项目文档

# Functional Requirement

# Chapter 1: Functional Requirements  
  
## 1.1 Email Creation Function   
\*\*Function ID:\*\* FR-01   
\*\*Description:\*\* Administrators can create new emails using a dedicated form. The email must be associated with an Email Account or Shared Account.   
\*\*Input:\*\* Subject, body, recipient(s), sender (Email Account or Shared Account).   
\*\*Output:\*\* A new Email entity saved in the system with the associated Email Account or Shared Account.  
  
## 1.2 Email Sending Function   
\*\*Function ID:\*\* FR-02   
\*\*Description:\*\* Administrators can send an email to specified recipients using the selected Email Account or Shared Account.   
\*\*Input:\*\* Selected Email entity, Email Account or Shared Account, configured Email Flow and Policy rules.   
\*\*Output:\*\* The email is sent to recipients, and the Email entity status is updated to "Sent".  
  
## 1.3 Email Receiving Function   
\*\*Function ID:\*\* FR-03   
\*\*Description:\*\* The system receives and processes incoming emails according to Email Flow and Policy configurations, and stores them in the appropriate folder (e.g., inbox, spam, archive).   
\*\*Input:\*\* Incoming email via Email Flow, associated Email Account or Shared Account.   
\*\*Output:\*\* A new Email entity stored in the system, optionally archived or flagged as Expired Email.  
  
## 1.4 Email Viewing Function   
\*\*Function ID:\*\* FR-04   
\*\*Description:\*\* Administrators can view the content of a selected email.   
\*\*Input:\*\* Selected Email entity from inbox or other folders.   
\*\*Output:\*\* Display of the Email content, including subject, body, sender, and recipient information.  
  
## 1.5 Email Reply Function   
\*\*Function ID:\*\* FR-05   
\*\*Description:\*\* Administrators can create and send a reply email based on an existing email.   
\*\*Input:\*\* Selected Email entity, reply content, Email Account or Shared Account for sending.   
\*\*Output:\*\* A new Email entity saved as a reply, associated with the original email and ready to be sent.  
  
## 1.6 Email Forwarding Function   
\*\*Function ID:\*\* FR-06   
\*\*Description:\*\* Administrators can create and send a forwarded email based on an existing email.   
\*\*Input:\*\* Selected Email entity, new recipient(s), forwarding message content, Email Account or Shared Account for sending.   
\*\*Output:\*\* A new Email entity saved as a forward, associated with the original email and ready to be sent.  
  
## 1.7 Email Deletion Function   
\*\*Function ID:\*\* FR-07   
\*\*Description:\*\* Administrators can delete an email, either permanently or move it to the Archive or Expired Email folder, depending on system settings.   
\*\*Input:\*\* Selected Email entity, system settings for deletion behavior.   
\*\*Output:\*\* Email removed from its current folder and optionally moved to Archive or Expired Email folder.  
  
## 1.8 Email Archiving Function   
\*\*Function ID:\*\* FR-08   
\*\*Description:\*\* Administrators can archive an email to either a local Archive folder or a Server Archive, as per system settings.   
\*\*Input:\*\* Selected Email entity, system settings for archive destination.   
\*\*Output:\*\* Email moved to Archive or Server Archive folder and marked as archived.  
  
## 1.9 Email Expiry Setting Function   
\*\*Function ID:\*\* FR-09   
\*\*Description:\*\* Administrators can set an expiry date for an email. The system will automatically move it to the Expired Email folder on the set date.   
\*\*Input:\*\* Selected Email entity, expiry date, system Policy for retention rules.   
\*\*Output:\*\* Email marked with an expiry date and moved to the Expired Email folder at expiry.  
  
## 1.10 Expired Email Management Function   
\*\*Function ID:\*\* FR-10   
\*\*Description:\*\* Administrators can manage expired emails by deleting them permanently or moving them to the Server Archive.   
\*\*Input:\*\* Selected Expired Email entity, action (e.g., delete or archive), system Policy for handling expired emails.   
\*\*Output:\*\* Expired Email entity either deleted or moved to the Server Archive.  
  
## 1.11 Email Recovery Function   
\*\*Function ID:\*\* FR-11   
\*\*Description:\*\* Administrators can recover an email from the Archive or Server Archive and restore it to the inbox or a designated folder.   
\*\*Input:\*\* Selected Archive or Server Archive email, RecoveryRequest entity.   
\*\*Output:\*\* Email restored to the user's inbox or a designated folder, and its status updated.  
  
## 1.12 Distribution Group Management Function   
\*\*Function ID:\*\* FR-12   
\*\*Description:\*\* Administrators can create, update, or delete a Distribution Group and manage its members.   
\*\*Input:\*\* Distribution Group details (name, description, members), associated Email Account or Shared Account.   
\*\*Output:\*\* Updated Distribution Group entity in the system, with associated members and accounts.  
  
## 1.13 Distribution Group Member Addition Function   
\*\*Function ID:\*\* FR-13   
\*\*Description:\*\* Administrators can add one or more members (Contacts or Email Accounts) to a Distribution Group.   
\*\*Input:\*\* Distribution Group entity, list of Contacts or Email Accounts to add.   
\*\*Output:\*\* Updated Distribution Group with new members added.  
  
## 1.14 Distribution Group Member Removal Function   
\*\*Function ID:\*\* FR-14   
\*\*Description:\*\* Administrators can remove one or more members from a Distribution Group.   
\*\*Input:\*\* Distribution Group entity, list of Contacts or Email Accounts to remove.   
\*\*Output:\*\* Updated Distribution Group with members removed.  
  
## 1.15 Email Account Management Function   
\*\*Function ID:\*\* FR-15   
\*\*Description:\*\* Administrators can create, update, or delete Email Accounts.   
\*\*Input:\*\* Email Account details (username, password, domain, etc.), Policy for account creation rules.   
\*\*Output:\*\* Updated list of Email Accounts, with new or modified entries.  
  
## 1.16 Shared Account Management Function   
\*\*Function ID:\*\* FR-16   
\*\*Description:\*\* Administrators can register, update, or delete Shared Accounts and manage their access permissions.   
\*\*Input:\*\* Shared Account details, associated Email Accounts, Policy for access control.   
\*\*Output:\*\* Updated Shared Account entity in the system.  
  
## 1.17 Calendar Creation Function   
\*\*Function ID:\*\* FR-17   
\*\*Description:\*\* Administrators can create a new calendar and optionally associate it with a Contact or Email.   
\*\*Input:\*\* Calendar name, description, associated Contact or Email Account.   
\*\*Output:\*\* New Calendar entity stored in the system.  
  
## 1.18 Calendar Update Function   
\*\*Function ID:\*\* FR-18   
\*\*Description:\*\* Administrators can update an existing calendar's name, description, or associated entities.   
\*\*Input:\*\* Selected Calendar entity, updated details.   
\*\*Output:\*\* Updated Calendar entity in the system.  
  
## 1.19 Calendar Deletion Function   
\*\*Function ID:\*\* FR-19   
\*\*Description:\*\* Administrators can delete a calendar, with associated events either deleted or moved based on system settings.   
\*\*Input:\*\* Selected Calendar entity, system settings for event handling.   
\*\*Output:\*\* Calendar removed from the system and related events processed.  
  
## 1.20 Event Scheduling Function   
\*\*Function ID:\*\* FR-20   
\*\*Description:\*\* Administrators can schedule a new event and optionally send an email invitation to attendees.   
\*\*Input:\*\* Event details (title, date, time, location, attendees), Email Account or Shared Account for sending invitations.   
\*\*Output:\*\* New Event entity created and associated with the Calendar and Email.  
  
## 1.21 Event Modification Function   
\*\*Function ID:\*\* FR-21   
\*\*Description:\*\* Administrators can modify the details of an existing event and update any related Email or Contact associations.   
\*\*Input:\*\* Selected Event entity, updated details.   
\*\*Output:\*\* Updated Event entity in the system with associated data modified.  
  
## 1.22 Event Cancellation Function   
\*\*Function ID:\*\* FR-22   
\*\*Description:\*\* Administrators can cancel an event and optionally send a cancellation notification via Email.   
\*\*Input:\*\* Selected Event entity, Email Account or Shared Account for sending notification.   
\*\*Output:\*\* Event marked as canceled, and cancellation email sent.  
  
## 1.23 Event Detail Viewing Function   
\*\*Function ID:\*\* FR-23   
\*\*Description:\*\* Administrators can view the details of an event, including associated Contact and Email data.   
\*\*Input:\*\* Selected Event entity.   
\*\*Output:\*\* Display of event details including title, time, location, and related data.  
  
## 1.24 Contact Creation Function   
\*\*Function ID:\*\* FR-24   
\*\*Description:\*\* Administrators can create a new Contact and optionally associate it with an Email Account or Shared Account.   
\*\*Input:\*\* Contact details (name, email, phone, etc.), associated Email Account or Shared Account.   
\*\*Output:\*\* New Contact entity stored in the system.  
  
## 1.25 Contact Update Function   
\*\*Function ID:\*\* FR-25   
\*\*Description:\*\* Administrators can update an existing Contact's information and its associations.   
\*\*Input:\*\* Selected Contact entity, updated details.   
\*\*Output:\*\* Updated Contact entity with modified information.  
  
## 1.26 Contact Deletion Function   
\*\*Function ID:\*\* FR-26   
\*\*Description:\*\* Administrators can delete a Contact, with any associated Email or Shared Account links removed or updated.   
\*\*Input:\*\* Selected Contact entity, system settings for dependency handling.   
\*\*Output:\*\* Contact removed from the system and related data processed.  
  
## 1.27 Contact Viewing Function   
\*\*Function ID:\*\* FR-27   
\*\*Description:\*\* Administrators can view the details of a selected Contact, including associated Email or Shared Account information.   
\*\*Input:\*\* Selected Contact entity.   
\*\*Output:\*\* Display of Contact details including name, email, phone, and related data.  
  
## 1.28 Contact Group Management Function   
\*\*Function ID:\*\* FR-28   
\*\*Description:\*\* Administrators can manage Contact Groups by creating, updating, or deleting them and modifying their members.   
\*\*Input:\*\* Contact Group details, list of Contacts to add or remove.   
\*\*Output:\*\* Updated Contact Group entity in the system with new members or details.  
  
## 1.29 Archive Search Function   
\*\*Function ID:\*\* FR-29   
\*\*Description:\*\* Administrators can search for archived emails in the Archive or Server Archive using filters or search terms.   
\*\*Input:\*\* Search query, filters (e.g., date, sender, subject).   
\*\*Output:\*\* Retrieved and displayed archived Email entities matching the query.  
  
## 1.30 Archive Policy Configuration Function   
\*\*Function ID:\*\* FR-30   
\*\*Description:\*\* Administrators can create, update, or delete Archive Policies, which define how emails should be archived based on rules.   
\*\*Input:\*\* Archive Policy details (name, retention period, archive destination, expiry action).   
\*\*Output:\*\* Updated Archive Policy entity in the system.  
  
