**Finalized Essential Components for Wellnex Component Diagram (with Databases)**

**1. Presentation Layer (Flutter App UI)**

* **User Interface Component**
  + ProfileScreen
  + HealthInputScreen
  + WorkoutPlanScreen
  + DietPlanScreen
  + ProgressTrackerScreen
  + TipsAndEducationScreen

**2. Business Logic Layer**

* **User Profile Manager**
  + Manages user data (age, goal, weight, activity level, etc.)
  + **Uses:** <<database>> UserDatabase
* **Workout Plan Generator**
  + Builds personalized workout routines based on:
    - Goal, age, weight, activity level
  + **Uses:** <<database>> WorkoutPlanStorage
* **Diet Plan Generator**
  + Suggests diet plans based on:
    - Calories burned, health goals, dietary preferences
  + **Uses:** <<database>> DietPlanStorage
* **Recommendation Engine**
  + Coordinates plan generation using the Factory Method Pattern
  + Integrates output from the workout and diet generators
* **Progress Tracker**
  + Tracks user performance and health metrics over time
  + **Uses:** <<database>> ProgressData
* **Health Input Parser**
  + Validates and normalizes input from users or wearable APIs
* **Notification Manager**
  + Manages and schedules reminders (exercise, hydration, meals)
* **Content Delivery System**
  + Fetches wellness tips and educational material
  + **Uses:** <<database>> TipContentCache

**3. Data Layer (Databases)**

* <<database>> UserDatabase
* <<database>> WorkoutPlanStorage
* <<database>> DietPlanStorage
* <<database>> ProgressData
* <<database>> TipContentCache

**4. Integration / External Systems**

* **API Connector**
  + Integrates with:
    - Wearable device APIs (for calorie burn, step count, etc.)
    - Nutrition APIs (for calorie tracking or food lookup)
* **Authentication Service**
  + Handles user login and registration (e.g., Firebase Auth)
* **Analytics System** *(optional)*
  + Tracks user behavior and feature usage for feedback or improvement

# Relationship

**Presentation Layer**

* **User Interface Component**:
  + **Depends on**:
    - Recommendation Engine
    - Content Delivery System
    - User Profile Manager
    - Progress Tracker

Because the UI would interact with these components to display content, recommendations, track progress, and allow profile updates.

**Component Layer**

* **Recommendation Engine**:
  + **Uses**:
    - User Profile Manager
    - WorkoutPlan DB, DietPlan DB, TipContent DB (from Data Layer)
* **Content Delivery System**:
  + **Uses**:
    - TipContent DB, WorkoutPlan DB (for delivering content)
* **User Profile Manager**:
  + **Uses**:
    - User DB (to fetch and update user information)
* **Diet Plan Generator**:
  + **Uses**:
    - User Profile Manager
    - DietPlan DB
* **Progress Tracker**:
  + **Uses**:
    - User DB (to store progress)
    - May also interact with WorkoutPlan DB