Student Wellness Information System Technical Documentation

Project Seminar I

Team Members:
Gjore Janevski
Pavel Jolakoski

Mentors:
Klen Čopič Pucihar
Matjaz Kljun

Definition of a Problem

Many students, particularly in the Balkans and Slovenia, struggle to maintain proper nutrition and hydration. This is especially apparent during demanding academic periods, leading to widespread health issues that affect both their physical well-being and academic performance.

A key challenge is the lack of support for traditional Balkan and Slovenian cuisines in existing calorie-tracking apps and fitness platforms. Students frequently consume local meals rich in carbs and calories without knowing their nutritional content. Even when they eat until full and no longer feel hungry, their bodies often remain unsatisfied due to insufficient essential nutrients. During intense study sessions and exam periods, this problem intensifies. Approximately 90% of students opt for the quickest available food options, further reducing their intake of balanced nutrient-rich meals.

Hydration is another critical issue. Students often forget the importance of drinking plain water, especially when busy or focused on studying. Instead, they turn to coffee, energy drinks, and sugary beverages, which provide temporary boosts but fail to meet the body's actual fluid needs. This lack of hydration commonly results in grogginess, headaches, and decreased concentration.

Overall, the combination of poor nutritional awareness, limited tracking tools for regional foods, and neglected hydration creates a cycle of unhealthy habits that leaves students physically drained and less effective in their studies.

Problem Solution

Our proposed solution is the Student Wellness Information System, a platform designed for students in the Balkans and Slovenia. It includes a database of traditional Balkan and West European dishes with full macro-nutrient data. A system that calculates personalized daily nutrition and water targets based on BMI, weight, height, and pals, and tracks intake with smart reminders. This enables students to easily monitor macros and hydration, build healthier habits, and improve their energy and academic performance.

Functional Requirements

1. **User Account Management**: Users can register an account using their name, surname, email, password, weight, height, weekly activity level, and weight goal (cut, maintain, or bulk). Registered users can log in securely and update any of their account details at any time.
2. **Personalized Calculations**: The system automatically calculates the user's BMI based on height, weight, and activity level. It also computes daily macronutrient needs (proteins, fats, carbohydrates, and calories) and daily water intake requirements based on the user's weight, height, activity level, and weight goal.
3. **Food and Nutrition Database**: The system provides a comprehensive database of traditional Balkan and Western European dishes with accurate macronutrient values. Users can search and log these meals, and can also add new dishes by entering the dish name and its nutritional values. Newly added dishes appear alongside existing ones in the database.
4. **Daily Tracking**: Users can log food intake at any time, with the system automatically updating their daily macronutrient consumption. Users can also log water intake throughout the day at any moment. Users should be able to edit or delete any of the logged in values at any time.
5. **Progress Monitoring**: The system displays real-time progress showing how much of daily macronutrient goals and water intake has been achieved, including how far users are under or over their targets.
6. **Notifications**: The system sends a notification every three hours if no water has been logged during that period.

Non-Functional Requirements

1. **Performance**: The system must be capable of processing up to 1000 logs per minute.
2. **Accesibility**: The system will be completely free to use, allowing anyone to create an account. It will be accessible from any location via a web browser on a personal computer connected to the internet.
3. **Availability**: The system will operate 24/7 with a minimum uptime of 90%.
4. Development and Deployment: The system will be developed and deployed within the second semester of 2026.
5. **Documentation**: The system will include complete documentation for installation, usage, and maintanace.
6. **Data Privacy**: The system will store only the minimum necessary user data, and it will follow the best practices to protect user data.
7. **Security**: All data exchange between client and server will be encrypted. User passwords will be securely hashed.
8. **Scalability**: The system will be designed to support future user growth without significant performance degredation.

Feasibility Study

The development of the Student Wellness Information System is highly feasible from economic, technical, and operational perspectives.

Technical Feasibility

The System is user-driven and will require regular updates to the food database. From a technical standpoint, the most complex components are BMI calculation and personalized daily macronutrient and water intake recommendations. These are well-documented and can be implemented using established algorithms available online. The system will use a stable database capable of handling frequent user logs and data updates. All other features (user registration, food/water logging, progress tracking, and notifications) involve standard data writing and reading operations and present no significant technical challenges.

