LLM Automated Evaluation Results

### **Functional Requirement Evaluation Table**

The following is a comprehensive evaluation based on the user-provided requirement documents (including the system overview and functional requirements sections). The evaluation is based on the following principles:

Data Entity Extraction: Only core internal system entities to be managed (such as asset, user, etc.) are counted, excluding external data sources or formats in interface descriptions. The entities mentioned in the requirement documents include 8 in total: Asset, User, Permission, Application, Approval, ApprovalProcess, Log, and Report.

Functional Requirement Merging: According to the defined rules, requirements with the same functional point under the same entity were merged (e.g., FR-02, FR-03, FR-04, FR-05, and FR-17 all concern “Submit Application” and are merged into one). General configuration-type requirements (like user preferences) are not present and thus not removed. After merging, the number of functional requirements is reduced from 35 to 31.

Hallucinated Requirement Judgment: Judged based on weak connectivity in the E-R diagram (all entities must be connected through some path). A hallucinated requirement is defined as one involving an entity that cannot be linked to the rest of the system, resulting in a disconnected E-R graph. In this system, all entities (Asset, User, Permission, Application, Approval, ApprovalProcess, Log, Report) are interrelated (e.g., Application connects Asset and User; Approval is tied to Application; Log records all actions), so there are no hallucinated requirements.

Judgment Principles: Strict but reasonable. Each requirement is independently assessed with no omissions. Hallucinated requirements: 0.

#### **Evaluation Table**

｜Functional Requirement｜Hallucinated Requirement｜Judgment Reason｜

｜FR-01: Asset Registration Function｜No｜Involves the Asset entity and is linked with system operations. Input and output are clearly defined in the requirement, forming a traceable system behavior path without breaking entity relations.｜

｜FR-SubmitApplication: Submit Application Function (merged from FR-02, FR-03, FR-04, FR-05, FR-17)｜No｜All types of application submissions revolve around the core entity ApplicationRecord, with clear subtypes. The process connects seamlessly with system functions and does not deviate.｜

｜FR-06: Asset Information Query Function｜No｜The query operation directly acts on the Asset entity. The query conditions and returned results are clearly described and form a valid system operation.｜

｜FR-07: Asset Status Update Function｜No｜Status change is part of asset lifecycle management. The function is fully described and logically consistent within the system flow.｜

｜FR-08: Create New User Function｜No｜New users are introduced as entities in the system. Fields and behaviors during the registration process are handled by the system’s management module and are not detached.｜

｜FR-09: Delete User Function｜No｜User deletion is based on User ID and updates the system state, closely tied to the existing User entity.｜

｜FR-10: Modify User Information Function｜No｜User information update is a typical entity attribute update behavior and aligns with the system’s data consistency requirement.｜

｜FR-11: View User Information Function｜No｜The query target is the User entity. The information display process matches the system structure with no detachment.｜

｜FR-12: Disable User Account Function｜No｜The disable action reflects system control mechanisms. The change of User entity status is well represented in the system design.｜

｜FR-13: Permission Assignment Function｜No｜Permissions are bound to the User entity and are central to system access control. The function setup is reasonable and valid.｜

｜FR-14: Revoke Permission Function｜No｜Permission revocation relates to both the Permission and User entities. The behavior logic is inferable from system design and is valid.｜

｜FR-15: View Permission Function｜No｜Querying permission information reflects the visualization needs of the management module. The operation target is clear and aligns with the system function structure.｜

｜FR-16: Modify Permission Function｜No｜Permission configuration is a core field of the Permission entity. Modification reflects configuration behavior on the admin side with no unreasonable elements.｜

｜FR-18: View Application Status Function｜No｜The demand for approval transparency is clear. The core dependency is the Application entity, and the display mechanism aligns with business logic.｜

｜FR-19: Revoke Application Function｜No｜The revoke process is closely related to the Application entity. The behavior is traceable and dependent on the original application action without independent hallucination.｜

｜FR-20: Submit Approval Function｜No｜The approval submission stage is constructed around the Approval entity. Despite some undefined fields, the logic path is smooth, making it a valid requirement.｜

｜FR-21: View Approval Progress Function｜No｜The visualization operation of the approval process reflects system transparency and links consistently with the Approval entity.｜

｜FR-22: Cancel Approval Function｜No｜Together with approval submission and process control, it forms a complete loop. The entity is clearly dependent and well described.｜

｜FR-23: View Approval Record Function｜No｜Approval history serves as the trace management of system operation and is clearly tied to the Approval entity, with no standalone detachment.｜

｜FR-24: Operation Log Recording Function｜No｜Log is a fundamental system monitoring entity. Recorded behaviors have clear system triggers and processing points.｜

｜FR-25: View Operation Log Function｜No｜Log queries are based on existing records and meet the need for behavior traceability.｜

｜FR-26: Export Operation Log Function｜No｜The export function is based on the existence of the Log entity and addresses auditing needs with a clear path.｜

｜FR-27: View Approval Process Function｜No｜The approval flowchart is a visualization of approval control. It is clearly bound to the ApprovalProcess entity and valid as a function.｜

｜FR-28: Modify Approval Process Function｜No｜Modifying processes is an effective operation in the system’s process control module. The change behavior is clear, with a complete entity path.｜

｜FR-29: Manage Asset Function｜No｜Management actions include adjustments to asset attributes and status, matching the full scope of the Asset entity’s management functions.｜

｜FR-30: Manage User Function｜No｜User management is a control-level operation in the system. The function covers creation, modification, and other processes, logically coherent.｜

｜FR-31: Manage Permission Function｜No｜Permission management integrates configuration and assignment, built upon the Permission and User entities with clear entity-level support.｜

｜FR-32: Manage Approval Process Function｜No｜Controlling and adjusting approval processes belongs to advanced configuration. The targets are clear and the description precise.｜

｜FR-33: Generate Report Function｜No｜Report generation involves statistics and formatted display of data from existing entities. Although some fields are not fully defined, the function is effectively dependent on valid entities.｜

｜FR-34: View Report Function｜No｜Report queries rely on already generated reports. The behavior is clear, the goal specific, and does not deviate from system structure.｜

｜FR-35: Export Report Function｜No｜Exporting reports is based on existing reports. The operation logic is a standard system feature, and the requirement is valid and reasonable.｜

### **Summary**

The entire document involves 9 data entities: Asset, User, Permission, Application, Approval, ApprovalProcess, Log, Report, and ApprovalRecord.

After merging requirements with the same functional point, there are 31 functional requirements (originally 35, with FR-02, FR-03, FR-04, FR-05, and FR-17 merged).

There are 0 hallucinated requirements: all requirements involve data entities that are connected to the core system architecture (weakly connected E-R diagram, no isolated entity).

There are 31 non-hallucinated requirements in total (no hallucinations).