## 1.31 Server Archive Configuration Function   
\*\*Function ID:\*\* FR-31   
\*\*Description:\*\* Administrators can configure the Server Archive, including associating it with Email Accounts or Shared Accounts and setting its compliance status.   
\*\*Input:\*\* Server Archive settings, associated Email Accounts or Shared Accounts.   
\*\*Output:\*\* Updated Server Archive entity with new configuration applied.  
  
## 1.32 Email Flow Capture Function   
\*\*Function ID:\*\* FR-32   
\*\*Description:\*\* The system captures and records email flows (incoming, outgoing, internal) associated with Email Accounts or Shared Accounts.   
\*\*Input:\*\* Email communication data, Email Account or Shared Account.   
\*\*Output:\*\* New EmailFlow entity stored in the system.  
  
## 1.33 Compliance Monitoring Function   
\*\*Function ID:\*\* FR-33   
\*\*Description:\*\* The system scans and evaluates emails against configured compliance policies to detect violations.   
\*\*Input:\*\* Email entities, Policy rules for compliance.   
\*\*Output:\*\* ComplianceReport entity with details of compliance violations.  
  
## 1.34 Compliance Report Management Function   
\*\*Function ID:\*\* FR-34   
\*\*Description:\*\* Administrators can view, modify, or delete Compliance Reports.   
\*\*Input:\*\* Compliance Report entity, modifications or deletion confirmation.   
\*\*Output:\*\* Updated or deleted Compliance Report entity.  
  
## 1.35 Recovery Request Processing Function   
\*\*Function ID:\*\* FR-35   
\*\*Description:\*\* Administrators can approve, deny, or delete Recovery Requests for restoring archived emails.   
\*\*Input:\*\* RecoveryRequest entity, action (approve, deny, delete).   
\*\*Output:\*\* Updated RecoveryRequest entity and restored email if approved.  
  
## 1.36 User Management Function   
\*\*Function ID:\*\* FR-36   
\*\*Description:\*\* Administrators can create, update, or delete user accounts and manage their roles and permissions.   
\*\*Input:\*\* User details, role, permissions, associated Email Accounts or Shared Accounts.   
\*\*Output:\*\* Updated User entity in the system.  
  
## 1.37 Administrator Management Function   
\*\*Function ID:\*\* FR-37   
\*\*Description:\*\* Administrators can manage other administrators by creating, updating, or deleting their roles and permissions.   
\*\*Input:\*\* Admin details, role, permissions, associated Email Accounts or Shared Accounts.   
\*\*Output:\*\* Updated Administrator entity in the system.  
  
## 1.38 Email Sharing Function   
\*\*Function ID:\*\* FR-38   
\*\*Description:\*\* Administrators can share an email with other users or groups and record the action.   
\*\*Input:\*\* Selected Email entity, recipient(s), sharing method (e.g., via email or calendar event).   
\*\*Output:\*\* Email marked as shared, and sharing event logged in the system.  
  
## 1.39 Email Expiry Handling Function   
\*\*Function ID:\*\* FR-39   
\*\*Description:\*\* The system automatically processes emails that have reached their expiry date, moving them to the Expired Email folder or deleting them.   
\*\*Input:\*\* Email with expiry date set, system settings for expiry handling.   
\*\*Output:\*\* Email moved to Expired Email folder or deleted, with logs updated.  
  
## 1.40 Event Record Management Function   
\*\*Function ID:\*\* FR-40   
\*\*Description:\*\* Administrators can manage Event Records by creating, updating, or deleting them.   
\*\*Input:\*\* EventRecord details, associated Calendar, Email, or Contact.   
\*\*Output:\*\* Updated or deleted EventRecord entity in the system.

# External Description

# Chapter 2: External Interfaces  
  
## 2.1 User Interface Output  
  
The system interacts with administrators through a dedicated graphical user interface (GUI) and/or command-line interface (CLI), allowing for the creation, viewing, modification, and deletion of entities such as emails, contacts, calendars, events, distribution groups, and recovery requests. The following user interface elements are defined for system-administrator interaction:  
  
- \*\*Email Creation and Management UI:\*\*   
 A form-based interface for administrators to create, send, view, reply, forward, delete, archive, or set expiry for emails. Input fields include subject, body, recipients, sender (Email Account or Shared Account), and expiry date. Output includes the saved Email entity and status updates (e.g., "Sent", "Archived", "Expired").  
  
- \*\*Distribution Group Management UI:\*\*   
 A user interface for creating, updating, or deleting Distribution Groups. It allows the selection and management of members (Contacts or Email Accounts) and is linked to the Email Account or Shared Account associated with the group.  
  
- \*\*Calendar and Event Management UI:\*\*   
 A dedicated UI for creating, updating, or deleting calendars and scheduling, modifying, or canceling events. Event scheduling can optionally be linked to the sending of email invitations.  
  
- \*\*Contact Management UI:\*\*   
 A user interface for creating, updating, or deleting contacts. It also allows viewing contact details and managing their associations with Email Accounts or Shared Accounts.  
  
- \*\*Archive and Expired Email Management UI:\*\*   
 An interface for searching archived emails, managing Archive Policies, and processing expired emails according to system settings. It supports filters (e.g., date, sender, subject) and actions (e.g., delete, restore, move to Server Archive).  
  
- \*\*Compliance Report UI:\*\*   
 A dedicated interface for viewing and managing Compliance Reports. It allows administrators to review violations detected by the Compliance Monitoring Function and take appropriate actions.  
  
- \*\*User and Administrator Management UI:\*\*   
 A UI for managing user accounts and administrator roles and permissions. It supports actions such as creating, updating, or deleting users and administrators, as well as associating them with Email Accounts or Shared Accounts.  
  
- \*\*Email Sharing UI:\*\*   
 A user interface for sharing selected emails with other users or groups. It records the sharing action and logs the event in the system.  
  
## 2.2 Hardware Interface Output  
  
This system does not have direct hardware dependencies. However, it may be deployed on server hardware that supports enterprise-level email and data management operations. The following hardware-related interfaces are relevant:  
  
- \*\*Server Infrastructure:\*\*   
 The system requires a reliable server environment for hosting the application and database. The hardware must support the required software stack (e.g., operating system, web server, application server, and database engine). Server hardware should be scalable to handle large volumes of email data and concurrent administrative actions.  
  
- \*\*Storage Devices:\*\*   
 The system may interact with external or internal storage devices for storing archived emails (local Archive folder or Server Archive). These devices must be compatible with the system's file storage and retrieval mechanisms.  
  
## 2.3 Software Interface Output  
  
The system interacts with various external software components to fulfill functional requirements. These include databases, third-party tools, and APIs for email flow, compliance monitoring, and policy enforcement.  
  
### 2.3.1 Database Interfaces  
  
- \*\*Email Account and Shared Account Database:\*\*   
 Stores information about Email Accounts and Shared Accounts, including credentials, domain, and access permissions. The system reads and writes to this database to manage account creation, updates, and deletion.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations for data retrieval and modification.   
 \*Input/Output:\*   
 Input includes account details (username, password, domain, etc.). Output includes updated account entries and status changes.  
  
- \*\*Email Entity Database:\*\*   
 Stores the content and metadata of all emails, including subject, body, sender, recipient, status, and expiry date. The system writes to this database when emails are created, sent, received, archived, or expired.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations.   
 \*Input/Output:\*   
 Input includes the email content and associated account. Output includes the saved Email entity and status updates.  
  
- \*\*Archive and Server Archive Database:\*\*   
 Stores archived emails and expired emails. The system retrieves emails from this database when performing archive search or email recovery.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations.   
 \*Input/Output:\*   
 Input includes search queries and filters. Output includes retrieved Email entities and their status.  
  
- \*\*Event and Calendar Database:\*\*   
 Stores Calendar and Event entities, along with their associations to Email Accounts, Shared Accounts, and Contacts.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations.   
 \*Input/Output:\*   
 Input includes event details (title, date, time, location, attendees). Output includes the updated Calendar and Event entities.  
  
- \*\*Contact and Contact Group Database:\*\*   
 Stores Contact and Contact Group entities, including contact details and group memberships.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations.   
 \*Input/Output:\*   
 Input includes contact or group details. Output includes the updated Contact or Contact Group entities.  
  
- \*\*Compliance Policy and Report Database:\*\*   
 Stores compliance policies and Compliance Reports. The system evaluates emails against these policies and logs violations.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations.   
 \*Input/Output:\*   
 Input includes Email entities and configured compliance policies. Output includes Compliance Reports detailing violations.  
  
- \*\*Recovery Request Database:\*\*   
 Stores RecoveryRequest entities, which are used to manage the restoration of archived emails.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations.   
 \*Input/Output:\*   
 Input includes RecoveryRequest entity and action (approve, deny, delete). Output includes the updated RecoveryRequest and restored Email entity if approved.  
  
- \*\*User and Administrator Database:\*\*   
 Stores user and administrator account details, roles, and permissions.   
 \*Interaction Method:\*   
 SQL queries or ORM-based operations.   
 \*Input/Output:\*   
 Input includes user details and permissions. Output includes updated User or Administrator entities.  
  
### 2.3.2 Third-Party Software Interfaces  
  
- \*\*Email Flow and Policy Engine API:\*\*   
 An external API or service used to capture and process email flows (incoming, outgoing, internal). It also applies configured policies to determine how emails should be handled (e.g., sent, archived, expired).   
 \*Interaction Method:\*   
 RESTful API calls or direct integration with a policy enforcement system.   
 \*Input/Output:\*   
 Input includes Email communication data and associated Email Account or Shared Account. Output includes the new EmailFlow entity and policy-based status updates.  
  
- \*\*Compliance Monitoring Service API:\*\*   
 An external service or API that evaluates emails against configured compliance rules.   
 \*Interaction Method:\*   
 RESTful API calls or integration with a compliance engine.   
 \*Input/Output:\*   
 Input includes Email entities and compliance policy rules. Output includes ComplianceReport entities with violation details.  
  
- \*\*Email Sending Service API (e.g., SMTP, Email Gateway):\*\*   
 Used to send emails to recipients. The system integrates with this service to execute the sending process.   
 \*Interaction Method:\*   
 SMTP protocol or RESTful API for email dispatch.   
 \*Input/Output:\*   
 Input includes the Email entity and sending account. Output includes confirmation that the email has been sent and the Email entity status updated to "Sent".  
  
- \*\*Email Receiving Service API (e.g., POP3, IMAP, Email Gateway):\*\*   
 Used to receive and process incoming emails. The system uses this service to capture emails and store them in the appropriate folder (e.g., inbox, spam, archive).   
 \*Interaction Method:\*   
 POP3 or IMAP protocol for receiving emails, or RESTful API for integration with an external email gateway.   
 \*Input/Output:\*   
 Input includes incoming email data and associated Email Account or Shared Account. Output includes a new Email entity in the system.  
  
- \*\*Archive Policy Engine API:\*\*   
 Used to create, update, or delete Archive Policies and to enforce archival rules based on retention periods and other criteria.   
 \*Interaction Method:\*   
 RESTful API or internal module for policy configuration.   
 \*Input/Output:\*   
 Input includes Archive Policy details (name, retention period, archive destination, expiry action). Output includes the updated Archive Policy entity.  
  