Economic Feasibility

Development and maintenance costs are expected to be low. Hosting can be done on the faculty server at no cost to students, or on a cloud platform if needed. Additional expenses may include marketing to attract users and potential server costs based on traffic volume. Overall, the project is economically viable with minimal financial investment required.

Operational Feasibility

The system will be fully web-based and accessible from any internet-connected device, supporting 24/7 availability. It will store only the essential data (name, surname, email, password, weight, height, weekly activity level and weight goal) while following best practices for data privacy and security.

Given this combination of low cost, proven technical solutions, and straightforward operations makes the Student Wellness Information System a practical and achievable project.

Matrix of User Roles

Feature	Unregistered Users	Registered Users
Register an account	Yes	/
Log in to an account	No	Yes
Browse / Search Food Database	No	Yes
Update Profile (weight, height, activity, goal, etc)	No	Yes
Calculate BMI & Personalized Goals (Macros + Water)	No	Yes
Log Food Intake	No	Yes
Log Water Intake	No	Yes
View Daily Progress (Macros & Water)	No	Yes
Add New Dishes to Database	No	Yes
Receive Notifications (Water reminders)	No	Yes
Access Historical Data	No	Yes

Data Dictionary

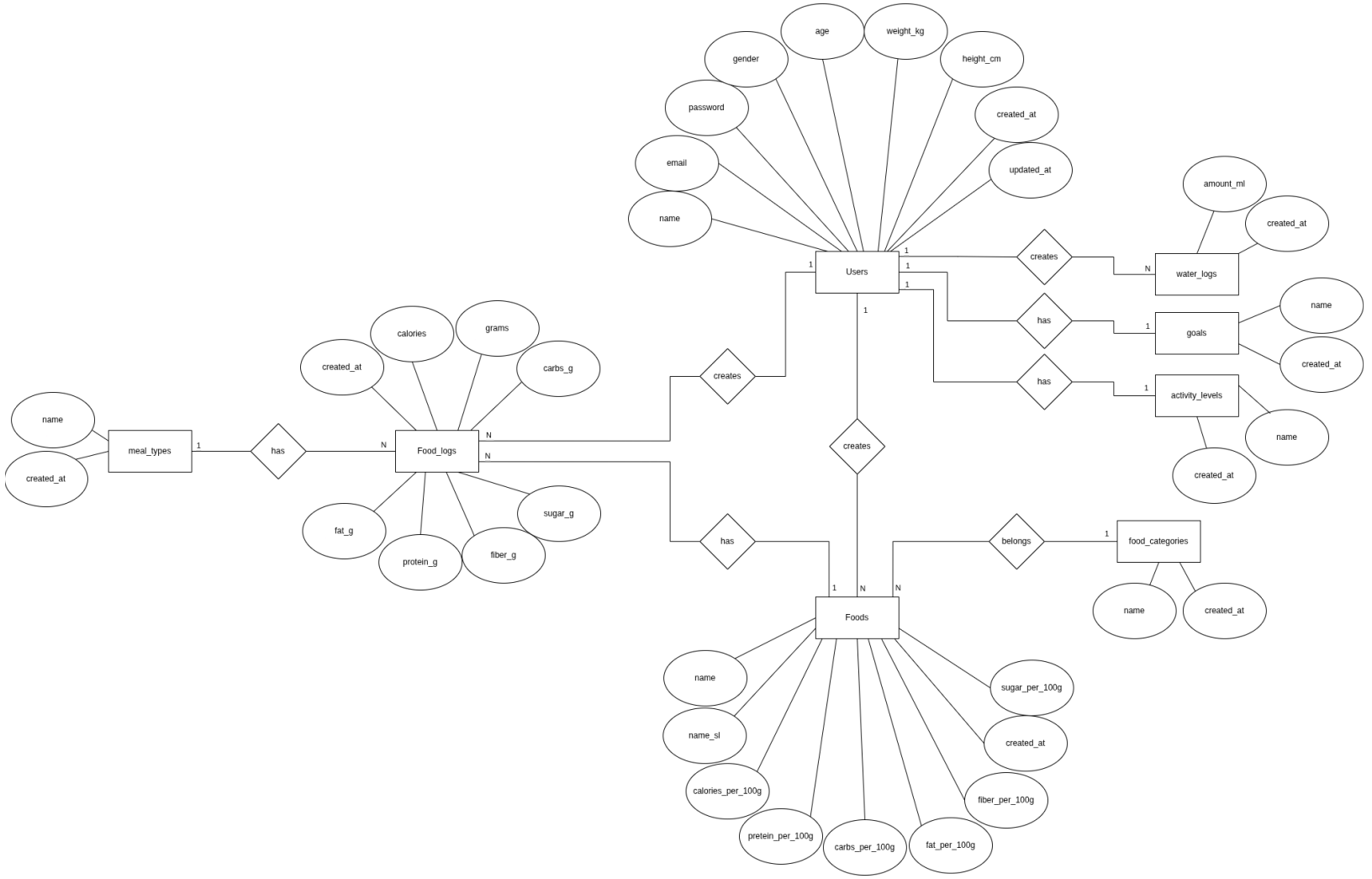
Entity	Description	Attribute	Type	Description
Users	User of the system.	id	int	Unique identifier of the user.
		activity_level_id	int	The unique identifier of the activity level.
		goal_id	int	The unique identifier of the goal.
		email	varchar(255)	The email of the user.
		password	varchar(255)	The password of the user, hashed.
		name	varchar(255)	The name of the user.
		gender	varchar(20)	The gender of the user.
		age	int	The age of the user.
		weight_kg	decimal(5,2)	The weight of the user.
		height_cm	decimal(5,2)	The height of the user.
		created_at	timestamp	The time when the user's account was created.
		updated_at	timesamp	The time when the user's account was updated.
Foods	Foods and their macros.	id	int	Unique identifier of the Food.
		food_category_id	int	The unique identifier of the food's category.

