LLM Automated Evaluation Results

Functional Requirement Evaluation Table

According to user requirements, I first conducted an analysis of the requirement document. The key steps and results of the evaluation are as follows:

First, data entity extraction and statistics:

Entities that need to be managed within the system were extracted from the functional requirements document, excluding external data sources or formats mentioned in interface descriptions.

The identified entities include: Email, Folder, Calendar Event, Contact, Distribution Group, Reminder Task, Mailbox Account, User Account, Admin Account, Shared Account, Mail Retention Policy, Archiving Policy, Mail Flow Record, Backup Log, Restore Log.

Total number of data entities: 15. The definitions are as follows:

Email: email content, status, etc.

Folder: email folder structure.

Calendar Event: event title, time, location, etc.

Contact: contact name, email, etc.

Distribution Group: group membership relations.

Reminder Task: reminder time, method, etc.

Mailbox Account: mailbox account configuration.

User Account: user registration information.

Admin Account: administrator account information.

Shared Account: shared account configuration.

Mail Retention Policy: email retention rules.

Archiving Policy: email archiving rules.

Mail Flow Record: captured mail flow data.

Backup Log: backup operation logs.

Restore Log: restore operation logs.

Second, functional requirement merging and statistics:

According to rules:

Rule 1: Merge same functional points under the same entity (e.g., same operations under different conditions into a generalized requirement).

Rule 2: Eliminate system-wide preference-type requirements (e.g., language, settings).

Analysis results:

No functional points were found to be mergeable (all operations target different actions or entities, unlike general “asset approval” cases).

No preference-type system requirements were found (e.g., no multilingual or preference settings), thus none were eliminated.

Total number of functional requirements remains 40 (FR-01 to FR-40), with no merges or removals.

Third, hallucinated requirement judgment:

Evaluation criterion: if the data entity involved in the functional requirement is not connected to existing entities, or makes the E-R diagram not weakly connected, it is a hallucinated requirement.

E-R connectivity analysis shows all core entities (e.g., User Account) are connected to others via system operations (e.g., users interacting with email, calendar, contact; admin managing policy; logs recording operations). Thus, the E-R graph is weakly connected.

Evaluation result: All 40 requirements are not hallucinated. Each requirement involves defined entities and is logically linked through operations, maintaining E-R connectivity.

Functional Requirement Evaluation Table

｜Functional Requirement｜Hallucinated Requirement｜Evaluation Justification｜

｜FR-01: Email Creation｜No｜The function is clearly defined with clear input/output paths; the system can trigger draft save or email send, exhibiting defined system behavior.｜

｜FR-02: Email Sending｜No｜The function involves the Email entity; sending action triggers status update and feedback with defined system response.｜

｜FR-03: Email Receiving｜No｜The system can receive emails from server and store them in inbox; inputs and outputs are linked and supported.｜

｜FR-04: Email Deletion｜No｜The deletion behavior is clearly defined and valid for the Email entity with standard triggers and feedback.｜

｜FR-05: Email Search｜No｜The system supports keyword queries and returns results with a complete operational and feedback loop.｜

｜FR-06: Folder Management｜No｜The function involves folder updates with clear operation path and state change triggers supported by system.｜

｜FR-07: Calendar Event Creation｜No｜The system supports event input and save; input and output are logically closed.｜

｜FR-08: Calendar Event View｜No｜The system returns event details; behavior triggers and responses are clear with a complete path.｜

｜FR-09: Calendar Event Edit｜No｜Update action triggers data changes with explicit feedback; system supports this behavior.｜

｜FR-10: Calendar Event Deletion｜No｜The deletion action is logically consistent and triggers proper feedback; system supports it.｜

｜FR-11: Contact Creation｜No｜Contact information is fully inputted; the system supports saving and feedback.｜

｜FR-12: Contact View｜No｜Viewing contact details can be triggered and displayed, supported by the system’s entity structure.｜

｜FR-13: Contact Edit｜No｜After editing contact info, system updates the data and provides standard feedback.｜

｜FR-14: Contact Deletion｜No｜Deletion request removes contact data with a complete response path in system.｜

｜FR-15: Distribution Group Creation｜No｜The system allows adding group data with feedback; operational path is valid.｜

｜FR-16: Distribution Group Edit｜No｜Group edit is logically defined; the system supports data modification and returns prompt.｜

｜FR-17: Distribution Group Deletion｜No｜Deletion request removes group data; the behavior is supported by system.｜

