

Phase 3: Data Modeling & Relationships

1. Standard & Custom Objects

- **Standard Objects (already in Salesforce):**
 - **User** → represents each app user (patient).
 - **Admin** → system admin role to oversee data.
 - **Report/Dashboard** → for analytics.

- **Custom Objects (create in Object Manager):**

1. **Health Log** → main table for user entries.
 - Tracks daily health metrics (date, symptoms, type).
2. **Fitness Activity** → stores exercise details.
3. **Diet Log** → stores meals and calories.
4. **Medical Record** → for doctor visits, medication, conditions.

Setup:

- **Setup → Object Manager → Create → Custom Object**
 - Fill in Label, Plural Label, Tab, and enable Reports/Activities.
-

2. Fields

Add fields to each custom object.

- **Health Log:**
 - Date (Date)
 - Log_Type (Picklist: Fitness, Diet, Medical)
 - Notes (Long Text)
 - Critical_Flag (Checkbox)
- **Fitness Activity:**
 - Activity_Type (Picklist: Running, Yoga, Gym)
 - Duration (Number)
 - Calories_Burned (Number)

Fitness Activity

Fields & Relationships
9 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Activity Type	Activity_Type__c	Picklist		
Calories Burned	Calories_Burned__c	Number(18, 0)		
Created By	CreatedBy	Lookup(User)		
Currency	CurrencyIsoCode	Picklist		
Duration	Duration__c	Number(18, 0)		
Fitness Activity Name	Name	Text(80)		✓
Health Log	Health_Log__c	Lookup(Health Log)		✓
Last Modified By	LastModifiedBy	Lookup(User)		
Owner	OwnerId	Lookup(User,Group)		✓

- **Diet Log:**

- Meal_Type (Picklist: Breakfast, Lunch, Dinner)
- Calories (Number)
- Notes (Text)

Health Log

Fields & Relationships
10 Items, Sorted by Field Label

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedBy	Lookup(User)		
Critical Falg	Critical_Falg__c	Checkbox		
Currency	CurrencyIsoCode	Picklist		
Date	Date__c	Date		
Diet Log	Diet_Log__c	Master-Detail(Diet Log)		✓
Health Log Name	Name	Text(80)		✓
Last Modified By	LastModifiedBy	Lookup(User)		
Log Type	Log_Type__c	Picklist		
Notes	Notes__c	Long Text Area(32768)		

- **Medical Record:**

- Doctor_Name (Text)
- Visit_Date (Date)
- Prescription (Long Text)

👉 Setup:

- Object Manager → [Object] → Fields & Relationships → New Field.

Medical Record

Fields & Relationships

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Currency	CurrencyIsoCode	Picklist		
Doctor Name	Doctor_Name__c	Text(50)		
Health Log	Health_Log__c	Lookup(Health Log)		
Last Modified By	LastModifiedById	Lookup(User)		
Medical Record Name	Name	Text(80)		
Owner	OwnerId	Lookup(User,Group)		
Prescription	Prescription__c	Long Text Area(32768)		
Visit Date	Visit_Date__c	Date		

3. Record Types

Use Record Types to separate **categories of Health Log**.

- **Health Log Record Types:**
 - Fitness Log
 - Diet Log
 - Medical Log

👉 Setup:

- Object Manager → Health Log → Record Types → New Record Type.
- Assign to profiles (User, Admin).

Health Log

Record Types

RECORD TYPE LABEL	DESCRIPTION	ACTIVE	MODIFIED BY
Diet Log		✓	Sapna Singh, 9/22/2025, 7:16 PM
Fitness Log		✓	Sapna Singh, 9/22/2025, 7:15 PM
Medical Log		✓	Sapna Singh, 9/22/2025, 7:16 PM

4. Page Layouts

Customize layouts for each Record Type.

- **Fitness Log Layout** → Show fields: Activity_Type, Duration, Calories_Burned.
- **Diet Log Layout** → Show fields: Meal_Type, Calories, Notes.
- **Medical Log Layout** → Show fields: Doctor_Name, Prescription, Visit_Date.

👉 Setup:

- **Object Manager** → **Health Log** → **Page Layouts** → **Edit/Create Layout**.
 - Assign layouts per Record Type.
-

5. Compact Layouts

Control what fields show in highlights (top panel, mobile).