		name	varchar(255)	The name of the food.
		name_sl	varchar(255)	The name of the food in Slovene.
		calories_per_100g	decimal(7,2)	The calories of the food per 100g.
		protein_per_100g	decimal(6,2)	The protein of the food per 100g.
		carbs_per_100g	decimal(6,2)	The carbs of the food per 100g.
		fat_per_100g	decimal(6,2)	The fat of the food per 100g.
		fiber_per_100g	decimal(6,2)	The fiber of the food per 100g.
		sugar_per_100g	decimal(6,2)	The sugar of the food per 100g.
		created_at	timestamp	The time when the food was created.
Food_logs	The saved food logs.	id	int	The unique identifier of the food logs.
		user_id	int	The unique identifier of the user creating the food log.
		food_id	int	The unique identifier of the food that was logged.
		meal_type_id	int	The unique identifier of the type of meal.
		grams	decimal(7,2)	The amount of grams of the food logged.
		calories	decimal(7,2)	The amount of calories of the food logged.

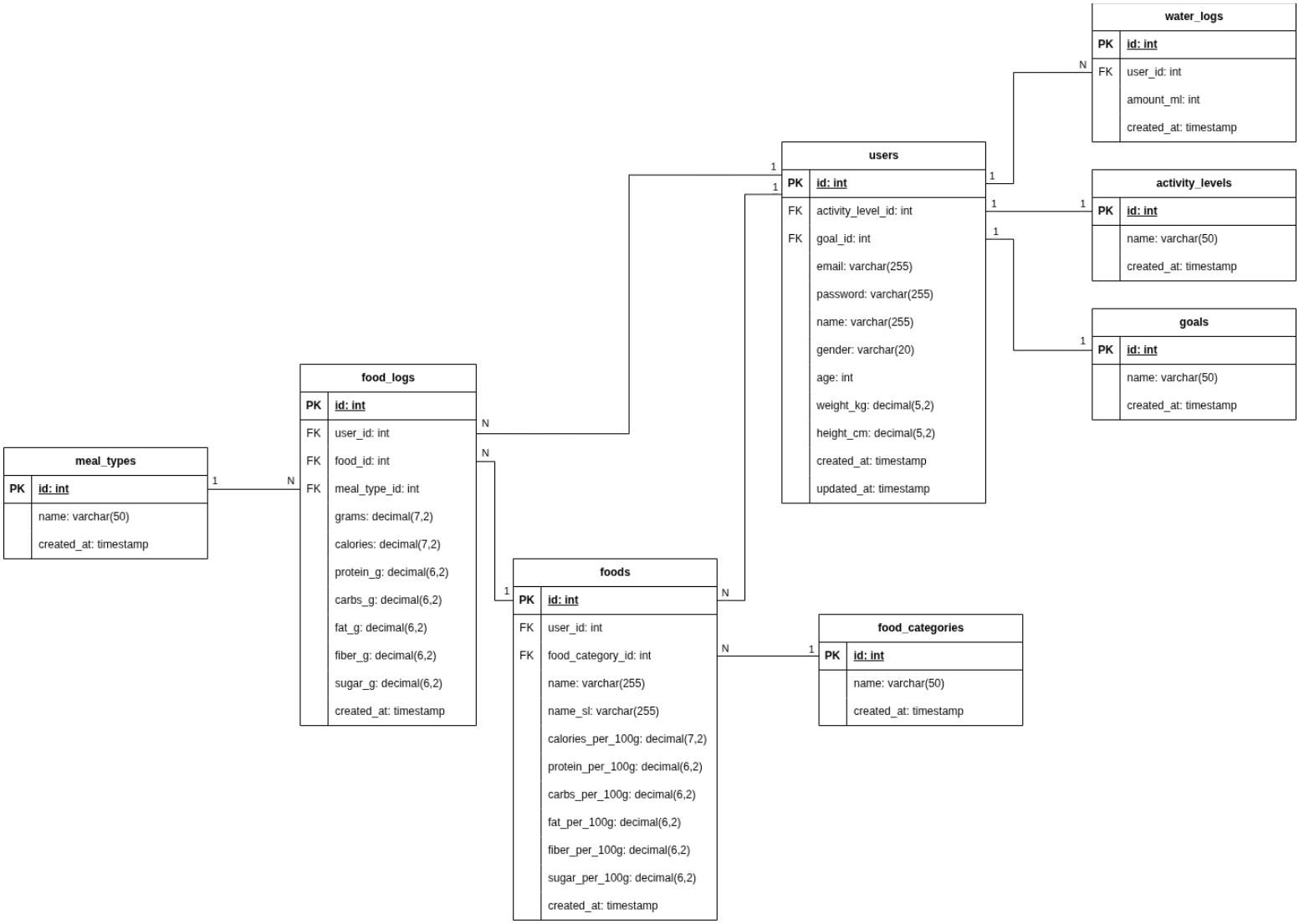
		protein_g	decimal(6,2)	The amount of protein of the food logged.
		carbs_g	decimal(6,2)	The amount of carbs of the food logged.
		fat_g	decimal(6,2)	The amount of fat of the food logged.
		fiber_g	decimal(6,2)	The amount of fiber of the food logged.
		sugar_g	decimal(6,2)	The amount of sugar of the food logged.
		created_at	timestamp	The time when the food log was created.
Meal-types	The meal types: breakfast, lunch, etc.	id	int	The unique identifier of the meal type.
		name	varchar(50)	The name of the meal type.
		created_at	timestamp	The time when the meal type was created.
Water_logs	The saved water logs.	id	int	The unique identifier of the water log.
		user_id	int	The unique identifier of the user creating the water log.
		amount_ml	int	The amount of water logged.
		created_at	timestamp	The time when the water log was created.
Activity_levels	The activity	id	int	The unique

	level of the user.			identifier of the activity level.
		name	varchar(50)	The name of the activity level.
		created_at	timestamp	The time when the activity level was created.
Goals	The set goals of the user.	id	int	The unique identifier of the goal.
		name	varchar(50)	The name of the goal.
		created_at	timestamp	The time when the goal was created.
Food_categories	The available food categories	id	int	The unique identifier of the food category.
		name	varchar(50)	The name of the food category.
		created_at	timestamp	The time when the food category was created.

