        ZOHO_BOOKS[Zoho Books
Orders & Invoices]
        EMAIL[Email Service
Future1
        PAYMENT[Payment Gateway
Future]
    end
    subgraph "Background Services"
        SYNC SVC[Zoho Sync Service
```

```
Scheduled Sync]
    end
    WEB --> FE
    MOB -.-> API
    FE --> API
    API --> AUTH
    API --> AUTH_SVC
    API --> CART_SVC
    API --> ORDER_SVC
    API --> QUOTE_SVC
    API --> INV_SVC
    API --> ANALYTICS_SVC
    API --> ORG_SVC
    API --> BRANCH_SVC
    API --> PROD_SVC
    API --> ZOHO_SVC
    API --> EXPORT_SVC
    AUTH_SVC --> USER_REPO
    CART_SVC --> CART_REPO
    ORDER_SVC --> ORDER_REPO
    QUOTE_SVC --> QUOTE_REPO
    INV_SVC --> INV_REPO
    ORG_SVC --> ORG_REPO
    BRANCH_SVC --> BRANCH_REPO
    PROD_SVC --> PROD_REPO
    USER_REPO --> PG
    ORG REPO --> PG
    BRANCH_REPO --> PG
    PROD_REPO --> PG
    CART_REPO --> PG
    ORDER_REPO --> PG
    QUOTE_REPO --> PG
    INV_REPO --> PG
    AUTH --> REDIS
    AUTH_SVC --> REDIS
    ZOHO_SVC --> ZOHO_INV
    ZOHO_SVC --> ZOHO_BOOKS
    ORDER_SVC --> ZOHO_BOOKS
    INV_SVC --> ZOHO_BOOKS
    BRANCH_SVC --> ZOHO_BOOKS
    SYNC_SVC --> ZOHO_INV
    SYNC_SVC --> PROD_REPO
    SYNC_SVC --> PG
    ORDER_SVC -.-> EMAIL
    INV_SVC -.-> EMAIL
    ORDER SVC -.-> PAYMENT
```

```
style WEB fill:#4A90E2
style MOB fill:#4A90E2,stroke-dasharray: 5 5
style FE fill:#61DAFB
style API fill:#0070F3
style PG fill:#336791
style REDIS fill:#DC382D
style ZOHO_INV fill:#FF6600
style ZOHO_BOOKS fill:#FF6600
style EMAIL fill:#999,stroke-dasharray: 5 5
style PAYMENT fill:#999,stroke-dasharray: 5 5
```

User Journey Flow

```
sequenceDiagram
   participant U as User
   participant WEB as Web App
   participant API as API Routes
   participant SVC as Services
   participant DB as Database
   participant ZOHO as Zoho Services
   Note over U, ZOHO: User Registration & Onboarding
   U->>WEB: Register Account
   WEB->>API: POST /api/auth/register
   API->>SVC: Auth Service
   SVC->>DB: Create User
   DB-->>SVC: User Created
   SVC-->>API: Success
   API-->>WEB: Redirect to Onboarding
   WEB->>API: Create Organization
   API->>SVC: Organization Service
   SVC->>DB: Create Org & Branch
   DB-->>SVC: Created
   SVC-->>API: Success
   Note over U, ZOHO: Zoho Integration Setup
   U->>WEB: Connect Zoho
   WEB->>API: GET /api/zoho/oauth/start
   API->>ZOHO: OAuth Flow
   ZOHO-->>API: OAuth Tokens
   API->>DB: Save Zoho Connection
   U->>WEB: Sync Products
   WEB->>API: POST /api/zoho/sync
   API->>ZOHO: Fetch Products
   ZOHO-->>API: Product Data
   API->>SVC: Product Service
   SVC->>DB: Store Products
   Note over U, ZOHO: Shopping Experience
   U->>WEB: Browse Products
```

```
WEB->>API: GET /api/products
API->>SVC: Product Service
SVC->>DB: Query Products
DB-->>SVC: Products
SVC-->>API: Product List
API-->>WEB: Display Products
U->>WEB: Add to Cart
WEB->>API: POST /api/cart
API->>SVC: Cart Service
SVC->>DB: Update Cart
DB-->>SVC: Cart Updated
U->>WEB: Checkout
WEB->>API: POST /api/orders
API->>SVC: Order Service
SVC->>DB: Create Order
DB-->>SVC: Order Created
SVC->>ZOHO: Create Sales Order
ZOHO-->>SVC: Sales Order ID
SVC->>DB: Update Order
SVC->>ZOHO: Generate Invoice
ZOHO-->>SVC: Invoice Created
SVC->>DB: Store Invoice
DB-->>SVC: Complete
SVC-->>API: Order Success
API-->>WEB: Order Confirmation
```

■ Data Flow Diagram

```
flowchart TD
    START([User Action]) --> TYPE{Action Type}
    TYPE --> | Browse | PROD[Product Catalog]
    TYPE --> | Cart | CART [Cart Management]
    TYPE -->|Order| ORDER[Order Processing]
    TYPE -->|Quote| QUOTE[Quotation Management]
    TYPE --> | Analytics | ANALYTICS [Analytics Dashboard]
    PROD --> PROD_DB[(Product DB)]
    PROD DB -.Sync.-> ZOHO SYNC[Zoho Sync Service]
    ZOHO_SYNC --> ZOHO_INV[(Zoho Inventory)]
    CART --> CART SVC[Cart Service]
    CART_SVC --> CART_DB[(Cart DB)]
    CART --> ORDER
    ORDER --> ORDER_SVC[Order Service]
    ORDER_SVC --> ORDER_DB[(Order DB)]
    ORDER_SVC --> ZOHO_BOOKS_INT[Zoho Books Integration]
    ZOHO_BOOKS_INT --> INVOICE[Invoice Generation]
```

```
INVOICE --> INVOICE_DB[(Invoice DB)]
INVOICE --> ZOHO_BOOKS[(Zoho Books)]
QUOTE --> QUOTE_SVC[Quotation Service]
QUOTE_SVC --> QUOTE_DB[(Quotation DB)]
QUOTE_SVC -.Convert.-> ORDER
ANALYTICS --> ANALYTICS_SVC[Analytics Service]
ANALYTICS_SVC --> ANALYTICS_DB[(Aggregated Data)]
ANALYTICS_SVC --> EXPORT[Export Service]
EXPORT --> PDF[PDF Export]
EXPORT --> CSV[CSV Export]
style START fill:#4A90E2
style ZOHO_INV fill:#FF6600
style ZOHO_BOOKS fill:#FF6600
style PDF fill:#28A745
style CSV fill:#28A745
```

S Contributing

- 1. Fork the repository
- 2. Create a feature branch (git checkout -b feature/amazing-feature)
- 3. Commit your changes (git commit -m 'Add amazing feature')
- 4. Push to the branch (git push origin feature/amazing-feature)
- 5. Open a Pull Request

I License

This project is licensed under the MIT License.



Acknowledgments

- Next.js
- Shadcn UI
- Prisma
- Zoho Inventory

Support

For issues and questions, please open an issue on GitHub.

Built with wusing Next.js & TypeScript