**M3V1: Contributions by Shivani Bokka (Database Administrator)**

**3. Data Definitions – Version 2 (based on the ERD diagram created by Hamed Emari)**

**1. User**

Users are the people who interact with the app. Each person has a profile that includes their username, first and last name, email, gender, and a securely encrypted password. The system also tracks the date the user joined (created\_at). Users can create custom habits and participate in wellness circles.

**2. Habit**

This is an abstract table that serves as the general structure for all habits. It does not store any attributes itself but acts as a parent to both Custom\_Habit and Defined\_Habit. Its purpose is to provide a unified structure for different types of habits that can be tracked in the system.

**3. Custom\_Habit**

Custom habits are routines defined by users themselves. Each habit includes a descriptive name, the value recorded (like “2L” for water or “30 mins” for a workout), and the timestamp of when it was recorded. Every custom habit is tied to the user\_id of the person who created it. This model directly inherits from the abstract Habit class.

**4. Defined\_Habit**

Defined habits are preconfigured routines that come built into the app, such as “Drink Water” or “Morning Walk.” These habits inherit from the abstract Habit entity and do not contain any additional attributes. They offer users a quick-start option and promote uniformity across wellness circles.

**5. Wellnest\_Circle**

A Wellnest Circle is a group created to support shared health goals. Each group can have only one goal assigned to it. Users can join multiple wellness circles—either by creating one themselves or by joining a group started by another user.

**6. Joins**

This is an associative entity that links users to wellness circles. Since users can belong to many groups and groups can have many users, this table handles the many-to-many relationship cleanly.

**7. Leaderboard**

The leaderboard keeps track of each user’s score within a specific wellness circle. It records the total score for each user, the group it belongs to, and the last time the leaderboard was updated. This brings in a sense of progress and healthy competition among users.

**8. Streak**

Streaks monitor how consistently users stick to a habit. For each user-habit pair, we record the current streak length, the longest streak they've ever had, and when the streak was last updated. It rewards users for forming long-lasting routines.

**9. WaterIntake**

This table captures daily water consumption. For every log, it stores how much water was consumed (in milliliters), the user’s ID, and the timestamp of the entry.

**10. FoodIntake**

This tracks what the user eats and when. It records the calorie count, timestamp, and type of meal—whether it’s breakfast, lunch, or dinner. Each entry also ties back to the user who logged it.

**11. SleepLog**

Sleep logs help track rest. Each entry records the user’s ID, when they fell asleep, when they woke up, and the total number of hours slept. This data helps users monitor sleep patterns over time.

**12. WorkoutLog**

Workout logs track physical activity. For each session, it stores the user’s ID, how long they exercised, and when it happened. This supports habit tracking around fitness and physical wellbeing.

**------------------------------------------------------------------------------------------------------------------**

**Variable types to create the UML diagram (helps the backend lead Hamed create the UML diagram to ensure consistency as per the instructions by the professor) – NO INCLUSION IN DOCUMENTATION**

**User**

* id: int (Primary Key, Auto-increment)
* username: varchar (50) (Unique, Not Null)
* first\_name: varchar (50) (Not Null)
* last\_name: varchar (50) (Not Null)
* email: varchar (100) (Unique, Not Null)
* encrypted\_password: varchar (255) (Not Null)
* gender: enum ('Male', 'Female', ‘Non-Binary’, 'Other')
* created\_at: timestamp (Default, Not Null)

**Custom\_Habit**

* id: int (Primary Key, Auto-increment)
* user\_id: int (Foreign Key, Not Null)
* name: varchar(100) (Not Null)
* recorded\_value: varchar(100) (e.g. "2L", "30 mins")
* timestamp: timestamp (Default, Not Null)

**Defined\_Habit**

* The application does not use a central Defined\_Habit table. Instead, all structured habits like sleep, food, water, and workouts are logged directly in their own dedicated tables.