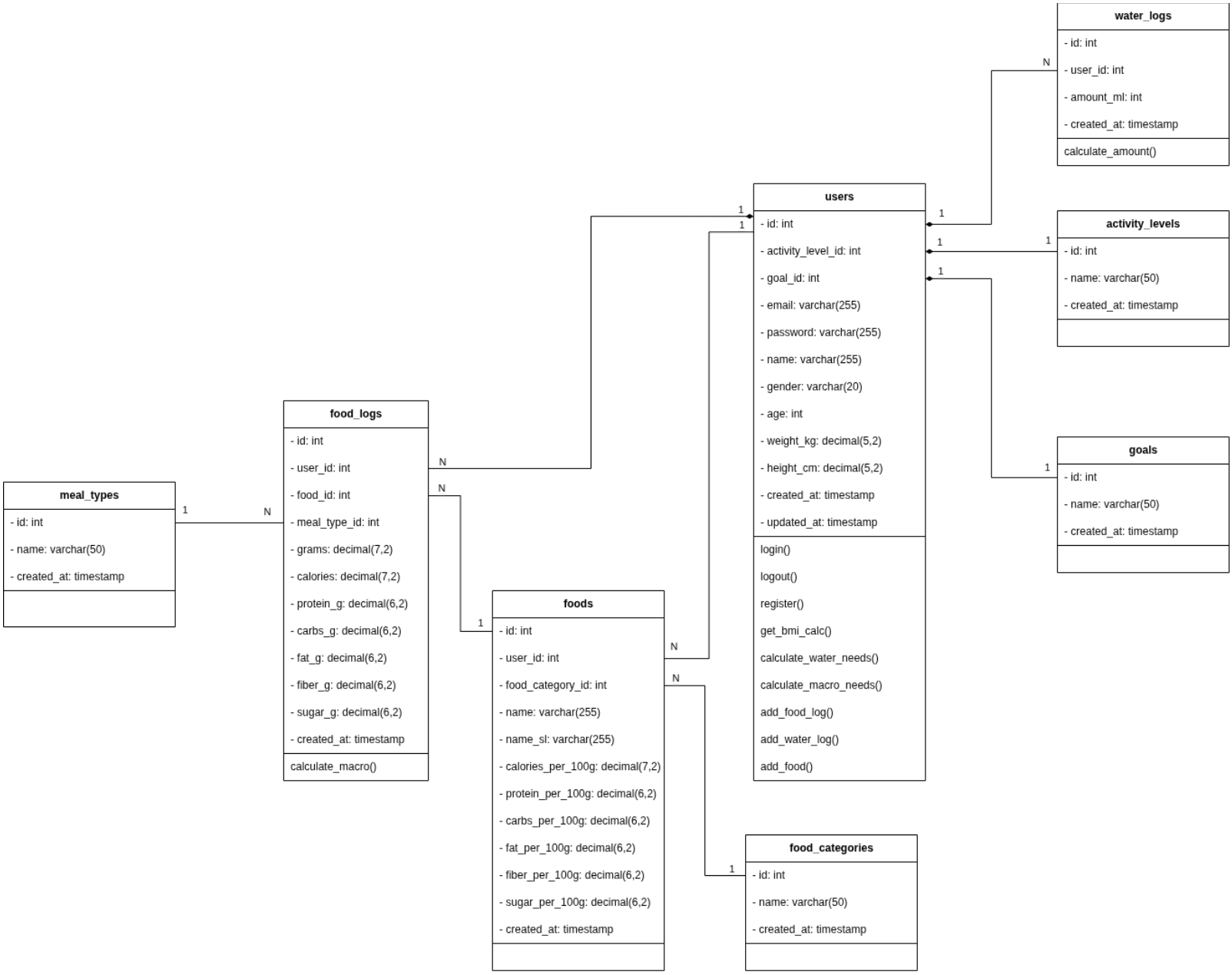
Entity Relationship Diagram



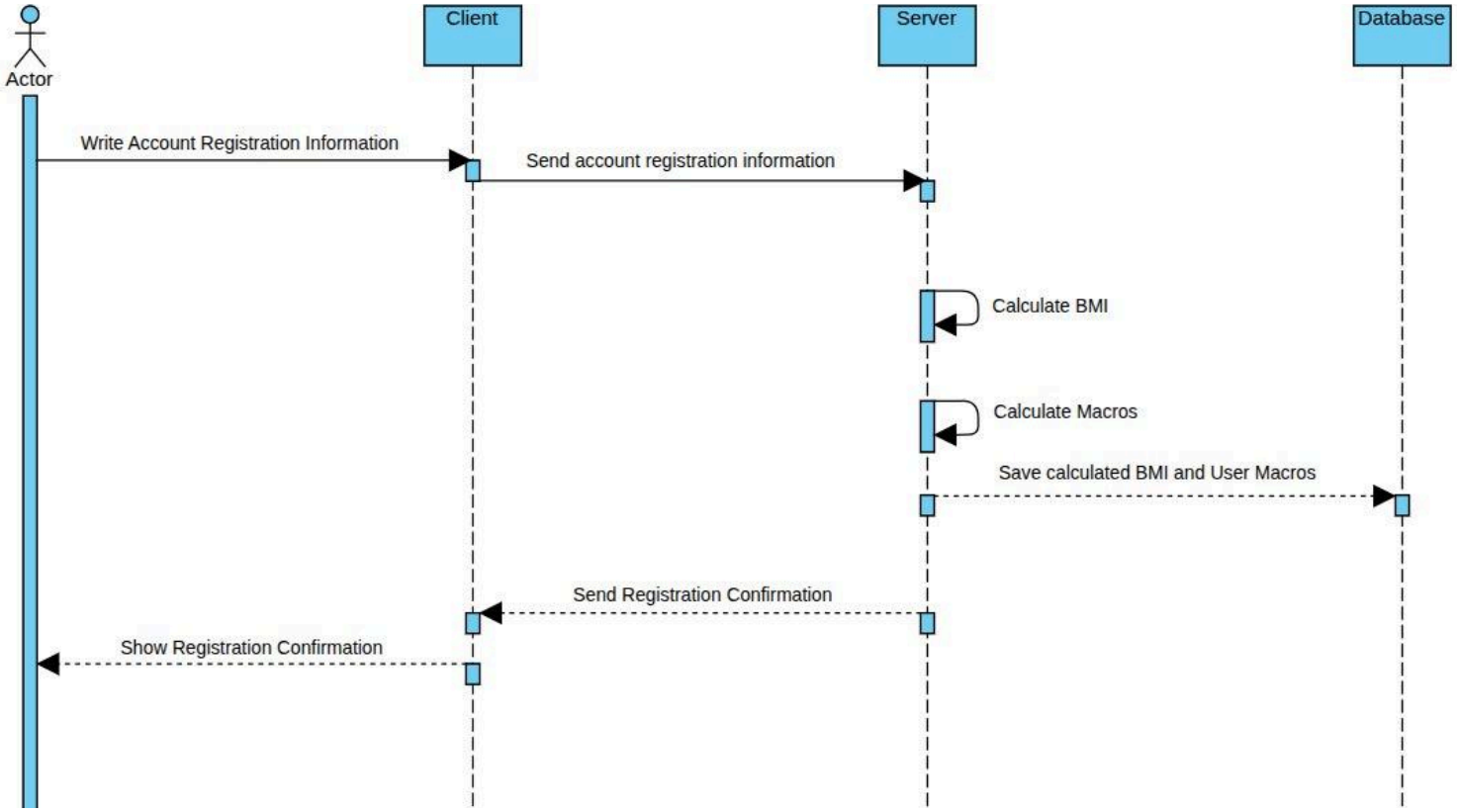
