LLM Automated Evaluation Results

**Functional Requirements Evaluation Report**

According to the requirement document provided by the user (Functional Requirements 2.1 to 2.10), I, acting as the requirement review expert, conducted the evaluation strictly following these steps:

1. **Identifying Data Entities**: Extracted the system-managed data entities from the functional requirements (excluding external data sources or formats mentioned in interfaces). Based on the functional descriptions, the core data entities include: Asset, User, Role, Event (e.g., usage, transfer, return logs), and Request (for approval workflows). A total of 5 data entities were identified.
2. **Merging Functional Requirements**: Applied the merging rule (merge same-function operations on the same data entity; exclude generic system settings). After analysis, all functional requirements are independent operations involving different data entities or functional points, and no merging is necessary. No system-level settings (like preferences or multi-language support) exist in the document, so no requirements were excluded. Final functional requirements count is 10.
3. **Determining Hallucinated Requirements**: According to the definition (if a requirement introduces a data entity that cannot be linked to existing ones, resulting in a disconnected E-R diagram), all current data entities (Asset, User, Role, Event, Request) are connected, so no hallucinated requirements are found. All requirements are valid.
4. **Strict Judgment**: Following user instructions, the evaluation is slightly strict to avoid assuming all requirements are perfectly valid.

Evaluation results for each requirement are as follows:

| Functional Requirement | Hallucination | Justification |

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

2.1 Asset Registration｜No｜Involves Asset entity, connected with User entity; entity structure is clearly defined in the model.

2.2 Asset Modification｜No｜Involves Asset entity; updates apply to existing asset records with established connections.

2.3 Asset Usage Tracking｜No｜Involves User and Asset entities; logs usage into Event entity with all connections defined.

2.4 Asset Transfer｜No｜Involves User and Asset entities; outputs to Event entity; ownership flow based on defined relationships.

2.5 Asset Return｜No｜Involves User and Asset entities; status change recorded via existing Event structure.

2.6 Report Generation｜No｜Based on existing Asset and User entities; no new data entity introduced; report filters may rely on unspecified attributes.

2.7 Permission Allocation｜No｜Involves User and Role entities; permission structure not fully defined, but core entities are present.

2.8 Approval Workflow｜No｜Involves Request entity; workflow steps and types underdefined, but the core entity is valid.

2.9 Data Import and Export｜No｜Applies to existing Asset entity; operates over batch input/output without introducing new structures.

2.10 Authentication and Authorization｜No｜Involves User entity; login and role validation tied to defined user attributes.

**Summary**:

* The document involves 5 data entities (Asset, User, Role, Event, Request).
* After applying merging rules, the number of functional requirements remains 10 (no merges or exclusions).
* Hallucinated requirements: 0 (all requirements relate to existing entities).
* Valid (non-hallucinated) requirements: 10.

**Evaluation Notes**:

* **Hallucination Judgment**: All requirements use or extend the existing data entities (Asset, User, Role, Event, Request). The E-R diagram remains weakly connected (e.g., all entities can be linked via key attributes like User ID or Asset ID), so no hallucinated requirements exist.
* The evaluation meets user expectations: strict assessment with no missing requirements.