**Wellnest\_Circle**

* id: int (Primary Key, Auto-increment)
* name: varchar (100) (Not Null)
* goal\_description: text
* goal\_type: enum ('Time-Based', 'Repetition-Based', 'Custom', 'Quantity', 'Duration')
* goal\_value: float

**Joins (Associative entity)**

* user\_id: int (Foreign Key, Not Null)
* group\_id: int (Foreign Key → Wellnest\_Circle.id, Not Null)

**Streak**

* id: int (Primary Key, Auto-increment)
* user\_id: int (Foreign Key, Not Null)
* habit\_id: int (Foreign Key, Not Null)
* current\_streak: int
* longest\_streak: int
* last\_updated: timestamp (Default, Not Null)

**Leaderboard**

* id: int (Primary Key, Auto-increment)
* group\_id: int (Foreign Key → Wellness\_Circle.id, Not Null)
* user\_id: int (Foreign Key, Not Null)
* score: int
* last\_updated: timestamp (Default, Not Null)

**WaterIntake**

* id: int (Primary Key, Auto-increment)
* user\_id: int (Foreign Key, Not Null)
* amount\_ml: float (Not Null)
* timestamp: timestamp (Default, Not Null)

**FoodIntake**

* id: int (Primary Key, Auto-increment)
* user\_id: int (Foreign Key, Not Null)
* calories: int
* type\_of\_meal: enum ('Breakfast', 'Lunch', 'Dinner') (Not Null)
* timestamp: timestamp (Default, Not Null)

**WorkoutLog**

* id: int (Primary Key, Auto-increment)
* user\_id: int (Foreign Key, Not Null)
* duration: float (in minutes)
* timestamp: timestamp (Default, Not Null)

