LLM Automated Evaluation Results

Functional Requirement Evaluation Table

According to the user’s instructions, I first extracted data entities from the functional requirements document. Data entities refer to core data objects managed within the system, excluding external interfaces or external data sources. Based on the descriptions of the functional requirements, I identified the following data entities:

User (including customer and admin), Product, Category, Cart, CartItem, Order, Inventory, Plugin, PurchaseHistory, Log.

A total of 10 data entities were identified.

Next, I aggregated the functional requirements according to the user-specified rules:

Rule 1: Functional points that target the same data entity with the same operation type were merged into a single functional requirement (e.g., modifying, removing, or updating quantity of CartItem are merged as one).

Rule 2: General system settings requirements (e.g., preference settings, multilingual support) are to be excluded; however, such requirements were not present in this document, so none were excluded.

Merge Details:

FR-08 (Modify Cart Item), FR-09 (Remove Cart Item), and FR-22 (Manage Quantity) all target the CartItem entity and involve “modification” operations (including updating quantity and removing items), thus merged into: Manage Cart Items (Modify Quantity or Remove).

FR-20 (Calculate Total Cost) is a core substep of FR-21 (View Cart Summary), as cost calculation is a prerequisite for summary viewing, so they were merged into: View Cart Summary (Include Calculate Total Cost).

Other requirements were not merged because they involve different data entities or different operation types (e.g., login and logout both involve the User entity but represent different functional points; creating and updating products differ in operation).

After merging, there are 21 functional requirements in total (originally 24, reduced by 3 through merging).

Then, I evaluated each merged functional requirement, including:

Hallucination Requirement Judgment: Based on the definition (a requirement is considered a hallucination if the operations introduced involve data entities that cannot be connected to any system entity, causing the E-R diagram to be disconnected). In this system’s E-R diagram, the core entities are User and Product, and all others are connected through relationships (e.g., CartItem connects Product and User), thus no hallucinations were found.

Evaluation results are shown in the table below. Each requirement has been rigorously reviewed, and no hallucinations were detected. All requirements were completely referenced and testable (see rationale in the table).

| Functional Requirement | Hallucination Requirement | Evaluation Rationale |

| FR-01: Customer Registration | No | Involves User entity, inputs and outputs are clearly defined, system structure supports registration |

| FR-02: Customer Login | No | Associated with User entity, login logic is complete, supported by external interface |

| FR-03: Customer Logout | No | Involves User entity and session management, requirement structure is clear |

| FR-04: Admin Login | No | Admin is a subtype of User, login logic is explicit |

| FR-05: Admin Logout | No | Admin logout is consistent with User, structurally reasonable |

| FR-06: View Product Details | No | Involves Product entity, display process is clear |

| FR-07: Add to Cart | No | Involves Product and CartItem, update logic is reasonable |

| Combined: Manage Cart Items (FR-08,09,22) | No | CartItem modification operations are consistent, structure and interface are unified, merging is reasonable |

| FR-10: Checkout Order | No | Involves Cart and Order, complete workflow, supported by external payment system |

| FR-11: Confirm Order | No | Order is linked with Inventory, confirmation logic is sound |

| FR-12: Send Order Confirmation Email | No | Involves Order and mail service, process is clear |

| FR-13: Create Product | No | Product creation follows standard behavior, write operation is structurally supported |

| FR-14: Update Product Information | No | Involves Product data modification, update process is well defined |

| FR-15: Manage Inventory | No | Inventory is tightly linked with Product, operational path is complete |

| FR-16: Categorize Products | No | Involves Product and Category, classification logic is consistent |

| FR-17: Manage Plugin | No | Plugin is a system extension module, management behavior is clearly defined |

| FR-18: Access Plugin Documentation | No | Plugin documentation access is clearly defined, entity linkage is valid |

| FR-19: View Purchase History | No | Purchase history is associated with User entity, viewing logic is connected |

| Combined: View Cart Summary (FR-20,21) | No | Viewing cart content and summary are part of the same behavior chain, merging is justified |

| FR-23: Manage Purchase History | No | Purchase history aligns with Order data, update logic is standardized |

| FR-24: Manage User | No | User management is based on the User entity, update path is clear |

Summary

The document involves 10 data entities: User, Product, Category, Cart, CartItem, Order, Inventory, Plugin, PurchaseHistory, Log.

After merging based on identical operation points, there are 21 functional requirements in total.

There are 0 hallucination requirements (all requirements are connected to the system architecture, with weak connectivity ensured in the E-R diagram).

There are 21 non-hallucination requirements (100% non-hallucination rate).

Evaluation Notes:

The assessment was performed rigorously.

Although the user suggested that “there might be a few problematic items,” no non-compliant requirements were found in the analysis. For example, operations like managing plugin documentation and purchase history may seem peripheral but are logically connected to the system via Plugin or Order entities with clearly defined input/output.

Merging was applied to ensure consistent granularity in functional requirements (e.g., merging shopping cart operations), while over-merging was avoided (e.g., login/logout was not merged due to differing behavior).、