- \*\*Server Archive Configuration Module:\*\*   
 A software component for configuring the Server Archive, including associating it with Email Accounts or Shared Accounts and setting its compliance status.   
 \*Interaction Method:\*   
 Internal module or external API for archive configuration.   
 \*Input/Output:\*   
 Input includes Server Archive settings and associated accounts. Output includes the updated Server Archive entity.  
  
## 2.4 Communication Interface Output  
  
The system communicates with external systems and users through various communication protocols and channels, primarily for email handling and notifications.  
  
- \*\*Email Communication (SMTP/POP3/IMAP):\*\*   
 The system sends and receives emails using standard email protocols (SMTP for sending, POP3/IMAP for receiving).   
 \*Interaction Method:\*   
 SMTP, POP3, or IMAP.   
 \*Input/Output:\*   
 Input includes recipient email addresses and message content. Output includes sent emails and received emails processed into the system.  
  
- \*\*Email Invitation and Notification Sending:\*\*   
 The system sends email invitations for events and cancellation notifications.   
 \*Interaction Method:\*   
 SMTP or external email sending service.   
 \*Input/Output:\*   
 Input includes event details and sending account. Output includes the email invitation or cancellation message sent to attendees.  
  
- \*\*Compliance Violation Notification (Email or In-System Alert):\*\*   
 When compliance violations are detected, the system may notify administrators via email or an in-system alert.   
 \*Interaction Method:\*   
 Email notification (SMTP) or internal alert system.   
 \*Input/Output:\*   
 Input includes ComplianceReport entity. Output includes the notification sent to the relevant administrator.  
  
- \*\*Recovery Request Notification (Email or In-System Alert):\*\*   
 Administrators may receive notifications about new Recovery Requests via email or within the system.   
 \*Interaction Method:\*   
 Email notification (SMTP) or internal alert system.   
 \*Input/Output:\*   
 Input includes RecoveryRequest entity. Output includes the notification sent to the administrator for review.  
  
- \*\*Email Sharing Notification:\*\*   
 When an email is shared with another user or group, a notification is sent to inform the recipient(s).   
 \*Interaction Method:\*   
 Email notification (SMTP) or internal notification system.   
 \*Input/Output:\*   
 Input includes the Email entity and sharing recipients. Output includes the sharing notification sent to the recipient(s).  
  
- \*\*Event Reminder Communication:\*\*   
 The system may send reminders for upcoming events via email or calendar integration.   
 \*Interaction Method:\*   
 Email notification (SMTP) or calendar integration (e.g., iCalendar).   
 \*Input/Output:\*   
 Input includes event details and reminder settings. Output includes the reminder notification sent to attendees.  
  
- \*\*System-to-System Communication (for Synchronization or Backup):\*\*   
 The system may communicate with external backup or synchronization services to ensure data integrity and availability.   
 \*Interaction Method:\*   
 RESTful API or scheduled data transfer protocols.   
 \*Input/Output:\*   
 Input includes data for synchronization or backup. Output includes confirmation of successful synchronization or backup.  
  
---  
  
This chapter provides a comprehensive overview of the external interfaces required for the system to function as described in the functional requirements. It ensures that developers and integrators understand the necessary interactions with user interfaces, hardware, software components, and communication protocols.

# Use Case

Use Case Name: Create Email   
Use Case ID: UC-01   
Actors: Administrator, Email Account, Shared Account   
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has access to an Email Account or a Shared Account.   
3. The system has a Calendar, Contact, and Email entities available for integration.   
  
Postconditions:   
1. A new email is created and saved in the system.   
2. The email is associated with the selected Email Account or Shared Account.   
3. The email is displayed in the user's email list.   
  
Main Flow:   
1. The Administrator selects the option to "Create Email" from the system's menu.   
2. The system displays a form for creating a new email.   
3. The Administrator fills in the subject, body, and recipient information for the email.   
4. The Administrator selects the Email Account or Shared Account to be used for sending the email.   
5. The Administrator clicks the "Save" button to create the email.   
6. The system creates the email and associates it with the selected Email Account or Shared Account.   
7. The system displays a confirmation message indicating that the email has been successfully created.   
  
Alternative Flow:   
1. If the Administrator does not select an Email Account or Shared Account, the system displays an error message and prompts the Administrator to select one.   
2. If the Administrator enters invalid recipient information, the system displays an error message and prompts the Administrator to correct it.   
3. If the Administrator cancels the email creation process, the system returns to the previous screen without saving the email.  
  
Use Case Name: Send Email   
Use Case ID: UC-02   
Actors: Administrator, Email Account, Shared Account, Email Flow, Policy   
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has created an email using the "Create Email" use case.   
3. The email is associated with an Email Account or a Shared Account.   
4. The system has an Email Flow and Policy entities configured for email sending.   
  
Postconditions:   
1. The email is sent to the specified recipients.   
2. The email is marked as sent in the system.   
3. The email is removed from the user's draft list and moved to the sent folder or archive, depending on the system settings.   
4. The system logs the sending event for auditing purposes.   
  
Main Flow:   
1. The Administrator selects the email from the email list.   
2. The Administrator clicks the "Send" button.   
3. The system checks the associated Email Account or Shared Account for sending permissions.   
4. The system validates the email content and recipient information against the Email Flow and Policy rules.   
5. The system sends the email to the recipients.   
6. The system updates the email status to "Sent".   
7. The system displays a confirmation message indicating that the email has been successfully sent.   
  
Alternative Flow:   
1. If the system detects a violation of the Email Flow or Policy rules, it displays an error message and prompts the Administrator to adjust the email content or recipient information.   
2. If the selected Email Account or Shared Account is not authorized to send emails, the system displays an error message and prevents the sending process.   
3. If the email sending process fails due to network issues or other technical problems, the system displays an error message and logs the failure. The Administrator is prompted to retry or cancel the sending.   
4. If the Administrator cancels the sending process, the system returns to the email preview screen without sending the email.  
  
Use Case Name: Receive Email   
Use Case ID: UC-03   
Actors: Administrator, Email Account, Shared Account, Calendar, Contact, Email Flow, Policy, Server Archive, Email, Expired Email   
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has access to an Email Account or a Shared Account for receiving emails.   
3. The system is configured to monitor incoming emails via the Email Flow and Policy entities.   
4. The system has a Calendar and Contact entities for integration with received emails.   
  
Postconditions:   
1. The incoming email is received and stored in the system.   
2. The email is associated with the corresponding Email Account or Shared Account.   
3. The email is displayed in the Administrator's inbox or relevant folder.   
4. If applicable, the email is archived to the Server Archive or marked as Expired Email based on system policies.   
  
Main Flow:   
1. The system detects an incoming email via the Email Flow and Policy configurations.   
2. The system authenticates the email and verifies it against the Email Account or Shared Account.   
3. The system retrieves the email content, subject, sender, and recipient information.   
4. The system saves the email in the appropriate folder (e.g., inbox, spam, or archive).   
5. The system notifies the Administrator of the new email via the system interface.   
6. The Administrator reviews the email and may choose to move it to the Calendar or Contact for further integration.   
7. The system logs the receipt of the email for auditing and tracking purposes.   
  
Alternative Flow:   
1. If the incoming email fails authentication, the system rejects it and logs the failure.   
2. If the system cannot determine the associated Email Account or Shared Account, the email is placed in a default or unassigned folder for review.   
3. If the email is flagged by the Policy as suspicious or spam, the system may automatically move it to the spam folder or notify the Administrator for manual review.   
4. If the system is unable to save the email due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the email is expired or no longer relevant, the system may automatically move it to the Expired Email or Server Archive folder.  
  
Use Case Name: View Email   
Use Case ID: UC-04   
Actors: Administrator, Email Account, Shared Account, Email, Calendar, Contact   
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has an email in the inbox or another folder that needs to be viewed.   
3. The email is associated with an Email Account or a Shared Account.   
  
Postconditions:   
1. The selected email is displayed to the Administrator.   
2. The Administrator can view the email's subject, body, sender, and recipient information.   
3. The email remains in its original folder unless moved or archived by the Administrator.   
  
Main Flow:   
1. The Administrator navigates to the email list or inbox.   
2. The Administrator selects an email from the list.   
3. The system retrieves the selected email from the database.   
4. The system displays the email's content, including the subject, body, sender, and recipient information.   
5. The Administrator reviews the email.   
6. The system logs the viewing event for auditing and tracking purposes.   
  
Alternative Flow:   
1. If the selected email cannot be retrieved due to technical issues, the system displays an error message and logs the failure.   
2. If the email is an Expired Email, the system displays a warning message and may prompt the Administrator to archive it.   
3. If the Administrator chooses to integrate the email with the Calendar or Contact, the system provides an option to add event details or contact information accordingly.   
4. If the Administrator decides to archive the email, the system moves it to the Server Archive or Archive folder based on the system settings.  
  
Use Case Name: Reply to Email   
Use Case ID: UC-05   
Actors: Administrator, Email Account, Shared Account, Email, Calendar, Contact, Email Flow, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has viewed an email using the "View Email" use case.   
3. The email is associated with an Email Account or a Shared Account.   
4. The system has an Email Flow and Policy entities configured for email sending and formatting.   
  
Postconditions:   
1. A reply email is created and associated with the original email.   
2. The reply email is saved in the system and displayed in the user's draft or outbox, depending on system settings.   
3. The email is ready to be sent or further edited.   
4. The system logs the reply action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator selects an email from the inbox or another folder.   
2. The Administrator clicks the "Reply" button.   
3. The system generates a reply email with the original email included as a reference.   
4. The system displays a reply form with the subject and body pre-filled for convenience.   
5. The Administrator edits the reply message as needed.   
6. The Administrator selects the Email Account or Shared Account to be used for sending the reply.   
7. The Administrator clicks the "Save" or "Send" button.   
8. The system saves the reply email as a draft or sends it immediately, depending on the selected action.   
9. The system displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the Administrator does not select an Email Account or Shared Account, the system displays an error message and prompts the Administrator to select one.   
2. If the system detects a violation of the Email Flow or Policy rules, it displays an error message and prompts the Administrator to adjust the reply content.   
3. If the reply email cannot be saved due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the reply process, the system returns to the email preview screen without saving or sending the reply.   
5. If the email is an Expired Email, the system may prompt the Administrator with a warning before allowing a reply.  
  
Use Case Name: Forward Email   
Use Case ID: UC-06   
Actors: Administrator, Email Account, Shared Account, Email, Email Flow, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has viewed an email using the "View Email" use case.   
3. The email is associated with an Email Account or a Shared Account.   
4. The system has an Email Flow and Policy entities configured for email sending and formatting.   
  
