LLM Automated Evaluation Results

Functional Requirements Evaluation Table

Based on the provided system overview and requirement document, I, acting as the requirement review expert, have conducted a comprehensive evaluation of the functional requirements according to your instructions. The evaluation process strictly followed these steps:

First, counting the number of data entities: I identified the core data entities managed within the system from the functional requirements (excluding external data sources or formats mentioned in interfaces). A total of 10 data entities were identified: Customer, Administrator, Product, Category, ProductCategory (association entity), ShoppingCart, CartItem, Order, OrderItem, and PaymentMethod.

Second, merging functional requirements: Requirements were merged according to specified rules:

• Rule: Merge requirements that involve the same data entity and the same functional point (e.g., multiple approval rules), and remove general system setting requirements (e.g., preferences, multilingual support – none found in this document).

• Merging Principle: Avoid over-merging (e.g., do not merge CRUD operations); merge only if the functional points are highly similar or target the same operation of the same entity.

• Result: Among the 29 functional requirements (FR-01 to FR-29) in the document, none met the criteria for merging (e.g., FR-21 Cancel Order and FR-22 Update Order Status both update the order status but differ in condition and purpose, thus not merged). No general setting requirements were found to exclude. Therefore, the total number of functional requirements remains 29 after merging.

Third, identifying hallucinated requirements: Definition: A hallucinated requirement introduces a functional operation involving a data entity that cannot be associated with any existing entities, leading to an entity-relationship (E-R) diagram becoming disconnected (i.e., having isolated nodes). Evaluation criteria: Check whether each functional point introduces a new entity or results in a disconnected E-R diagram (existing entities are extracted from functional requirements).

• Result: None of the 29 requirements are hallucinated since all involved data entities exist in the system and the E-R diagram remains connected (e.g., Customer, Order, Product are connected via relational paths).

The complete evaluation table is as follows. Each functional requirement was evaluated independently, with justification based on the document content. The evaluation is rigorous yet reasonable, identifying a few minor issues (e.g., input details of FR-18 are not fully defined, but overall acceptable).

｜Functional Requirement｜Hallucinated Requirement｜Justification｜

｜FR-01 Customer Registration｜No｜Involves the Customer entity, input and output definitions are complete, E-R diagram is connected, conforms to the core user registration behavior of the system.｜

｜FR-02 User Login｜No｜Users log in via email/password, session establishment logic is clear, associated with Customer/Administrator entities, E-R structure is valid.｜

｜FR-03 User Logout｜No｜Involves the Session entity, user logout process is clearly defined, session termination behavior aligns with system structure.｜

｜FR-04 Update Customer Information｜No｜Updates Customer information, input fields match output behavior, system structure supports the operation.｜

｜FR-05 View Customer Information｜No｜Views personal information based on CustomerID, data paths are clear, entity relationships are explicit.｜

｜FR-06 Add Product｜No｜Involves Product and Category entities, product addition logic aligns with category structure, paths are connected.｜

｜FR-07 Update Product｜No｜Modifies Product data, structure remains connected, data update path is clear.｜

｜FR-08 Delete Product｜No｜Deletes the specified Product, the entity exists in the system, deletion logic is controllable.｜

｜FR-09 Browse Products｜No｜Displays Product list as a standard operation, Product entity exists, data structure supports this behavior.｜

｜FR-10 View Product Details｜No｜Queries detailed information via ProductID, involves Product and Category entities, system connectivity is good.｜

｜FR-11 Add Category｜No｜Adds a new Category entity, name and description fields are clear, system structure supports it.｜

｜FR-12 Update Category｜No｜Updates Category attributes, modification logic is consistent with data structure, paths are connected.｜

｜FR-13 Delete Category｜No｜Deleting the Category entity requires checking reference constraints, process logic is reasonable.｜

｜FR-14 View Category Details｜No｜Displays Category details and associated products, system supports many-to-many association display.｜

｜FR-15 Add to Shopping Cart｜No｜Relationships among CartItem, Product, and Customer are clear, product addition logic is complete.｜

｜FR-16 Update Shopping Cart｜No｜Modifies CartItem quantity or removes items, involved entities have clear structure, behavior is reasonable.｜

｜FR-17 View Shopping Cart｜No｜Views cart contents and total, CartItem structure and Customer association are clear, paths are reachable.｜

｜FR-18 Place Order｜No｜Order generation involves multiple entities, fields and dependent interfaces are reasonable, overall path is closed.｜

｜FR-19 Browse Orders｜No｜Order summary display conforms to system behavior specifications, Order entity is clearly linked to Customer.｜

｜FR-20 View Order Details｜No｜Displays detailed order data, Order and OrderItem structure is consistent.｜

｜FR-21 Cancel Order｜No｜Order cancellation logic exists for stock recovery, system supports status update and entity write-back.｜

｜FR-22 Update Order Status｜No｜Order status modification is reasonable, related notification behavior is compatible with system architecture.｜

｜FR-23 Manage Inventory｜No｜Product inventory update is a basic function, system supports this operational path.｜

｜FR-24 Manage Customer Payment Methods｜No｜Payment methods are sub-properties of Customer, entity relationships are clear, management logic is complete.｜

｜FR-25 Assign Product to Category｜No｜Association managed via intermediate table ProductCategory, entity structure is reasonable, path is clear.｜

｜FR-26 Manage Administrator Accounts｜No｜Administrator account is an independent entity, login verification process supports account management needs.｜

｜FR-27 Delete Customer｜No｜Deleting Customer includes cascade deletion logic (e.g., payment methods), structure is reasonable, system is controllable.｜

｜FR-28 Delete Order｜No｜Removes Order and its sub-items, Order and OrderItem structure is closed, deletion process is standardized.｜

｜FR-29 Update Order Items｜No｜Modifying OrderItem quantity automatically updates stock and order total, inter-entity logic is connected.｜

Summary

• Total number of data entities involved in the document: 10 (Customer, Administrator, Product, Category, ProductCategory, ShoppingCart, CartItem, Order, OrderItem, PaymentMethod)

• Total number of functional requirements after merging same functional points: 29 (no merging or removal)

• Number of hallucinated requirements: 0 (all requirements involve existing entities, E-R diagram is connected, no isolated nodes)

• Number of non-hallucinated requirements: 29 (since no hallucinations, all are valid)

Evaluation Note:

• E-R diagram connectivity ensures no hallucinations: Customer is the core, linking Order and PaymentMethod; Order connects to OrderItem and Product; Product links to Category; ShoppingCart links Customer and Product.

Please let me know if further revisions are required!