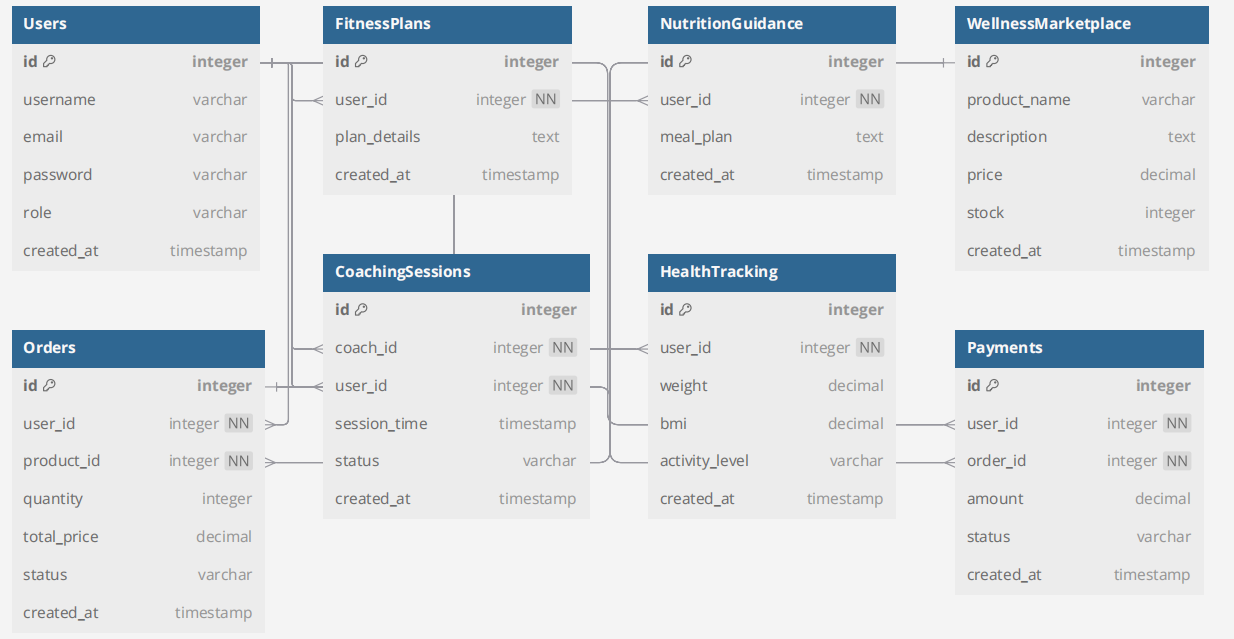
ER DIAGRAM:-

The collections based on this ER diagram (assuming a NoSQL database like MongoDB) would be:

1. **Users**
   * \_id: ObjectId
   * username: string
   * email: string
   * password: string
   * role: string
   * created\_at: timestamp
2. **FitnessPlans**
   * \_id: ObjectId
   * user\_id: ObjectId (Reference to Users)
   * plan\_details: text
   * created\_at: timestamp
3. **NutritionGuidance**
   * \_id: ObjectId
   * user\_id: ObjectId (Reference to Users)
   * meal\_plan: text
   * created\_at: timestamp
4. **CoachingSessions**
   * \_id: ObjectId
   * coach\_id: ObjectId (Reference to Users)
   * user\_id: ObjectId (Reference to Users)
   * session\_time: timestamp
   * status: string
   * created\_at: timestamp
5. **HealthTracking**
   * \_id: ObjectId
   * user\_id: ObjectId (Reference to Users)
   * weight: decimal
   * bmi: decimal
   * activity\_level: string
   * created\_at: timestamp
6. **WellnessMarketplace**
   * \_id: ObjectId
   * product\_name: string
   * description: text
   * price: decimal
   * stock: integer
   * created\_at: timestamp
7. **Orders**
   * \_id: ObjectId
   * user\_id: ObjectId (Reference to Users)
   * product\_id: ObjectId (Reference to WellnessMarketplace)
   * quantity: integer
   * total\_price: decimal
   * status: string
   * created\_at: timestamp
8. **Payments**
   * \_id: ObjectId
   * user\_id: ObjectId (Reference to Users)
   * order\_id: ObjectId (Reference to Orders)
   * amount: decimal
   * status: string
   * created\_at: timestamp

Each collection corresponds to a table in the relational model, with references (user\_id, product\_id, etc.) acting like foreign key