Postconditions:   
1. A forward email is created and associated with the original email.   
2. The forward email is saved in the system as a draft or sent, depending on the Administrator's action.   
3. The forward email is displayed in the user's draft or outbox.   
4. The system logs the forwarding action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator selects an email from the inbox or another folder.   
2. The Administrator clicks the "Forward" button.   
3. The system generates a forward email with the original email included as a reference.   
4. The system displays a forward form with the subject and body pre-filled for convenience.   
5. The Administrator adds new recipients and edits the message if needed.   
6. The Administrator selects the Email Account or Shared Account to be used for sending the forward email.   
7. The Administrator clicks the "Save" or "Send" button.   
8. The system saves the forward email as a draft or sends it immediately, depending on the selected action.   
9. The system displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the Administrator does not select an Email Account or Shared Account, the system displays an error message and prompts the Administrator to select one.   
2. If the system detects a violation of the Email Flow or Policy rules, it displays an error message and prompts the Administrator to adjust the forward content or recipient information.   
3. If the forward email cannot be saved due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the forward process, the system returns to the email preview screen without saving or sending the forward email.   
5. If the selected email is an Expired Email, the system may prompt the Administrator with a warning before allowing a forward.  
  
Use Case Name: Delete Email   
Use Case ID: UC-07   
Actors: Administrator, Email Account, Shared Account, Email, Archive, Expired Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has an email in the inbox, draft, or another folder that needs to be deleted.   
3. The email is associated with an Email Account or a Shared Account.   
4. The system has an Archive and Expired Email entities for managing deleted emails.   
  
Postconditions:   
1. The selected email is removed from its current folder.   
2. The email is either permanently deleted or moved to the Archive or Expired Email folder, depending on the system settings.   
3. The system updates the email list to reflect the deletion.   
4. The system logs the deletion action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the email list or inbox.   
2. The Administrator selects the email to be deleted.   
3. The Administrator clicks the "Delete" button.   
4. The system prompts the Administrator to confirm the deletion.   
5. The Administrator confirms the deletion.   
6. The system checks system settings to determine if the email should be archived or permanently deleted.   
7. The system removes the email from its current folder and moves it to the Archive or Expired Email folder, or deletes it permanently.   
8. The system updates the email list to reflect the change.   
9. The system displays a confirmation message indicating the email has been successfully deleted.   
  
Alternative Flow:   
1. If the Administrator cancels the deletion confirmation, the system returns to the email preview screen without taking any action.   
2. If the system is unable to move the email to the Archive or Expired Email folder, it displays an error message and logs the failure.   
3. If the email is already marked as Expired Email, the system may display a message indicating that it cannot be deleted or confirm the deletion regardless.   
4. If the deletion fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.  
  
Use Case Name: Archive Email   
Use Case ID: UC-08   
Actors: Administrator, Email Account, Shared Account, Archive, Server Archive, Email, Expired Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has viewed or selected an email that is eligible for archiving.   
3. The email is associated with an Email Account or a Shared Account.   
4. The system has Archive and Server Archive entities configured.   
  
Postconditions:   
1. The selected email is archived and removed from the current folder.   
2. The email is stored in the Archive or Server Archive folder, depending on system settings.   
3. The email list is updated to reflect the archiving action.   
4. The system logs the archiving event for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the email list or inbox.   
2. The Administrator selects an email to be archived.   
3. The Administrator clicks the "Archive" button.   
4. The system prompts the Administrator to confirm the archiving action.   
5. The Administrator confirms the archiving.   
6. The system checks the system settings to determine whether the email should be moved to the Archive folder or the Server Archive.   
7. The system moves the email to the designated Archive or Server Archive folder.   
8. The system updates the email list to reflect the change.   
9. The system displays a confirmation message indicating the email has been successfully archived.   
  
Alternative Flow:   
1. If the Administrator cancels the archiving confirmation, the system returns to the email preview screen without taking any action.   
2. If the system is unable to move the email to the Archive or Server Archive folder, it displays an error message and logs the failure.   
3. If the selected email is an Expired Email, the system may prompt the Administrator with a warning before allowing the archiving process.   
4. If the archiving fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.  
  
Use Case Name: Search Archive   
Use Case ID: UC-09   
Actors: Administrator, Archive, Server Archive, Email, Email Account, Shared Account, Expired Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains one or more emails stored in the Archive or Server Archive.   
3. The Administrator has the necessary permissions to access the Archive or Server Archive.   
4. The system supports search functionality for archived emails.   
  
Postconditions:   
1. The selected email is retrieved from the Archive or Server Archive.   
2. The email is displayed to the Administrator for viewing or further actions.   
3. The system logs the search action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Archive or Server Archive section of the system.   
2. The Administrator enters a search term or selects filters (e.g., date, sender, subject).   
3. The Administrator clicks the "Search" button.   
4. The system searches the Archive or Server Archive for matching emails.   
5. The system displays the search results to the Administrator.   
6. The Administrator selects an email from the search results.   
7. The system retrieves and displays the selected email.   
8. The system logs the search and retrieval actions.   
  
Alternative Flow:   
1. If no matching emails are found, the system displays a message indicating that no results were found.   
2. If the search fails due to technical issues, the system displays an error message and logs the failure.   
3. If the selected email is an Expired Email, the system may prompt the Administrator with a warning before displaying its content.   
4. If the Administrator cancels the search process, the system returns to the previous screen without displaying results.  
  
Use Case Name: Set Email Expiry   
Use Case ID: UC-10   
Actors: Administrator, Email, Expired Email, Policy, Email Account, Shared Account   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has selected an email that needs to have an expiry date set.   
3. The system has a Policy entity that defines email retention rules.   
4. The email is associated with an Email Account or a Shared Account.   
  
Postconditions:   
1. The selected email is marked with an expiry date.   
2. The system automatically moves the email to the Expired Email folder on the set expiry date.   
3. The email list is updated to reflect the expiry setting.   
4. The system logs the expiry setting for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the email list or inbox.   
2. The Administrator selects an email and clicks the "Set Expiry" option.   
3. The system displays a form to set the expiry date for the email.   
4. The Administrator enters the expiry date or selects it from the Calendar.   
5. The Administrator clicks the "Save" button.   
6. The system updates the email with the expiry date and associates it with the Expiry Email entity.   
7. The system displays a confirmation message indicating the expiry date has been successfully set.   
  
Alternative Flow:   
1. If the Administrator does not select a valid expiry date, the system displays an error message and prompts for correction.   
2. If the system detects a conflict with the Policy (e.g., expiry date is earlier than the minimum retention period), it displays a warning and prevents the expiry setting.   
3. If the selected email is already marked as Expired Email, the system displays a message indicating that it cannot be modified.   
4. If the system fails to save the expiry setting due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the expiry setting process, the system returns to the email preview screen without making any changes.  
  
Use Case Name: Manage Expired Email   
Use Case ID: UC-11   
Actors: Administrator, Email Account, Shared Account, Expired Email, Archive, Server Archive, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains one or more emails marked as Expired Email.   
3. The system has an Archive and Server Archive entities for handling expired emails.   
4. The system has a Policy entity configured to determine actions for expired emails.   
  
Postconditions:   
1. The expired email is either permanently deleted, moved to the Archive, or moved to the Server Archive, based on system settings or Policy rules.   
2. The email list is updated to reflect the removal of the expired email.   
3. The system logs the management action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Expired Email section of the system.   
2. The Administrator selects an expired email from the list.   
3. The Administrator chooses an action (e.g., delete, archive to local Archive, or move to Server Archive).   
4. The system prompts the Administrator to confirm the selected action.   
5. The Administrator confirms the action.   
6. The system performs the selected action on the expired email.   
7. The system updates the email list to reflect the change.   
8. The system displays a confirmation message indicating the expired email has been successfully managed.   
  
Alternative Flow:   
1. If the selected action cannot be performed due to Policy restrictions, the system displays an error message and prevents the action.   
2. If the system is unable to move the expired email to the Archive or Server Archive, it displays an error message and logs the failure.   
3. If the Administrator cancels the action confirmation, the system returns to the Expired Email list without making any changes.   
4. If the expired email is already in the Server Archive, the system may display a message indicating that it cannot be moved again.   
5. If the management process fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.  
  
Use Case Name: Recover Email   
Use Case ID: UC-12   
Actors: Administrator, Email Account, Shared Account, Archive, Server Archive, Email, Expired Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The email to be recovered is stored in the Archive, Server Archive, or marked as Expired Email.   
3. The Administrator has the necessary permissions to recover emails.   
4. The system has a recovery mechanism for emails from the Archive and Server Archive.   
  
Postconditions:   
1. The recovered email is restored to the Administrator's inbox or a designated folder.   
2. The email is no longer marked as Expired Email.   
3. The email list is updated to reflect the recovery.   
4. The system logs the recovery action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Archive or Server Archive section.   
2. The Administrator searches for the email to be recovered using filters or search terms.   
3. The Administrator selects the email from the search results.   
4. The Administrator clicks the "Recover" button.   
5. The system prompts the Administrator to confirm the recovery action.   
6. The Administrator confirms the recovery.   
7. The system restores the email to the inbox or another designated folder.   
8. The system updates the email status to remove the "Expired" flag if applicable.   
9. The system displays a confirmation message that the email has been successfully recovered.   
  
Alternative Flow:   
1. If the selected email is not eligible for recovery (e.g., already deleted permanently), the system displays an error message.   
2. If the system is unable to locate the email in the Archive or Server Archive, it displays an error message and logs the failure.   
3. If the Administrator cancels the recovery confirmation, the system returns to the Archive/Server Archive list without making any changes.   
4. If the system settings prevent recovery of certain types of emails, the system displays a warning and does not allow the action.   
5. If the recovery process fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.  
  
Use Case Name: Manage Distribution Group   
Use Case ID: UC-13   
Actors: Administrator, Distribution Group, Email Account, Shared Account, Email, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the Distribution Group entity configured.   
3. The Administrator has the necessary permissions to manage Distribution Groups.   
4. The system has access to an Email Account or Shared Account for email operations.   
  
Postconditions:   
1. The Distribution Group is either created, updated, or deleted, based on the Administrator's action.   
2. The system updates the group's associated Email Account or Shared Account if applicable.   
3. The system logs the management action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Distribution Group management section of the system.   
2. The Administrator selects an action (e.g., create, edit, or delete a Distribution Group).   
3. If creating, the Administrator enters the group name, adds members, and associates it with an Email Account or Shared Account.   
4. If editing, the Administrator selects a group and modifies its details (e.g., name, members, associated account).   
5. If deleting, the Administrator selects a group and confirms the deletion.   
6. The system validates the action against the Policy rules.   
7. The system updates the Distribution Group accordingly.   
8. The system displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the system detects a violation of the Policy rules (e.g., unauthorized group deletion), it displays an error message and prevents the action.   
2. If the Administrator does not provide a valid group name or member list, the system displays an error message and prompts for correction.   
3. If the selected group is already associated with an Email or cannot be deleted due to dependencies, the system displays a warning message.   
4. If the system fails to update the Distribution Group due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the operation, the system returns to the Distribution Group management screen without making any changes.  
  
Use Case Name: Register Shared Account   
Use Case ID: UC-14   
Actors: Administrator, Shared Account, Email Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system supports the registration of a Shared Account.   
3. The Administrator has the necessary permissions to create or register a Shared Account.   
4. The system has an Email Account and Policy entities configured.   
  