Relational Model



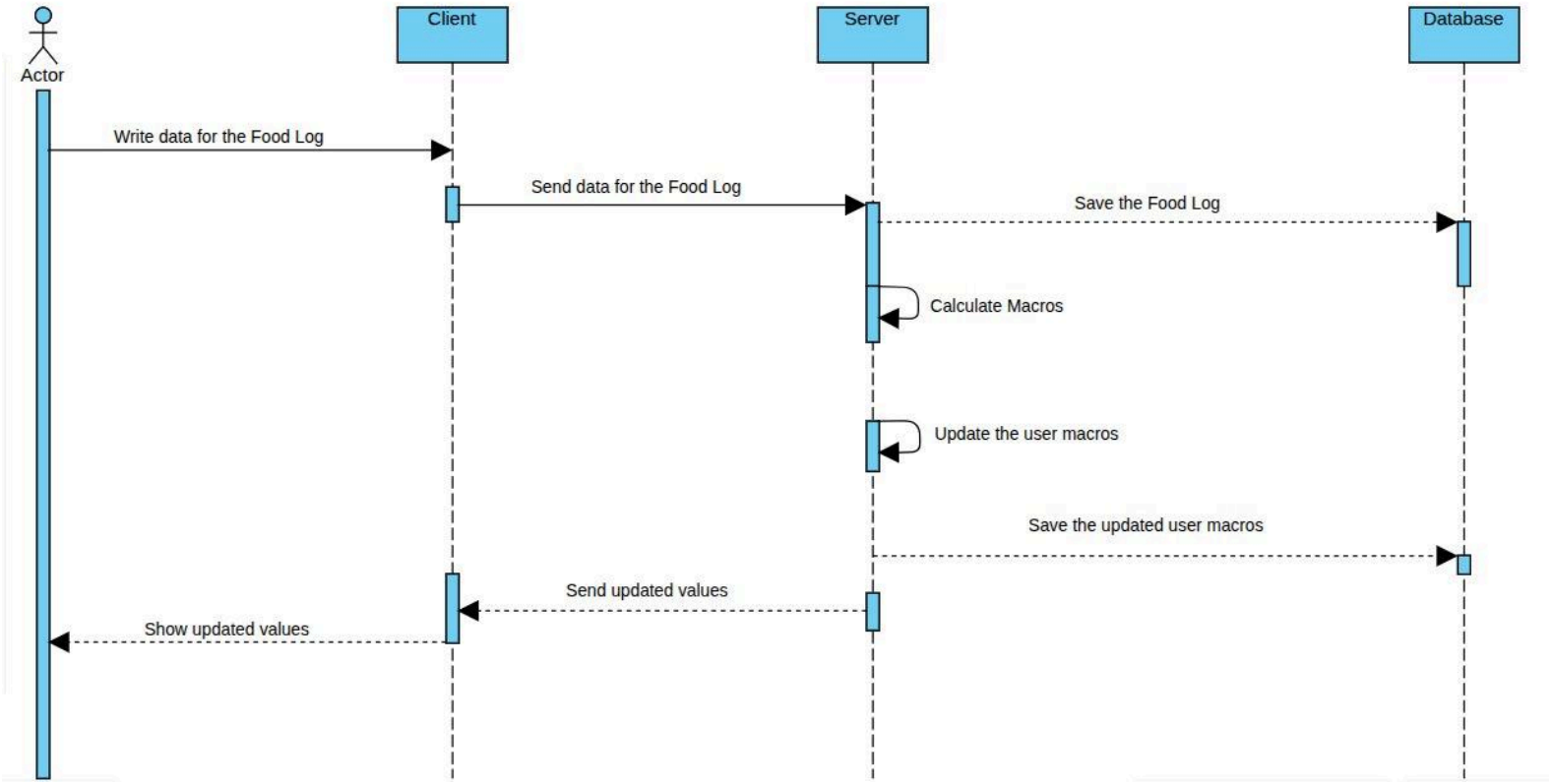
Class Diagram



Sequence Diagram - Account Registration



Sequence Diagram - User adds a Food Log Entry



Physical Database