- **Health Log Compact Layout:**

- Date
- Log_Type
- Critical_Flag
- Status

👉 Setup:

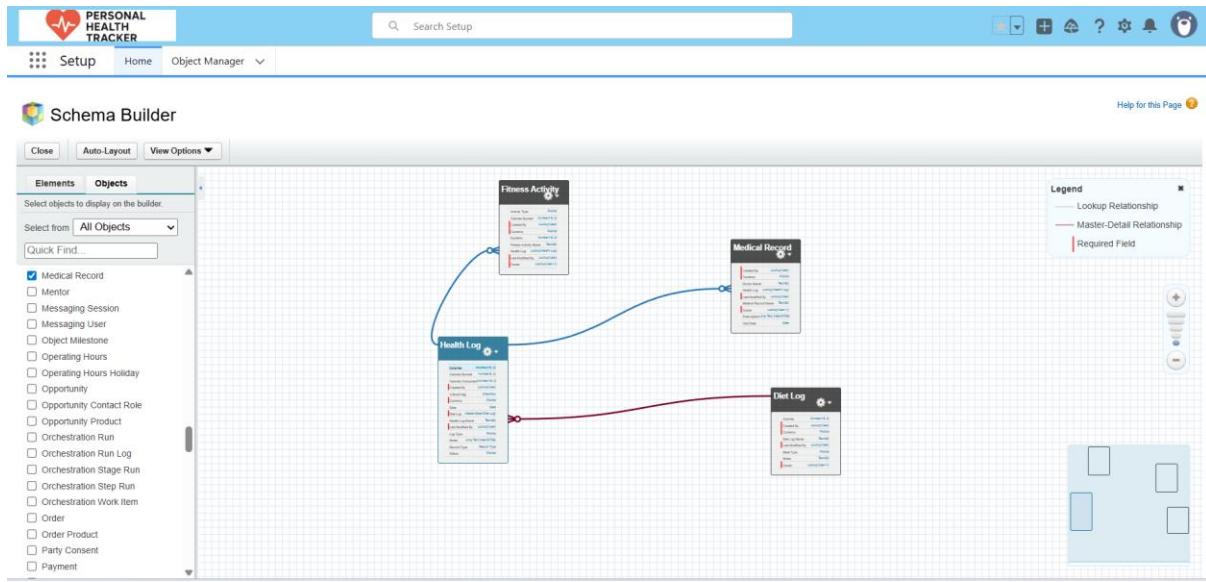
- **Object Manager** → **Health Log** → **Compact Layouts** → **New**.
 - Assign via **Compact Layout Assignment**.
-

6. Schema Builder

Use Schema Builder to **visualize relationships**.

👉 Setup:

1. Go to **App Launcher** → **Schema Builder**.
2. Drag **User, Health Log, Fitness Activity, Diet Log, Medical Record**.
3. Draw relationships (see below).



7. Lookup vs Master-Detail vs Hierarchical

- **Health Log → User = Master-Detail** (Health Logs belong to a User).
- **Fitness Activity → Health Log = Lookup** (optional details).
- **Diet Log → Health Log = Lookup.**
- **Medical Record → Health Log = Lookup.**
- **Hierarchical = available only on User object** (not needed here unless you want Manager → User reporting).

👉 Setup:

- **Object Manager → Fields & Relationships → New → Relationship Field.**
- Choose **Lookup or Master-Detail.**

8. Junction Objects

Needed for **many-to-many relationships**.

Example:

- **A Medical Record** can relate to **multiple Health Logs**.
- **A Health Log** can also be related to multiple Medical Records.

👉 Solution:

- Create a junction object: **Health Log – Medical Record**.
- Add two Master-Detail fields:
 - To **Health Log**

- To Medical Record

FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Created By	CreatedById	Lookup(User)		
Currency	CurrencyIsoCode	Picklist		
Health Log	Health_Log__c	Master-Detail(Health Log)		
Health Log - Medical Record Name	Name	Text(80)		
Last Modified By	LastModifiedById	Lookup(User)		
Medical Record	Medical_Record__c	Master-Detail(Medical Record)		

9. External Objects

If your app pulls external health data (e.g., Fitbit, hospital API):



1. Go to **Setup → External Data Sources**.
2. Define a data source (OData, API, etc.).
3. Create **External Objects** (e.g., Fitbit_Activity_x).
4. Relate them to Health Log (Lookup).