Postconditions:   
1. A new Shared Account is registered in the system.   
2. The Shared Account is associated with one or more Email Accounts as defined by the Administrator.   
3. The system updates the list of available Shared Accounts.   
4. The system logs the registration action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Shared Account management section of the system.   
2. The Administrator selects the "Register Shared Account" option.   
3. The system displays a form for entering the Shared Account details (e.g., name, description, associated Email Accounts).   
4. The Administrator fills in the required information, including selecting the associated Email Accounts.   
5. The Administrator clicks the "Register" button.   
6. The system validates the input data and checks for compliance with the Policy rules.   
7. The system registers the Shared Account and associates it with the selected Email Accounts.   
8. The system updates the Shared Account list and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not select any Email Accounts for the Shared Account, the system displays an error message and prompts for selection.   
2. If the system detects invalid or incomplete information in the registration form, it displays an error message and prompts the Administrator to correct it.   
3. If the system detects a violation of the Policy rules (e.g., unauthorized account creation), it displays an error message and prevents the registration.   
4. If the Shared Account registration fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the registration process, the system returns to the Shared Account management screen without making any changes.  
  
Use Case Name: Manage Email Account   
Use Case ID: UC-15   
Actors: Administrator, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the Email Account and Shared Account entities available.   
3. The Administrator has the necessary permissions to manage Email Accounts.   
4. The system has a Policy entity configured to define rules for account management.   
  
Postconditions:   
1. The Email Account is either created, updated, or deleted, based on the Administrator's action.   
2. The system updates the list of available Email Accounts to reflect the changes.   
3. The system logs the management action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Email Account management section of the system.   
2. The Administrator selects an action (e.g., create, edit, or delete an Email Account).   
3. If creating, the Administrator enters the email account details (e.g., username, password, server settings).   
4. If editing, the Administrator selects an existing Email Account and modifies its details.   
5. If deleting, the Administrator selects an Email Account and confirms the deletion.   
6. The system validates the input data and checks for compliance with the Policy rules.   
7. The system updates the Email Account accordingly.   
8. The system displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the system detects a violation of the Policy rules (e.g., unauthorized account deletion), it displays an error message and prevents the action.   
2. If the Administrator does not provide valid or complete information for creating or editing an Email Account, the system displays an error message and prompts for correction.   
3. If the selected Email Account is associated with active emails or workflows, the system displays a warning message before proceeding with deletion.   
4. If the system fails to update the Email Account due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the operation, the system returns to the Email Account management screen without making any changes.  
  
Use Case Name: Create Calendar   
Use Case ID: UC-16   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has the necessary permissions to create a Calendar.   
3. The system has the Calendar and Contact entities available for integration.   
4. The system supports linking Calendar events to emails or contacts.   
  
Postconditions:   
1. A new Calendar is created in the system.   
2. The Calendar is associated with the Administrator or a designated Contact.   
3. The system displays the new Calendar in the list of available calendars.   
4. The system logs the creation of the Calendar for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Calendar management section of the system.   
2. The Administrator selects the option to "Create Calendar."   
3. The system displays a form for entering the Calendar details (e.g., name, description, associated contact or email).   
4. The Administrator fills in the required information for the new Calendar.   
5. The Administrator clicks the "Save" button to create the Calendar.   
6. The system validates the input and creates the Calendar.   
7. The system updates the Calendar list and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not provide a valid Calendar name, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized calendar creation), it displays an error message and prevents the action.   
3. If the Calendar creation fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the creation process, the system returns to the Calendar management screen without making any changes.  
  
Use Case Name: View Calendar   
Use Case ID: UC-17   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has at least one Calendar created and associated with the Administrator or a Contact.   
3. The Administrator has permission to view the Calendar.   
4. The system supports integration with Email and Contact for event details.   
  
Postconditions:   
1. The Calendar and its associated events are displayed to the Administrator.   
2. The Administrator can view details of each event, including time, location, and related contacts or emails.   
3. The system logs the viewing action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The system displays the list of available calendars to which the Administrator has access.   
3. The Administrator selects a specific Calendar to view.   
4. The system loads and displays the Calendar with all its events.   
5. The Administrator can click on an event to view detailed information.   
6. The system shows the event details, including associated contacts or emails, if any.   
7. The system logs the viewing action.   
  
Alternative Flow:   
1. If the selected Calendar is not accessible due to permissions, the system displays an error message and logs the unauthorized access attempt.   
2. If the system cannot load the Calendar due to technical issues, it displays an error message and logs the failure for troubleshooting.   
3. If the Calendar contains no events, the system displays a message indicating the Calendar is empty.   
4. If the Administrator cancels the viewing process, the system returns to the previous screen without further action.  
  
Use Case Name: Update Calendar   
Use Case ID: UC-18   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Calendar created and associated with the Administrator or a Contact.   
3. The Administrator has permission to modify the selected Calendar.   
4. The system supports integration with Email and Contact for updating event details.   
  
Postconditions:   
1. The selected Calendar is updated with new or modified information.   
2. The changes are saved and reflected in the Calendar view.   
3. The system logs the update action for auditing and tracking purposes.   
4. If applicable, related events or contacts are also updated accordingly.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The Administrator selects a specific Calendar to update.   
3. The Administrator clicks the "Edit" button to modify the Calendar details.   
4. The system displays the Calendar form with current information.   
5. The Administrator updates the Calendar details (e.g., name, description, associated contact or email).   
6. The Administrator clicks the "Save" button.   
7. The system validates the input and saves the updated Calendar.   
8. The system updates the Calendar list and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not provide a valid Calendar name, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized calendar modification), it displays an error message and prevents the action.   
3. If the Calendar update fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the update process, the system returns to the Calendar management screen without making any changes.  
  
Use Case Name: Delete Calendar   
Use Case ID: UC-19   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Calendar created and associated with the Administrator or a Contact.   
3. The Administrator has permission to delete the selected Calendar.   
4. The system supports checking for dependencies (e.g., events linked to emails or contacts).   
  
Postconditions:   
1. The selected Calendar is removed from the system.   
2. Any events associated with the Calendar are either deleted or moved, depending on system settings.   
3. The Calendar list is updated to reflect the deletion.   
4. The system logs the deletion action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The Administrator selects a specific Calendar to delete.   
3. The Administrator clicks the "Delete" button.   
4. The system prompts the Administrator to confirm the deletion.   
5. The Administrator confirms the deletion.   
6. The system checks for any associated events and determines how to handle them (e.g., delete, archive).   
7. The system removes the Calendar from the system and updates the Calendar list.   
8. The system displays a confirmation message indicating the Calendar has been successfully deleted.   
  
Alternative Flow:   
1. If the selected Calendar is not accessible due to permissions, the system displays an error message and logs the unauthorized access attempt.   
2. If the system cannot delete the Calendar due to dependencies (e.g., events linked to active emails or contacts), it displays a warning and prompts the Administrator to resolve the issue.   
3. If the deletion fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the deletion confirmation, the system returns to the Calendar management screen without making any changes.  
  
Use Case Name: Schedule Event   
Use Case ID: UC-20   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has a Calendar and Contact entities available for integration.   
3. The Administrator has access to an Email Account or Shared Account for sending event invitations.   
4. The system supports creating and scheduling events linked to emails or contacts.   
  
Postconditions:   
1. A new event is scheduled and added to the Calendar.   
2. The event is associated with relevant Contact(s) or Email(s) if specified.   
3. An email invitation is sent to attendees using the selected Email Account or Shared Account.   
4. The system logs the scheduling action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The Administrator selects the option to "Schedule Event."   
3. The system displays a form for entering event details (e.g., title, date, time, location, attendees).   
4. The Administrator fills in the event details and selects associated Contact(s) or Email Account(s).   
5. The Administrator clicks the "Send Invitation" button to generate and send event emails.   
6. The system validates the event data and sends the invitation emails.   
7. The system creates the event in the Calendar and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not select a valid date or time, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized sending of emails), it displays an error message and prevents the action.   
3. If the email sending process fails due to network or configuration issues, the system displays an error message and logs the failure. The Administrator is prompted to retry or cancel.   
4. If the Administrator cancels the scheduling process, the system returns to the Calendar view without creating the event.  
  
Use Case Name: Modify Event   
Use Case ID: UC-21   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains an event already scheduled in the Calendar.   
3. The Administrator has permission to modify the selected event.   
4. The system supports integration with Contact and Email for updating event details.   
  
Postconditions:   
1. The selected event is updated with new or modified information.   
2. The changes are saved and reflected in the Calendar view.   
3. If the event is linked to an Email or Contact, the system updates the association accordingly.   
4. The system logs the modification action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The Administrator selects an event from the Calendar view.   
3. The Administrator clicks the "Edit" button to modify the event details.   
4. The system displays the event form with current information.   
5. The Administrator updates the event details (e.g., title, date, time, location, attendees).   
6. The Administrator selects the associated Contact(s) or Email Account(s) if needed.   
7. The Administrator clicks the "Save" button.   
8. The system validates the input and updates the event in the Calendar.   
9. The system displays a confirmation message indicating the event has been successfully modified.   
  
Alternative Flow:   
1. If the Administrator does not provide a valid date or time for the event, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized modification of events), it displays an error message and prevents the action.   
3. If the selected event is linked to an Email and the system cannot update the association, it displays an error message and logs the failure.   
4. If the modification fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the modification process, the system returns to the Calendar view without making any changes.  
  
Use Case Name: Cancel Event   
Use Case ID: UC-22   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains an event already scheduled in the Calendar.   
3. The Administrator has permission to cancel the selected event.   
4. The system supports integration with Contact and Email for managing event-related communications.   
  
Postconditions:   
1. The selected event is marked as canceled in the Calendar.   
2. If applicable, the system sends a cancellation notification to attendees via Email or updates the associated Contact records.   
3. The Calendar view is updated to reflect the cancellation.   
4. The system logs the cancellation action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The Administrator selects an event from the Calendar view.   
3. The Administrator clicks the "Cancel Event" button.   
4. The system prompts the Administrator to confirm the cancellation.   
5. The Administrator confirms the cancellation.   
6. The system updates the event status to "Canceled" and removes it from the Calendar schedule.   
7. The system sends a cancellation email to the attendees using the associated Email Account or Shared Account.   
8. The system logs the cancellation action and displays a confirmation message.   
  
Alternative Flow:   
1. If the selected event is not accessible due to permissions, the system displays an error message and logs the unauthorized access attempt.   
2. If the system cannot send the cancellation email due to network or configuration issues, it displays an error message and logs the failure. The Administrator is prompted to retry or cancel.   
3. If the Administrator cancels the cancellation confirmation, the system returns to the Calendar view without making any changes.   
4. If the system fails to update the event in the Calendar due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the event is already canceled or expired, the system displays a message indicating that it cannot be canceled again.  
  
Use Case Name: View Event Details   
Use Case ID: UC-23   
Actors: Administrator, Calendar, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one event scheduled in the Calendar.   
3. The Administrator has permission to view the selected event.   
4. The system supports integration with Contact and Email to display event-related details.   
  
