นี่คือสรุปงานทั้งหมดค่ะ:

**CRAFTHER Project - Overall Progress Summary**

**Overall Progress (Estimated): ~35-40% Complete**

**I. Phase 1: Core MVP Development (Backend & Initial Frontend)**

* **Backend Core & Infrastructure:** (Completed: 100%)
  + User Authentication & Authorization (JWT-based): 100%
  + Organization-based Data Isolation: 100%
  + API Service Registration: 100%
  + Database Migrations: 100%
* **Master Data Management:** (Completed: 100%)
  + Unit Group Management: 100%
  + Unit of Measure Management: 100%
  + Item Category Management: 100%
  + Stock Adjustment Types: 100%
* **Component Management:** (Completed: 100%)
  + API (CRUD): 100%
  + Includes PurchaseUnit, InventoryUnit, PurchaseToInventoryConversionFactor, CurrentStockQuantity: 100%
  + Unique ComponentCode per Organization: 100%
* **Product Management:** (Completed: 100%)
  + API (CRUD): 100%
  + Includes ProductUnit, SaleUnit, SellingPrice, CalculatedCost, IsSubProduct: 100%
  + Unique ProductCode per Organization: 100%
* **Unit Conversion Management:** (Completed: 100%)
  + Service (Direct, Reverse, Base Unit Conversion): 100%
  + Remarks field added: 100%
* **Stock Adjustment Management:** (Completed: 100%)
  + API (CRUD): 100%
  + Real-time CurrentStockQuantity update: 100%
  + Calculates quantity using Unit Conversion: 100%
  + Get Current Stock Balances: 100%
  + StockAdjustment Model & DTOs (Component/Product support): 100%
* **Bill of Materials (BOM) Management:**
  + BOMItem Management (API, Component/SubProduct support): 100%
  + CalculatedCost & What-If Analysis (Multi-level, WastePercentage): 100% (Backend Logic)
  + ComponentPriceHistory: 0% (Backend Logic & Model)
* **Production Order Management (Basic):** (Backend: 100%)
  + Models (ProductionOrder, ProductionOrderItem): 100%
  + DTOs: 100%
  + ApplicationDbContext & Database Migrations: 100%
  + IProductionOrderService & ProductionOrderService (Structure): 100%
  + Implement Logic: Create, Update, Delete, Complete, GetRequiredMaterials: 100%
  + Build ProductionOrdersController & API Endpoints with Authorization: 100%

**II. Remaining Work for Phase 1 MVP (Backend & Frontend)**

* **Basic Reporting & Analytics:**
  + Stock Movement History Report (Backend API & Logic): 0%
  + Low Stock Alert (Backend API & Logic): 50% (Logic in StockAdjustmentService exists, needs API)
  + Cost per Menu/Piece Report (Backend API & Logic): 0%
  + Dashboard / Overview Summary (Backend data preparation): 0%
* **Import Raw Material Prices from Excel/CSV (Backend Logic):** 0%
  + API & Logic for mass import: 0%
* **Technical Enhancements (Ongoing):**
  + Error Handling & Logging: 20% (Needs more detailed implementation)
  + Pagination & Filtering for large data APIs: 10% (Initial design, needs implementation for relevant APIs)
  + Unit Tests for critical Business Logic: 30% (Initial coverage, needs more comprehensive testing)
  + Role-Based Access Control (RBAC) refinement: 50% (Basic setup, needs detailed policy/role mapping)
* **User Onboarding & Initial Setup (Backend Implications):** 20% (Backend APIs for initial data/templates are placeholders)
* **Frontend Integration & UI/UX Development:** 0% (Overall Frontend work)
  + Build UI/UX for all developed APIs: 0%
  + Connect with Gamification (Points, Badges, Leaderboard, Challenges): 0%

**III. Future Phases (Roadmap - Not started in current scope)**

* **Production Order Management (Remaining):**
  + Actual Waste Recording: 0%
  + Actual UnitPrice at Production: 0%
* **Perishable Goods Management (Batch/Lot, FEFO, Expiry Alerts):** 0%
* **Supplier Management:** 0%
* **Purchase Order (PO) Management:** 0%
* **Reporting & Analytics (Advanced):**
  + Actual Production Cost Report: 0%
  + Advanced Dashboard/KPIs: 0%
* **Mobile-First Design (Frontend):** 0%
* **Gamification (Full Implementation):** 0%
* **Integrations (External Systems):**
  + OCR Receipt Integration: 0%
  + POS & E-commerce Integration (Shopify, Wongnai POS): 0%
* **Demand Forecasting (AI/ML):** 0%
* **In-App Currency & Marketplace:** 0%

**Overall Backend Progress (Calculated):** (All completed features for Backend) + (Partial for remaining Phase 1 Backend work) = 100% (Backend Core) + 100% (Master Data) + 100% (Component) + 100% (Product) + 100% (Unit Conv) + 100% (Stock Adj) + (100% BOMItem + 100% CalculatedCost + 0% PriceHistory) + 100% (Production Order) + (0% Report + 50% Low Stock + 0% Cost Report + 0% Dashboard) + 0% (Import Excel) + (20% Error Handling + 10% Paging + 30% Unit Test + 50% RBAC) + 20% (User Onboarding)

ถ้าเฉลี่ยจากน้ำหนักงานและสถานะที่ทำไปแล้วของ Backend ทั้งหมดใน Phase 1 และส่วน Core ที่ทำเสร็จแล้ว คิดเป็นประมาณ **75-80%** ของ Backend ทั้งหมดที่วางแผนไว้ใน Phase 1 นะคะ

