Here's a professional email draft summarizing the MRV Definitions Update process and attaching the functional requirements document:

**Subject:** Functional Requirements Document for MRV Definitions Update Process

**Hi [Manager's Name],**

As discussed, please find attached the functional requirements document outlining the MRV Definitions Update process for the Staging ARC Risk Model.

The goal of this initiative is to introduce a streamlined and efficient way to validate MRV changes in a controlled staging environment before deploying them to production. This approach helps reduce the risk associated with updates, ensures accuracy, and facilitates business review and reconciliation without interfering with the main production model.

### **Summary of the Process**

1. **Staging Environment Introduction**
2. The Staging ARC Model allows users to test newly defined MRV measures and validate their impact against real data without requiring a production release. This significantly reduces development cycle time and provides business users the opportunity to assess accuracy and relevance.
3. **Architecture Overview**

The high-level design includes:

* 1. A copy of production metadata into staging tables via the ARC-JDBC-Loader from the on-prem Oracle control.
  2. A separate MRV Builder in staging that constructs the ARC Risk Model (Staging) using the copied definitions.
  3. A reconciliation step where exposure data is compared to show the impact of changes.

1. **Definition Lifecycle**
   1. Updates are created and uploaded through a dedicated MRV Definition API.
   2. Users perform validation and reconciliation within the staging environment.
   3. Once validated and approved, definitions are released into production every Friday (COB), automating the final release process and updating the live schema accordingly.
2. **Functional Requirements Highlights**
   1. Automated metadata load from PROD to staging multiple times per day.
   2. Execution of the MRV Builder in staging.
   3. Automated reconciliation process post-build.
   4. User interface for MRV Definition approval with version tracking.
   5. Automated production release with schema update and cleanup of previous versions.

Please review the attached document for detailed workflows, diagrams, and implementation steps. Let me know if you’d like to set up a walkthrough session or need any clarifications.

Best regards,

**[Your Name]**

**[Your Position]**

**[Your Contact Info]**

Let me know if you'd like a version tailored for a presentation or meeting invite as well.