Postconditions:   
1. The selected event's details are displayed to the Administrator.   
2. The Administrator can view information such as event title, date, time, location, attendees, and associated Contact or Email data.   
3. The system logs the viewing action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The Administrator selects an event from the Calendar view.   
3. The system retrieves and displays the event details in a dedicated view.   
4. The Administrator reviews the event information, including any associated Contact or Email details.   
5. The system logs the viewing action.   
  
Alternative Flow:   
1. If the selected event is not accessible due to permissions, the system displays an error message and logs the unauthorized access attempt.   
2. If the system cannot retrieve the event due to technical issues, it displays an error message and logs the failure for troubleshooting.   
3. If the event is linked to an Email or Contact and the system fails to display the association, it displays an error message.   
4. If the Administrator cancels the viewing process, the system returns to the Calendar view without further action.  
  
Use Case Name: Share Calendar   
Use Case ID: UC-24   
Actors: Administrator, Calendar, Email Account, Shared Account, Distribution Group, Contact, Email   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Calendar that is eligible for sharing.   
3. The Administrator has permission to share the selected Calendar.   
4. The system supports associating the shared Calendar with an Email Account, Shared Account, or Distribution Group.   
5. The system has the Email and Contact entities available for integration with the sharing process.   
  
Postconditions:   
1. The selected Calendar is shared with the specified recipient(s).   
2. The recipient(s) receive a notification or invitation to access the shared Calendar.   
3. The system logs the sharing action for auditing and tracking purposes.   
4. The shared Calendar is accessible to the recipient(s) based on the system's sharing policies.   
  
Main Flow:   
1. The Administrator navigates to the Calendar section of the system.   
2. The Administrator selects a specific Calendar to share.   
3. The Administrator clicks the "Share" button.   
4. The system displays a sharing form with options to specify the recipient(s) (e.g., individual Contact, Distribution Group, or Email address).   
5. The Administrator enters the recipient(s) and selects the appropriate Email Account or Shared Account to be used for sending the invitation.   
6. The Administrator clicks the "Send Invitation" button.   
7. The system generates a sharing invitation and sends it via email to the recipient(s).   
8. The system updates the Calendar's access permissions to include the recipient(s).   
9. The system displays a confirmation message indicating that the Calendar has been successfully shared.   
  
Alternative Flow:   
1. If the Administrator does not select a valid recipient or Email Account, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized sharing of calendars), it displays an error message and prevents the action.   
3. If the email sending process fails due to network or configuration issues, the system displays an error message and logs the failure. The Administrator is prompted to retry or cancel.   
4. If the Administrator cancels the sharing process, the system returns to the Calendar view without making any changes.   
5. If the selected Calendar is already shared with the specified recipient(s), the system displays a message indicating the existing sharing status.  
  
Use Case Name: Create Contact   
Use Case ID: UC-25   
Actors: Administrator, Contact, Email Account, Shared Account   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The Administrator has the necessary permissions to create a Contact.   
3. The system has a Contact entity available for integration.   
4. The Administrator has access to an Email Account or Shared Account.   
  
Postconditions:   
1. A new Contact is created in the system.   
2. The Contact is associated with an Email Account or Shared Account if specified.   
3. The system displays the new Contact in the Contact list.   
4. The system logs the creation of the Contact for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Contact management section of the system.   
2. The Administrator selects the option to "Create Contact."   
3. The system displays a form for entering the Contact details (e.g., name, email, phone, address).   
4. The Administrator fills in the required information for the new Contact.   
5. The Administrator selects an Email Account or Shared Account to associate with the Contact (optional).   
6. The Administrator clicks the "Save" button.   
7. The system validates the input data and creates the Contact.   
8. The system updates the Contact list and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not provide a valid name or contact information, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized creation of contacts), it displays an error message and prevents the action.   
3. If the system fails to save the Contact due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the creation process, the system returns to the Contact management screen without making any changes.  
  
Use Case Name: Update Contact   
Use Case ID: UC-26   
Actors: Administrator, Contact, Email Account, Shared Account   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains a Contact that needs to be updated.   
3. The Administrator has permission to modify the selected Contact.   
4. The system supports associating the Contact with an Email Account or Shared Account.   
  
Postconditions:   
1. The selected Contact is updated with new or modified information.   
2. The changes are saved and reflected in the Contact list.   
3. If the Contact is associated with an Email Account or Shared Account, the association is updated accordingly.   
4. The system logs the update action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Contact management section of the system.   
2. The Administrator selects a specific Contact to update.   
3. The Administrator clicks the "Edit" button.   
4. The system displays the Contact form with the current information.   
5. The Administrator updates the Contact details (e.g., name, email, phone, address).   
6. The Administrator selects or modifies the associated Email Account or Shared Account if needed.   
7. The Administrator clicks the "Save" button.   
8. The system validates the input data and updates the Contact.   
9. The system updates the Contact list and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not provide valid or complete information for the Contact, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized modification of contacts), it displays an error message and prevents the action.   
3. If the system fails to update the Contact due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the update process, the system returns to the Contact management screen without making any changes.  
  
Use Case Name: Delete Contact   
Use Case ID: UC-27   
Actors: Administrator, Contact, Email Account, Shared Account   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Contact in the Contact list.   
3. The Administrator has permission to delete the selected Contact.   
4. The system supports checking for dependencies (e.g., emails or shared accounts linked to the Contact).   
  
Postconditions:   
1. The selected Contact is removed from the system.   
2. Any associations between the Contact and Email Account or Shared Account are either removed or updated, depending on system settings.   
3. The Contact list is updated to reflect the deletion.   
4. The system logs the deletion action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Contact management section of the system.   
2. The Administrator selects a specific Contact to delete.   
3. The Administrator clicks the "Delete" button.   
4. The system prompts the Administrator to confirm the deletion.   
5. The Administrator confirms the deletion.   
6. The system checks for any dependencies (e.g., emails or shared accounts linked to the Contact).   
7. The system removes the Contact from the system and updates related entities if necessary.   
8. The system updates the Contact list and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not confirm the deletion, the system returns to the Contact management screen without making any changes.   
2. If the system detects dependencies (e.g., the Contact is linked to an active Email or Shared Account), it displays a warning and prompts the Administrator to resolve the issue before proceeding.   
3. If the system fails to delete the Contact due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the selected Contact is already deleted or archived, the system displays a message indicating that the Contact is no longer available for deletion.  
  
Use Case Name: View Contact   
Use Case ID: UC-28   
Actors: Administrator, Contact, Email Account, Shared Account   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Contact in the Contact list.   
3. The Administrator has permission to view the selected Contact.   
4. The system supports integration with Email Account and Shared Account for displaying contact-related email details.   
  
Postconditions:   
1. The selected Contact's details are displayed to the Administrator.   
2. The Administrator can view information such as name, email, phone, address, and associated Email Account or Shared Account.   
3. The system logs the viewing action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Contact management section of the system.   
2. The Administrator selects a specific Contact from the list.   
3. The system retrieves and displays the Contact's details, including associated Email Account or Shared Account information if available.   
4. The Administrator reviews the displayed information.   
5. The system logs the viewing action.   
  
Alternative Flow:   
1. If the selected Contact is not accessible due to permissions, the system displays an error message and logs the unauthorized access attempt.   
2. If the system cannot retrieve the Contact due to technical issues, it displays an error message and logs the failure for troubleshooting.   
3. If the Contact is linked to an Email Account or Shared Account and the system fails to display the association, it displays an error message.   
4. If the Administrator cancels the viewing process, the system returns to the Contact list without further action.  
  
Use Case Name: Manage Contact Group   
Use Case ID: UC-29   
Actors: Administrator, Contact, Contact Group, Email Account, Shared Account   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the Contact Group entity configured.   
3. The Administrator has the necessary permissions to manage Contact Groups.   
4. The system contains one or more Contacts that can be added to or removed from a group.   
  
Postconditions:   
1. The Contact Group is either created, updated, or deleted, based on the Administrator's action.   
2. The system updates the group's associated Contacts and reflects changes in the Contact Group list.   
3. The system logs the management action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Contact Group management section of the system.   
2. The Administrator selects an action (e.g., create, edit, or delete a Contact Group).   
3. If creating, the Administrator enters the group name and adds selected Contacts to the group.   
4. If editing, the Administrator selects a Contact Group and modifies its name or membership.   
5. If deleting, the Administrator selects a Contact Group and confirms the deletion.   
6. The system validates the action against the Policy rules.   
7. The system updates the Contact Group accordingly.   
8. The system displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the system detects a violation of the Policy rules (e.g., unauthorized group deletion), it displays an error message and prevents the action.   
2. If the Administrator does not provide a valid group name, the system displays an error message and prompts for correction.   
3. If the selected group is already associated with active Contacts or workflows, the system displays a warning message before proceeding with deletion.   
4. If the system fails to update the Contact Group due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the operation, the system returns to the Contact Group management screen without making any changes.  
  
Use Case Name: Create Distribution Group   
Use Case ID: UC-30   
Actors: Administrator, Distribution Group, Email Account, Shared Account, Email, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the Distribution Group entity configured.   
3. The Administrator has the necessary permissions to create a Distribution Group.   
4. The system has access to one or more Email Accounts or Shared Accounts for association.   
5. The system has a Policy entity to ensure compliance with group creation rules.   
  
Postconditions:   
1. A new Distribution Group is created in the system.   
2. The group is associated with selected Email Accounts or Shared Accounts.   
3. The system updates the list of available Distribution Groups.   
4. The system logs the creation action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Distribution Group management section of the system.   
2. The Administrator selects the option to "Create Distribution Group."   
3. The system displays a form for entering the group name, description, and selecting members.   
4. The Administrator fills in the required information and selects associated Email Accounts or Shared Accounts.   
5. The Administrator clicks the "Save" button to finalize the creation.   
6. The system validates the input and checks for Policy compliance.   
7. The system creates the Distribution Group and adds it to the group list.   
8. The system displays a confirmation message indicating the group has been successfully created.   
  
Alternative Flow:   
1. If the Administrator does not provide a valid group name, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized group creation), it displays an error message and prevents the action.   
3. If the system fails to save the Distribution Group due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the creation process, the system returns to the Distribution Group management screen without making any changes.  
  
Use Case Name: Update Distribution Group   
Use Case ID: UC-31   
Actors: Administrator, Distribution Group, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Distribution Group that needs to be updated.   
3. The Administrator has permission to modify the selected Distribution Group.   
4. The system has Email Account and Shared Account entities available for association.   
5. The system has a Policy entity configured to validate group modifications.   
  
Postconditions:   
1. The selected Distribution Group is updated with new or modified information.   
2. The group's associated Email Accounts or Shared Accounts are updated accordingly.   
3. The system updates the Distribution Group list to reflect the changes.   
4. The system logs the update action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Distribution Group management section of the system.   
2. The Administrator selects a specific Distribution Group to update.   
3. The Administrator clicks the "Edit" button.   
4. The system displays the group form with current information.   
5. The Administrator modifies the group name, description, or members as needed.   
6. The Administrator selects or modifies the associated Email Account or Shared Account if necessary.   
7. The Administrator clicks the "Save" button.   
8. The system validates the input data and checks for Policy compliance.   
9. The system updates the Distribution Group with the new information.   
10. The system updates the group list and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not provide a valid group name, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized group modification), it displays an error message and prevents the action.   
3. If the system fails to update the Distribution Group due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the update process, the system returns to the Distribution Group management screen without making any changes.  
  
