* Project Report
  + Project Plan
    1. Development Process
       1. Gather requirements
          - Track Inventory
          - Track Stock Levels
          - Log changes
       2. Design
          - Deciding on the database schema to be used. A normalized schema along with relationship diagrams helped to avoid redundancy and maintain consistency
       3. Development
          - MySQL for the database construction
  + Technical Details
    1. Logic behind Key Features
       1. Use of Joins for Data Retrieval
  + Database Design
    1. Schema Logic
       1. Tables
          - Products

Columns for storing product and stock data. Columns include "product\_id, product\_name, stock\_quantity, supplier\_id, created\_at".

* + - * + Suppliers

Columns for storing supplier information. Columns include "supplier\_id, supplier\_name, contact\_email, contact\_phone, created\_at".

* + - * + Shipments

Columns for tracking incoming stock and supplier information. Columns include "shipment\_id, supplier\_id, product\_id, quantity, shipment\_date".

* + - * + Orders

Columns for managing customer orders. Columns include "order\_id, customer\_name, order\_date, total\_amount, status".

* + - * + Order\_Details

Columns for linking orders to products. Link is created with a many to many relationship. Columns include "order\_detail\_id, order\_id, product\_id, quantity, price".

* + - * + Users

Columns for managing authentication and user access roles. Columns include "user\_id, username, password\_hash, role, created\_at"

* + - 1. Relationships
         * Products.supplier\_id → suppliers.supplier\_id - Join (foreign key) is established on the supplier\_id table.
         * Order\_details.order\_id → orders.order\_id - Join is made on "order\_id" linking order details to an order.
         * Order\_details.product\_id → products\_product\_id is linked to order details on specific products using the 'product\_id' column.