**Overall Project Progress (Combined Backend & Frontend):** ถ้าให้ Backend และ Frontend มีสัดส่วนความสำคัญเท่ากัน (50% Backend, 50% Frontend)

* **Backend Progress:** ~75-80%
* **Frontend Progress:** 0% (ยังไม่ได้เริ่มจริงจังตามที่ระบุ)

ดังนั้น **ความคืบหน้าโดยรวมของโปรเจกต์ (Overall Project Progress)** อยู่ที่ประมาณ **35-40%** ค่ะ (75% \* 0.5 + 0% \* 0.5 = 37.5%)

**Entity Relationship**

**คำอธิบายความสัมพันธ์ (Cardinality):**

* **Organization (1) <---> (Many) ApplicationUser**: หนึ่งองค์กรมีผู้ใช้ได้หลายคน
* **Organization (1) <---> (Many) Component**: หนึ่งองค์กรมีวัตถุดิบได้หลายรายการ
* **Organization (1) <---> (Many) Product**: หนึ่งองค์กรมีสินค้าได้หลายรายการ
* **Organization (1) <---> (Many) UnitConversion**: หนึ่งองค์กรมีกฎการแปลงหน่วยได้หลายรายการ
* **Organization (1) <---> (Many) StockAdjustment**: หนึ่งองค์กรมีการปรับปรุงสต็อกได้หลายรายการ
* **Organization (1) <---> (1) SubscriptionPlan**: หนึ่งองค์กรมีแผนการสมัครสมาชิกหนึ่งแผน
* **Organization (1) <---> (1) OrganizationIndustryType**: หนึ่งองค์กรมีประเภทอุตสาหกรรมหนึ่งประเภท
* **UnitGroup (1) <---> (Many) UnitOfMeasure**: หนึ่งกลุ่มหน่วยมีหน่วยวัดได้หลายหน่วย
* **UnitOfMeasure (Many) <---> (1) Component (PurchaseUnit)**: วัตถุดิบมีหน่วยซื้อหนึ่งหน่วย
* **UnitOfMeasure (Many) <---> (1) Component (InventoryUnit)**: วัตถุดิบมีหน่วยสต็อกหนึ่งหน่วย
* **UnitOfMeasure (Many) <---> (1) Product (ProductUnit)**: สินค้ามีหน่วยผลิตภัณฑ์หนึ่งหน่วย
* **UnitOfMeasure (Many) <---> (1) Product (SaleUnit)**: สินค้ามีหน่วยขายหนึ่งหน่วย
* **UnitOfMeasure (Many) <---> (1) BOMItem (UsageUnit)**: รายการในสูตรมีหน่วยใช้งานหนึ่งหน่วย
* **UnitOfMeasure (Many) <---> (1) UnitConversion (FromUnit/ToUnit)**: การแปลงหน่วยเกี่ยวข้องกับหน่วยต้นทางและปลายทาง
* **UnitOfMeasure (Many) <---> (1) StockAdjustment (UnitOfMeasureId)**: การปรับปรุงสต็อกมีหน่วยวัดหนึ่งหน่วย
* **ItemCategory (1) <---> (Many) Component**: หนึ่งหมวดหมู่มีวัตถุดิบได้หลายรายการ
* **ItemCategory (1) <---> (Many) Product**: หนึ่งหมวดหมู่มีสินค้าได้หลายรายการ
* **Product (1:Parent) <---> (Many) BOMItem**: หนึ่งสินค้า (Parent Product) มีส่วนประกอบในสูตร (BOMItem) ได้หลายรายการ
* **Component (1) <---> (Optional) BOMItem (ComponentId)**: หนึ่งวัตถุดิบสามารถเป็นส่วนประกอบใน BOMItem ได้ (BOMItem.ComponentId เป็น Nullable)
* **Product (1:SubProduct) <---> (Optional) BOMItem (ProductId)**: หนึ่งสินค้า (SubProduct) สามารถเป็นส่วนประกอบย่อยใน BOMItem ได้ (BOMItem.ProductId เป็น Nullable)
* **StockAdjustmentType (1) <---> (Many) StockAdjustment**: หนึ่งประเภทการปรับปรุงมีรายการปรับปรุงสต็อกได้หลายรายการ
* **QuestType (1) <---> (Many) Quest**: หนึ่งประเภทภารกิจมีภารกิจได้หลายรายการ
* **Quest (1) <---> (Many) UserQuest**: หนึ่งภารกิจมีบันทึกภารกิจของผู้ใช้ได้หลายรายการ
* **ApplicationUser (1) <---> (Many) UserQuest**: หนึ่งผู้ใช้มีบันทึกภารกิจได้หลายรายการ
* **Level (1) <---> (Optional) UserScore**: หนึ่ง Level สามารถเกี่ยวข้องกับ UserScore ได้
* **ApplicationUser (1) <---> (1) UserScore**: หนึ่งผู้ใช้มีคะแนนสะสมหนึ่งรายการ

**หมายเหตุ:**

* **Optional (Nullable):** ความสัมพันธ์ที่มีเครื่องหมายดอกจัน (\*) หรือคำว่า "(Nullable)" หมายความว่าฝั่ง Many หรือฝั่ง Optional อาจไม่มีค่าเชื่อมโยงเสมอไป (เช่น ComponentId หรือ ProductId ใน BOMItem จะมีค่าเพียงตัวใดตัวหนึ่งตาม ComponentType)
* **Self-referencing:** Product สามารถเป็น ParentProduct และเป็น SubProduct (ถูกใช้ใน BOMItem ของ Product อื่น) ได้