Use Case Name: Delete Distribution Group   
Use Case ID: UC-32  
Actors: Administrator, Distribution Group, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Distribution Group that needs to be deleted.   
3. The Administrator has permission to delete the selected Distribution Group.   
4. The system has a Policy entity to ensure compliance with group deletion rules.   
5. The system supports checking for dependencies (e.g., emails or shared accounts linked to the group).   
  
Postconditions:   
1. The selected Distribution Group is removed from the system.   
2. Any associations between the group and Email Accounts or Shared Accounts are either removed or updated, depending on system settings.   
3. The Distribution Group list is updated to reflect the deletion.   
4. The system logs the deletion action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Distribution Group management section of the system.   
2. The Administrator selects a specific Distribution Group to delete.   
3. The Administrator clicks the "Delete" button.   
4. The system prompts the Administrator to confirm the deletion.   
5. The Administrator confirms the deletion.   
6. The system checks for any dependencies (e.g., emails or shared accounts linked to the group).   
7. The system removes the Distribution Group from the system and updates related entities if necessary.   
8. The system updates the Distribution Group list and displays a confirmation message.   
  
Alternative Flow:   
1. If the selected Distribution Group is not accessible due to permissions, the system displays an error message and logs the unauthorized access attempt.   
2. If the system detects dependencies (e.g., the group is linked to active emails or shared accounts), it displays a warning and prompts the Administrator to resolve the issue before proceeding.   
3. If the system detects a violation of the Policy rules (e.g., unauthorized group deletion), it displays an error message and prevents the action.   
4. If the deletion fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the deletion confirmation, the system returns to the Distribution Group management screen without making any changes.  
  
Use Case Name: Add Member to Distribution Group   
Use Case ID: UC-33   
Actors: Administrator, Distribution Group, Email Account, Shared Account, Contact   
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has at least one Distribution Group available for modification.   
3. The Administrator has permission to manage the selected Distribution Group.   
4. The system contains one or more Contacts or Email Accounts that can be added as members.   
5. The system has a Policy entity configured to validate member additions.   
  
Postconditions:   
1. A new member is added to the selected Distribution Group.   
2. The system updates the Distribution Group’s member list.   
3. The system logs the addition action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Distribution Group management section of the system.   
2. The Administrator selects a specific Distribution Group to modify.   
3. The Administrator clicks the "Add Member" option.   
4. The system displays a list of available Contacts or Email Accounts for selection.   
5. The Administrator selects one or more members to add to the group.   
6. The Administrator clicks the "Save" button.   
7. The system validates the selected members and checks for Policy compliance.   
8. The system updates the Distribution Group with the new members.   
9. The system displays a confirmation message indicating the members have been successfully added.   
  
Alternative Flow:   
1. If the Administrator does not select any members, the system displays an error message and prompts for selection.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized addition of members), it displays an error message and prevents the action.   
3. If the system fails to update the Distribution Group due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the addition process, the system returns to the Distribution Group management screen without making any changes.  
  
Use Case Name: Remove Member from Distribution Group   
Use Case ID: UC-34   
Actors: Administrator, Distribution Group, Email Account, Shared Account, Contact, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Distribution Group with members.   
3. The Administrator has permission to manage the selected Distribution Group.   
4. The system supports removing members (e.g., Contacts or Email Accounts) from the group.   
5. The system has a Policy entity to validate the removal process.   
  
Postconditions:   
1. The selected member is removed from the Distribution Group.   
2. The system updates the Distribution Group's member list to reflect the removal.   
3. The system logs the removal action for auditing and tracking purposes.   
4. If applicable, the system ensures no dependencies or policies are violated by the removal.   
  
Main Flow:   
1. The Administrator navigates to the Distribution Group management section of the system.   
2. The Administrator selects a specific Distribution Group to modify.   
3. The Administrator clicks the "Remove Member" option.   
4. The system displays the list of current members in the group.   
5. The Administrator selects one or more members to remove.   
6. The Administrator clicks the "Confirm Removal" button.   
7. The system validates the removal against the Policy rules.   
8. The system removes the selected members from the group.   
9. The system updates the Distribution Group member list and displays a confirmation message.   
  
Alternative Flow:   
1. If the system detects a violation of the Policy rules (e.g., unauthorized removal of members), it displays an error message and prevents the action.   
2. If the Administrator does not select any member to remove, the system displays an error message and prompts for selection.   
3. If the system fails to update the Distribution Group due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the selected members are linked to active emails or workflows, the system displays a warning and prompts the Administrator to confirm or resolve the issue.   
5. If the Administrator cancels the removal process, the system returns to the Distribution Group management screen without making any changes.  
  
Use Case Name: Configure Server Archive   
Use Case ID: UC-35   
Actors: Administrator, Server Archive, Email Account, Shared Account   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the Server Archive entity configured.   
3. The Administrator has the necessary permissions to configure the Server Archive.   
4. The system supports defining archive policies and retention rules.   
5. The system has access to one or more Email Accounts or Shared Accounts for association.   
  
Postconditions:   
1. The Server Archive is configured with the specified settings.   
2. Archive policies and retention rules are updated as defined by the Administrator.   
3. The system logs the configuration action for auditing and tracking purposes.   
4. The Server Archive is ready to store emails according to the new configuration.   
  
Main Flow:   
1. The Administrator navigates to the Server Archive configuration section of the system.   
2. The system displays the current configuration settings for the Server Archive.   
3. The Administrator modifies the configuration settings, including archive policies and retention rules.   
4. The Administrator selects which Email Accounts or Shared Accounts will be associated with the Server Archive.   
5. The Administrator clicks the "Save Configuration" button.   
6. The system validates the input data and checks for compliance with Policy rules.   
7. The system applies the new configuration to the Server Archive.   
8. The system displays a confirmation message indicating that the Server Archive has been successfully configured.   
  
Alternative Flow:   
1. If the Administrator does not select any Email Account or Shared Account for association, the system displays an error message and prompts for selection.   
2. If the system detects invalid or incomplete configuration data, it displays an error message and prompts the Administrator to correct it.   
3. If the system detects a violation of the Policy rules (e.g., unauthorized archive configuration), it displays an error message and prevents the configuration.   
4. If the configuration fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the configuration process, the system returns to the Server Archive management screen without making any changes.  
  
Use Case Name: Set Archiving Policy   
Use Case ID: UC-36   
Actors: Administrator, Email Account, Shared Account, Archive, Server Archive, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the Archive and Server Archive entities available for configuration.   
3. The Administrator has the necessary permissions to configure archiving policies.   
4. The system contains at least one Email Account or Shared Account.   
5. The system has a Policy entity configured to define rules for archiving.   
  
Postconditions:   
1. The archiving policy is configured and applied to the selected Email Account or Shared Account.   
2. Emails associated with the account will be archived based on the defined rules.   
3. The system logs the policy configuration for auditing and tracking purposes.   
4. The system updates the policy settings to reflect the new configuration.   
  
Main Flow:   
1. The Administrator navigates to the Archiving Policy configuration section of the system.   
2. The system displays a form for defining the archiving rules (e.g., time-based retention, folder destination).   
3. The Administrator selects the Email Account or Shared Account to which the policy will apply.   
4. The Administrator sets the archiving criteria, such as retention period and archive destination (Archive or Server Archive).   
5. The Administrator clicks the "Save Policy" button.   
6. The system validates the input and checks for Policy compliance.   
7. The system applies the archiving policy to the selected account.   
8. The system displays a confirmation message indicating the archiving policy has been successfully set.   
  
Alternative Flow:   
1. If the Administrator does not select an Email Account or Shared Account, the system displays an error message and prompts for selection.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized policy changes), it displays an error message and prevents the action.   
3. If the system fails to save the archiving policy due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the policy configuration process, the system returns to the previous screen without making any changes.  
  
Use Case Name: Capture Email Flow   
Use Case ID: UC-37   
Actors: Administrator, Email Flow, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has an Email Flow entity configured to monitor and analyze email activities.   
3. The Administrator has access to an Email Account or a Shared Account.   
4. The system has a Policy entity to enforce rules for capturing and managing email flows.   
  
Postconditions:   
1. The system captures and records the email flow details.   
2. The captured flow is associated with the Email Account or Shared Account.   
3. The email flow is stored in the system for auditing and analysis.   
4. The system logs the flow capture action for tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Email Flow management section of the system.   
2. The Administrator selects the option to "Capture Email Flow."   
3. The system prompts the Administrator to select the Email Account or Shared Account for flow monitoring.   
4. The Administrator selects the relevant account.   
5. The system initiates the capture process and begins monitoring outgoing, incoming, and internal email communications.   
6. The system logs the flow details, including sender, recipient, timestamp, and subject.   
7. The system associates the captured flow with the selected Email Account or Shared Account.   
8. The system displays a confirmation message indicating that the email flow has been successfully captured.   
  
Alternative Flow:   
1. If the Administrator does not select an Email Account or Shared Account, the system displays an error message and prompts for a valid selection.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized flow capture), it displays an error message and prevents the action.   
3. If the email flow capture fails due to technical issues, the system displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the capture process, the system returns to the Email Flow management screen without capturing any data.  
  
Use Case Name: Monitor Compliance   
Use Case ID: UC-38   
Actors: Administrator, Email, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has access to one or more Email Accounts or Shared Accounts.   
3. The system has a Policy entity configured to define compliance rules for email management.   
4. The system is capable of scanning and evaluating emails for compliance violations.   
  
Postconditions:   
1. The system evaluates the selected or system-wide emails for compliance with the defined Policy rules.   
2. Any compliance violations are reported to the Administrator.   
3. The system logs the compliance check for auditing and tracking purposes.   
4. The Administrator can take corrective actions based on the compliance report.   
  
Main Flow:   
1. The Administrator navigates to the Compliance Monitoring section of the system.   
2. The Administrator selects the scope of the compliance check (e.g., specific emails, all emails from an account, or system-wide).   
3. The system retrieves the selected emails and begins scanning for compliance violations.   
4. The system evaluates each email against the configured Policy rules (e.g., content restrictions, attachment size, unauthorized recipients).   
5. The system compiles a compliance report with details of any violations.   
6. The system displays the compliance report to the Administrator.   
7. The system logs the compliance monitoring action.   
  
Alternative Flow:   
1. If the Administrator does not select a valid scope for compliance monitoring, the system displays an error message and prompts for a valid selection.   
2. If the system detects no compliance violations, it displays a message indicating that no issues were found.   
3. If the system fails to retrieve or evaluate the emails due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the system is unable to access the Policy rules for evaluation, it displays an error message and prevents the compliance check.   
5. If the Administrator cancels the compliance monitoring process, the system returns to the compliance management screen without performing the check.  
  
Use Case Name: Restore Archived Email   
Use Case ID: UC-39   
Actors: Administrator, Archive, Server Archive, Email, Email Account, Shared Account, Policy   
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains one or more emails stored in the Archive or Server Archive.   
3. The Administrator has the necessary permissions to restore emails.   
4. The system has a Policy entity configured to define rules for email restoration.   
5. The system supports retrieving emails from the Archive or Server Archive.   
  