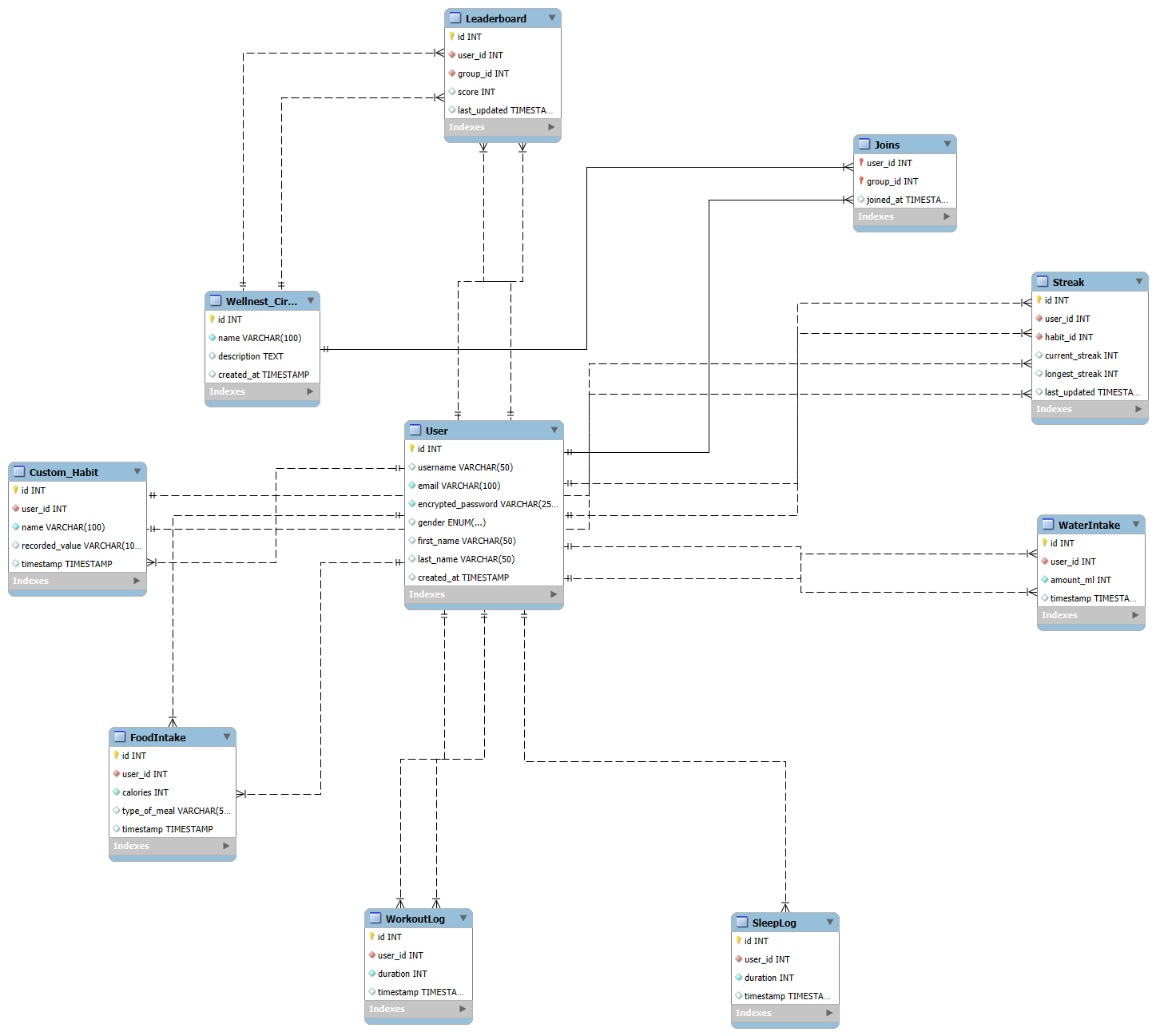
**SleepLog**

* id: int (Primary Key, Auto-increment)
* user\_id: int (Foreign Key, Not Null)
* duration: float (4,2) (in hours)
* timestamp: timestamp (Default, Not Null)

**------------------------------------------------------------------------------------------------------------------**

**6.1 High-Level Database Architecture**

(Include ERD Diagram here)

**Enhanced Entity-Relationship (EER) Diagram:**

**M3V1 – Checkpoint 3 – Database Organization**

**Database Organization**

The database structure for our WellNest app has been thoughtfully designed to align with the app’s current functionality, while keeping future improvements in mind. We’re using MySQL as our relational database, and Django for the backend. Instead of relying on Django’s built-in user model, we chose to define custom tables to maintain full control over the schema and adapt it to our specific needs.

At this stage, our schema consists of 10 main tables: User, Custom\_Habit, WaterIntake, FoodIntake, WorkoutLog, SleepLog, Wellnest\_Circle, Joins, Streak, and Leaderboard. While our conceptual model includes abstract or inherited entities like Habit and Defined\_Habit, these do not translate into physical tables in the database.

The schema is well-normalized, which helps reduce redundancy and ensures consistent, reliable data. At the center of the design is the User table, which connects directly to various personal health logs such as WaterIntake, FoodIntake, WorkoutLog, and SleepLog. Each of these includes a foreign key to the User table and uses timestamps for accurate chronological tracking. Foreign keys are used consistently throughout the schema to maintain referential integrity and enforce proper relationships between tables.

To support user-driven habit formation, the Custom\_Habit table allows users to define and track their own routines. Consistency over time is tracked via the Streak table, which contributes to user rankings in the Leaderboard. Social engagement is encouraged through wellness groups, which are implemented using the Wellnest\_Circle and Joins tables to support many-to-many participation.

As shown in the ER and EER diagrams, each entity has a clearly defined role, and the inheritance logic is handled cleanly through the separation of Custom\_Habit and Defined\_Habit.

Given the current scope of the project, we've prioritized building a solid foundation with the flexibility to scale in the future. While the current setup is intentionally kept simple for a small user base, it’s been built with future enhancements in mind—such as Docker-based containerization and broader deployment.

In terms of practical use, the schema allows for efficient queries such as:

* Retrieving a user's daily water, food, workout, or sleep logs
* Listing all wellness circles a user is part of
* Identifying users with the longest streaks
* Fetching top scorers in each wellness circle
* Aggregating health metrics for leaderboard calculations

Overall, the database is cleanly organized, logically structured, and well-prepared to support both the app’s current goals and potential future developments.

**------------------------------------------------------------------------------------------------------------------**