LLM Automated Evaluation Results

Functional Requirement Evaluation Table

Based on your request, I, as a requirement review expert, have conducted a detailed assessment of the provided functional requirements document. The evaluation was based on the following steps:

1. Identification of Data Entities: Core data entities to be managed within the system were extracted from the functional requirements (excluding external data sources or formats mentioned in interfaces). The core data entities include: Asset, User, Approval Process, and Permission. Approval Record is considered part of the Approval Process and thus is not counted separately. Total data entities: 4.

2. Merging of Functional Requirements: According to the rules, similar functional points were merged to ensure consistency of granularity:

• Functional operations on the same data entity were merged.

• Generic system configuration requirements (such as preferences or multilingual support) were excluded, but this document contains none.

• After merging, the total number of functional requirements is 16 (originally 19, reduced by 3; details of merging are shown in the table below).

3. Identification of Hallucinated Requirements: The criterion is that if the operation introduced in the requirement involves a data entity that cannot be connected to existing entities, leading to a non-weakly connected E-R diagram, then it is considered a hallucination. All requirements involve existing entities; thus, no hallucinated requirements exist.

The evaluation table is as follows. Each functional requirement is based on the merged description, and the justification includes analysis regarding hallucination.

｜Functional Requirement｜Hallucinated Requirement｜Justification｜

｜Register Asset｜No｜Involves the Asset entity, with clearly defined inputs and outputs. The entity has weak connections to other parts of the system and is not isolated.｜

｜Delete Asset｜No｜Involves the Asset entity, associated with the approval process and asset handling. The entity connectivity is sound.｜

｜Transfer Asset｜No｜Involves Asset and Approval Process entities. The change of asset ownership can be traced within the system; entity relationships are clear.｜

｜Return Asset｜No｜Involves Asset and Approval Process entities. The return process is defined within the system, forming a closed data structure.｜

｜Query Asset｜No｜Involves the Asset entity. The asset information display path is clear; the entity is not isolated.｜

｜Generate and View Reports｜No｜Involves the Asset entity. Although some fields are undefined, the data source is clear, and the entity structure is connected.｜

｜Add User｜No｜Involves the User entity. The creation operation is complete, and the entity is connected to system authentication and permission structures.｜

｜Delete User｜No｜Involves the User entity. The deletion operation is linked to the approval process, forming a closed data structure. The entity is not isolated.｜

｜Manage User｜No｜Involves the User entity. Operations like permission modification are supported by entities, with rational input/output and connected structure.｜

｜Initiate Approval Process｜No｜Involves the Approval Process entity, which is associated with User or Asset entities. The structure is coherent.｜

｜Approve Approval Process｜No｜Involves the Approval Process entity. Status updates are based on existing data structures, and the path exists.｜

｜Cancel Approval Process｜No｜Involves the Approval Process entity. Although the term “related information” is vague, the entity itself is not isolated.｜

｜Query Approval Process｜No｜Involves the Approval Process entity. The query function is based on an existing structure, and entity relationships are clear.｜

｜Assign Permissions｜No｜Involves the Permission entity, associated with the User entity. Although some details are undefined, the entities are connected.｜

｜Approve Permission Application｜No｜Involves Permission and Approval Process entities. The requirement description aligns with the system structure and is not isolated.｜

｜Generate Approval Record｜No｜Involves the Approval Process entity. Approval records are part of this entity, and logical paths between entities exist without disconnection.｜

Summary:

The entire document involves seven data entities (Asset, User, Approval Process, Permission, Approval Record, User, and Report). After merging similar functional points, there are 19 functional requirements in total (merged details: FR-06 and FR-07 merged into “Generate and View Reports”; FR-10 and FR-15 merged into “Manage User”; FR-14 and FR-19 merged into “Query Approval Process”). Among them, there are 0 hallucinated requirements (all requirements involve existing entities, and the E-R diagram is weakly connected: Asset is linked to User through approvals; Permission is connected to User; Approval Process is associated with both Asset and User). There are 16 non-hallucinated requirements (0 hallucinated).