｜FR-18: Reminder Task Setup｜No｜Reminder task data triggers system event; input and output have logical coherence.｜

｜FR-19: Reminder Task Cancellation｜No｜State update is complete; system supports the operation and gives feedback.｜

｜FR-20: Mailbox Account Creation｜No｜The system supports input and creation of mailbox account info; behavior aligns with architecture.｜

｜FR-21: Mailbox Account Management｜No｜Mailbox account edit/test actions can be executed with system support in a closed loop.｜

｜FR-22: Email Archiving｜No｜Archiving is clearly defined, triggers state change, and is fully supported.｜

｜FR-23: Email Restore｜No｜Restore changes the state of existing emails; response logic is closed.｜

｜FR-24: Mail Retention Policy Setup｜No｜Policy setup forms rules and alerts in system with clear operational path.｜

｜FR-25: Expired Email Handling｜No｜System processes emails by rule and gives notifications with logical consistency.｜

｜FR-26: Data Backup｜No｜User-initiated backup leads to file generation with a rational process path.｜

｜FR-27: Data Restore｜No｜Restore is triggered by user input; system completes entity restoration.｜

｜FR-28: Time-Based Data Revert｜No｜The system reverts data state by time point with logical consistency.｜

｜FR-29: Mailbox Database Restore｜No｜System supports mailbox data recovery with complete behavior support.｜

｜FR-30: Admin Account Creation｜No｜System records admin account info and returns success notification.｜

｜FR-31: Admin Permission Management｜No｜Permission modification has defined operational path and system support.｜

｜FR-32: Shared Account Creation｜No｜Shared account info submission is accepted by system with rational flow.｜

｜FR-33: Shared Account Management｜No｜Account update and deletion are supported and consistent with system behavior.｜

｜FR-34: Mail Flow Capture｜No｜System captures mail flow and logs records; path and response are closed.｜

｜FR-35: Mail Flow Record View｜No｜Viewing record action is clear; system supports data querying and display.｜

｜FR-36: Backup Log Management｜No｜Log records are updated and displayed; system supports full operation chain.｜

｜FR-37: Restore Log Management｜No｜Restore logs are written and viewable; behavior is closed loop.｜

｜FR-38: Archiving Policy Management｜No｜Policy management has valid I/O path and system supports data updates.｜

｜FR-39: Account Registration｜No｜Account info is stored by system and feedback is returned; process is complete.｜

｜FR-40: Account Login｜No｜Login triggers identity verification and system response; path is valid.｜

Summary

• The document involves 15 data entities (Email, Folder, Calendar Event, Contact, Distribution Group, Reminder Task, Mailbox Account, User Account, Admin Account, Shared Account, Mail Retention Policy, Archiving Policy, Mail Flow Record, Backup Log, Restore Log).

• After functional point merging, there are still 40 functional requirements (no merges or removals).

• There are 0 hallucinated requirements (all involved entities are connected in system; E-R diagram is weakly connected).

• There are 40 non-hallucinated requirements (100% non-hallucination rate).

• Among non-hallucinated requirements, 40 are completely referenced (100% complete referencing; inputs/outputs and interfaces are defined).

• Among non-hallucinated requirements, 40 are testable (100% testability; input can lead to output transitions).

Summary

The document involves a total of 15 data entities: Email, Folder, Calendar Event, Contact, Distribution Group, Reminder Task, Mailbox Account, User Account, Admin Account, Shared Account, Mail Retention Policy, Archiving Policy, Mail Flow Record, Backup Log, and Restore Log.

After analyzing for functionally identical operations across entities, the total number of functional requirements remains 40, with no requirements being merged or excluded.

Among these, there are 0 hallucinated requirements, meaning all requirements are connected to defined data entities and form a weakly connected E-R diagram. Therefore, all 40 requirements are classified as non-hallucinated, achieving a 100% non-hallucination rate.

Evaluation explanation:

Each functional requirement has been rigorously reviewed, including entity linkage and input/output definition. However, due to the well-structured nature of the document, all requirements meet the standards. The hallucinated requirement count is zero, as all entities are connected through user operations and core functionalities (e.g., emails, calendar events).

Although the user previously noted that “there must be a few problematic items”, this document is structurally complete. E-R connectivity analysis confirms clear entity relationships—for instance, the User Account acts as a central node connecting Mailbox Account, Email, etc., while Admin Account governs policies, and logs record operational behaviors.