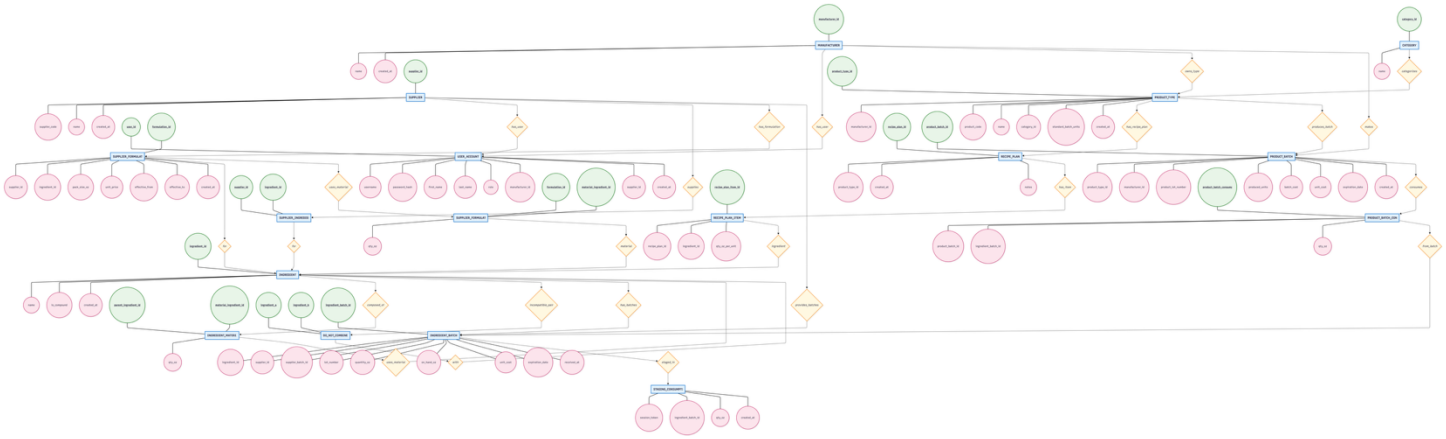
DBMS Project

Final ER Diagram



In the initial version of our schema, we designed a more generic, system with a broad `users' table, many possible role values, recipe versioning, separate health-risk rules/violations tables, and extensive use of `updated_at' triggers and indexes.

In the revised schema, we refocused and simplified the design in MySQL to align more tightly with the project requirements. Roles are now modeled explicitly through a `user_account' table with a constrained ENUM (`MANUFACTURER', `SUPPLIER', `VIEWER') and clear foreign keys to `manufacturer' and `supplier, instead of free-form roles and back-links from organizations. We also streamlined the ingredient and supplier side: we replaced multiple composition layers with a single `ingredient material' table enforced by a trigger to keep composition to one level, and we manage supplier pricing and pack sizes via `supplier formulation' and `supplier_formulation_material' with non-overlapping effective date ranges. Health and incompatibility logic has been simplified from dedicated rule/violation tables to a global `do_not_combine' list, supported by views and stored procedures that detect violations based on actual batch consumption. Finally, we cleaned up inventory and production by moving to a clearer separation between `ingredient_batch' (on-hand quantity, cost, and a 90-day expiry rule) and `product_batch' (units and costs only), with all consumption going through `product_batch_consumption', `staging_consumption', and `sp_record_product_batch'. Overall, both designs are normalized, but the current schema is more focused, less redundant, and relies on database-level triggers, views, and procedures to enforce the core business rules for the project.