1. API List with Expected Input/Output + Test Case Documentation

Endpoint	Method	Description	Input Example	Success Output
/register	POST	Register a new user	{"username": "yoyo123", "password": "mypassword"}	{"message": "User registered successfully"}
/login	POST	User login	{"username": "yoyo123", "password": "mypassword"}	{"message": "Login successful", "user_id": 1}
/create_meal	POST	Create a meal event	{"title": "Sushi Night", "description": "Join me!", "restaurant": "Wasabi", "time": "2025-08-01 19:00", "max_people": 4, "creator_id": 1}	{"message": "Meal event created", "meal_id": 1}
/join_meal	POST	Join a meal event	{"meal_id": 1, "user_id": 2}	{"message": "Joined meal successfully"}
/meals	GET	View all meals	(no input needed)	[{"id":1, "title":"Sushi Night", ...}]
/comment	POST	Post a comment	{"user_id": 2, "meal_id": 1, "content": "Count me in!"}	{"message": "Comment added"}

2. Database Design and Implementation

Table Name	Key Fields	Relationships
users	id, username, password	One-to-many with meals, comments, meal_participants

meals	id, title, description, restaurant, time, max_people, creator_id	Many-to-one with users (creator), one-to-many with comments & meal_participants
meal_participants	id, meal_id, user_id	Many-to-one with meals and users
comments	id, user_id, meal_id, content	Many-to-one with users and meals