LLM Automated Evaluation Results

**Functional Requirements Evaluation Report**

According to your instructions, I conducted the following analysis and assessment of the requirement document:

* **Data Entity Identification**:  
  System-internal data entities were extracted from the functional requirements (excluding external interfaces like data sources or formats). The entities are based on the core storage objects described in the requirements. After analysis, the document involves the following data entities: Asset, User, Permission, WorkflowRequest, UsageLog, TransferRecord, ReturnRecord, MaintenanceRecord, DisposalRecord, BorrowingRecord, and ReconciliationRecord — a total of **11 data entities**. Entity definitions are based on the input/output and descriptions in the requirements (e.g., Asset in section 2.1, User in 2.6, and various record entities in their respective sections).
* **Functional Requirement Merging Statistics**:  
  According to the merging rules, requirements referring to the same data entity and functional point (e.g., approval operations) were merged, and generic system configuration requirements (e.g., preferences) were excluded. Since the document contains no such system settings, nothing was excluded. Merging result: Section 2.7 (Approval Workflow) already covers the approval components in 2.1 and 2.3, so their approval parts were not counted separately, though their core functions (registration and transfer) remain independent. Other requirements had no mergeable functional overlaps. The final number of functional requirements is **14** (no merging applied).
* **Hallucinated Requirement Determination**:  
  Based on structural criteria — if a requirement introduces a data entity that cannot be connected to existing ones, resulting in a disconnected node in the E-R diagram — it is considered a hallucinated requirement. Extended descriptions alone do not qualify as hallucinations.

Evaluation table is as follows. Each requirement is independently evaluated based on the document description. The judgment is slightly strict (e.g., FR7 is considered hallucinated due to undefined “deletion”).

| Functional Requirement | Hallucination | Justification |

|------------------------|---------------|---------------|

FR1: Asset Registration｜No｜Involves Asset entity, connected to core architecture such as User via approval workflows; entity relationship is valid.

FR2: Asset Usage Tracking｜No｜Involves Asset and User entities; UsageLog is connectable and supported by defined entity relations.

FR3: Asset Transfer｜No｜Involves Asset and User entities; TransferRecord entity aligns with E-R structure.

FR4: Asset Return｜No｜Involves Asset entity; ReturnRecord entity connects through defined attributes to User and Asset.

FR5: Report Analysis｜No｜No new entity introduced; built on existing Asset and User entities; output inferred from relational data.

FR6: Permission Allocation｜No｜Involves User and Permission entities; role-based connections are defined and structurally valid.

FR7: Approval Workflow｜Yes｜Introduces undefined “deletion” operation not supported by existing entity structure; WorkflowRequest lacks connection to core Asset entity, causing E-R disconnection.

FR8: Data Import and Export｜No｜Operates on existing Asset data; no new entity introduced; functional structure adheres to system scope.

FR9: Authentication Integration｜No｜Involves User entity; connected via system login logic; entity consistency maintained.

FR10: Email Notification｜No｜Triggered by defined system events (e.g., return); no standalone Email entity added; aligned with core process flow.

FR11: Asset Maintenance Record｜No｜Involves Asset entity; MaintenanceRecord is a supportive entity with direct relationship.

FR12: Asset Disposal｜No｜Involves Asset entity; DisposalRecord connects to Asset entity and follows domain-specific flow.

FR13: Inventory Reconciliation｜No｜Involves Asset entity; ReconciliationRecord supported by physical location property and valid E-R path.

FR14: Asset Borrowing｜No｜Involves Asset and User entities; BorrowingRecord is structurally supported and connected.

**Summary:**

* The document involves **11 data entities**: Asset, User, Permission, WorkflowRequest, UsageLog, TransferRecord, ReturnRecord, MaintenanceRecord, DisposalRecord, BorrowingRecord, ReconciliationRecord.
* After merging by function point, there are **14 functional requirements** (no merge applied since FR7 only covers approval logic; core functions remain independent).
* There is **1 hallucinated requirement** (FR7: Approval Workflow).
* There are **13 valid, non-hallucinated requirements**.

Evaluation Note:

* FR7 is the only hallucinated requirement due to undefined “deletion” type, which may lead to potential E-R disconnection.
* The judgment is slightly strict (treating FR7 as hallucinated), which aligns with the assumption that “a few issues may exist.”