Postconditions:   
1. The selected archived email is restored and moved back to the Administrator's inbox or a designated folder.   
2. The email is no longer marked as archived or expired.   
3. The system updates the Archive or Server Archive folder to reflect the removal.   
4. The system logs the restoration action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Archive or Server Archive section of the system.   
2. The Administrator selects an email to be restored.   
3. The Administrator clicks the "Restore" button.   
4. The system prompts the Administrator to confirm the restoration.   
5. The Administrator confirms the restoration.   
6. The system retrieves the email from the Archive or Server Archive.   
7. The system moves the email to the inbox or a designated folder based on system settings.   
8. The system logs the restoration action.   
9. The system displays a confirmation message indicating the email has been successfully restored.   
  
Alternative Flow:   
1. If the selected email is not accessible due to permissions, the system displays an error message and logs the unauthorized access attempt.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized email restoration), it displays an error message and prevents the action.   
3. If the system fails to retrieve the email from the Archive or Server Archive, it displays an error message and logs the failure for troubleshooting.   
4. If the email is already restored or not archived, the system displays a message indicating the email is no longer in an archived state.   
5. If the Administrator cancels the restoration confirmation, the system returns to the Archive/Server Archive list without making any changes.  
  
Use Case Name: Manage Archiving Policy   
Use Case ID: UC-40  
Actors: Administrator, Email Account, Shared Account, Archive, Server Archive, Policy   
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains at least one Email Account or Shared Account.   
3. The system has Archive and Server Archive entities available for configuration.   
4. The system has a Policy entity configured to define archiving rules.   
5. The Administrator has the necessary permissions to manage archiving policies.   
  
Postconditions:   
1. The archiving policy is configured and applied to the selected Email Account or Shared Account.   
2. Emails associated with the account will be archived based on the defined rules.   
3. The system logs the policy configuration for auditing and tracking purposes.   
4. The system updates the policy settings to reflect the new configuration.   
  
Main Flow:   
1. The Administrator navigates to the Archiving Policy configuration section of the system.   
2. The system displays a form for defining the archiving rules (e.g., retention period, archive destination).   
3. The Administrator selects the Email Account or Shared Account to which the policy will apply.   
4. The Administrator sets the archiving criteria, such as retention period and archive destination (Archive or Server Archive).   
5. The Administrator clicks the "Save Policy" button.   
6. The system validates the input and checks for Policy compliance.   
7. The system applies the archiving policy to the selected account.   
8. The system displays a confirmation message indicating the archiving policy has been successfully set.   
  
Alternative Flow:   
1. If the Administrator does not select an Email Account or Shared Account, the system displays an error message and prompts for selection.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized policy changes), it displays an error message and prevents the action.   
3. If the system fails to save the archiving policy due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the Administrator cancels the policy configuration process, the system returns to the previous screen without making any changes.  
  
Use Case Name: Manage Shared Account Access   
Use Case ID: UC-41   
Actors: Administrator, Shared Account, Email Account, Policy, Contact   
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains one or more Shared Accounts.   
3. The Administrator has the necessary permissions to manage access to Shared Accounts.   
4. The system has a Policy entity to validate access rules.   
5. The system supports associating access rights with Contacts or Email Accounts.   
  
Postconditions:   
1. Access rights to the Shared Account are configured or modified based on the Administrator's action.   
2. The system updates the access list for the Shared Account.   
3. The system logs the access management action for auditing and tracking purposes.   
4. The associated Contact or Email Account is granted or revoked access accordingly.   
  
Main Flow:   
1. The Administrator navigates to the Shared Account management section.   
2. The Administrator selects a specific Shared Account to manage access.   
3. The system displays the current access list for the Shared Account.   
4. The Administrator selects an action (e.g., grant access to a Contact or Email Account, revoke access).   
5. If granting access, the Administrator selects the Contact or Email Account to add.   
6. If revoking access, the Administrator selects the Contact or Email Account to remove.   
7. The Administrator confirms the action.   
8. The system validates the action against the Policy rules.   
9. The system updates the access rights of the Shared Account.   
10. The system logs the access change and displays a confirmation message.   
  
Alternative Flow:   
1. If the Administrator does not select a valid Contact or Email Account, the system displays an error message and prompts for a valid selection.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized access grant), it displays an error message and prevents the action.   
3. If the system fails to update the access rights due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the selected Shared Account is already configured with the desired access, the system displays a message indicating the existing access status.   
5. If the Administrator cancels the access management process, the system returns to the Shared Account management screen without making any changes.  
  
Use Case Name: Manage Event Record   
Use Case ID: UC-42   
Actors: Administrator, EventRecord, Calendar, Email, Contact, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains one or more Event Records associated with the Calendar.   
3. The Administrator has the necessary permissions to manage Event Records.   
4. The system has a Policy entity configured to validate modifications to event records.   
5. The system supports integration with Calendar, Email, and Contact entities for managing event details.   
  
Postconditions:   
1. The Event Record is either created, updated, or deleted based on the Administrator's action.   
2. The Calendar is updated to reflect changes in the event record.   
3. If the event is linked to an Email or Contact, the association is updated accordingly.   
4. The system logs the management action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Event Record management section of the system.   
2. The Administrator selects an action (e.g., create, edit, or delete an Event Record).   
3. If creating, the Administrator fills in the event details, selects a Calendar, and optionally links the event to an Email or Contact.   
4. If editing, the Administrator selects an existing Event Record, modifies its details (e.g., title, date, time, attendees, or associated Email/Contact).   
5. If deleting, the Administrator selects an Event Record and confirms the deletion.   
6. The system validates the action and checks for compliance with the Policy rules.   
7. The system updates the Event Record and synchronizes changes with the Calendar and associated entities.   
8. The system displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the Administrator does not provide valid event details, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized modification or deletion of an event), it displays an error message and prevents the action.   
3. If the selected Event Record is linked to an active Email or Contact and the system cannot update the association, it displays an error message.   
4. If the system fails to update or delete the Event Record due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the operation, the system returns to the Event Record management screen without making any changes.  
  
Use Case Name: Manage Archive Policy   
Use Case ID: UC-43  
Actors: Administrator, Archive Policy, Email Account, Shared Account, Archive, Server Archive, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has an Archive Policy entity configured.   
3. The Administrator has the necessary permissions to manage Archive Policies.   
4. The system contains one or more Email Accounts or Shared Accounts.   
5. The system supports defining and applying rules for email archiving to the Archive or Server Archive.   
  
Postconditions:   
1. The Archive Policy is either created, updated, or deleted based on the Administrator's action.   
2. The system updates the policy settings for the selected Email Account or Shared Account.   
3. Emails associated with the account will be archived based on the updated Archive Policy.   
4. The system logs the management action for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Archive Policy management section of the system.   
2. The Administrator selects an action (e.g., create, edit, or delete an Archive Policy).   
3. If creating, the Administrator enters the policy name and defines archiving rules, such as retention period, archive destination (Archive or Server Archive), and conditions for automatic archiving.   
4. If editing, the Administrator selects an existing Archive Policy and modifies its details, including retention settings and destinations.   
5. If deleting, the Administrator selects an Archive Policy and confirms the deletion.   
6. The system validates the action and checks for compliance with the overarching Policy rules.   
7. The system applies the changes to the Archive Policy and associates it with the relevant Email Account or Shared Account.   
8. The system updates the Archive Policy list and displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the Administrator does not provide a valid policy name or incomplete rules, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., attempting to delete a mandatory Archive Policy), it displays an error message and prevents the action.   
3. If the selected Archive Policy is already applied to active Email Accounts or Shared Accounts, the system displays a warning and prompts the Administrator to resolve the dependency.   
4. If the system fails to update or delete the Archive Policy due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the operation, the system returns to the Archive Policy management screen without making any changes.  
  
Use Case Name: Manage Recovery Request   
Use Case ID: UC-44   
Actors: Administrator, Recovery Request, Email, Archive, Server Archive, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system contains one or more Recovery Requests submitted by users or system processes.   
3. The system has access to the Archive or Server Archive where the requested emails are stored.   
4. The system has a Policy entity configured to define rules for handling recovery requests.   
5. The Administrator has the necessary permissions to view and manage Recovery Requests.   
  
Postconditions:   
1. The Recovery Request is either approved, denied, or deleted based on the Administrator's action.   
2. If approved, the associated email is recovered and moved back to the user's inbox or a designated folder.   
3. The system logs the action taken on the Recovery Request for auditing and tracking purposes.   
4. The Recovery Request list is updated to reflect the changes in request status.   
  
Main Flow:   
1. The Administrator navigates to the Recovery Request management section of the system.   
2. The system displays a list of pending Recovery Requests.   
3. The Administrator selects a specific Recovery Request to review.   
4. The system retrieves and displays the details of the request, including the email subject, sender, recipient, and the reason for recovery.   
5. The Administrator evaluates the request based on the system Policy and other contextual information.   
6. The Administrator chooses an action (e.g., approve, deny, or delete the request).   
7. The system prompts the Administrator to confirm the selected action.   
8. The Administrator confirms the action.   
9. The system performs the selected action:   
 - If approved, the system retrieves the email from the Archive or Server Archive and moves it back to the user's inbox or a specified folder.   
 - If denied, the system updates the request status to "Denied" and optionally provides a denial reason.   
 - If deleted, the system removes the Recovery Request from the system.   
10. The system updates the Recovery Request list and logs the action.   
11. The system displays a confirmation message indicating the action has been successfully completed.   
  
Alternative Flow:   
1. If the selected email is not found in the Archive or Server Archive, the system displays an error message and logs the failure.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized approval of a request), it displays an error message and prevents the action.   
3. If the Administrator does not provide a valid reason for denial or deletion, the system displays an error message and prompts for input.   
4. If the system fails to update the Recovery Request or the email due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the action confirmation, the system returns to the Recovery Request list without making any changes.  
  
Use Case Name: Manage Compliance Report   
Use Case ID: UC-45   
Actors: Administrator, Compliance Report, Email, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has generated at least one Compliance Report based on email monitoring or policy violations.   
3. The Administrator has the necessary permissions to view, modify, or delete Compliance Reports.   
4. The system has a Policy entity configured to define rules for compliance reporting.   
5. The system supports retrieving and analyzing emails for compliance violations.   
  
Postconditions:   
1. The Compliance Report is either viewed, modified, or deleted based on the Administrator's action.   
2. If modified, the Compliance Report is updated to reflect new or corrected information.   
3. If deleted, the Compliance Report is removed from the system or archived based on system settings.   
4. The system logs all actions taken on the Compliance Report for auditing and tracking purposes.   
  
Main Flow:   
1. The Administrator navigates to the Compliance Report section of the system.   
2. The system displays a list of available Compliance Reports, including details such as report date, affected Email Accounts, and summary of violations.   
3. The Administrator selects a Compliance Report from the list.   
4. The system retrieves and displays the selected Compliance Report, including detailed violation descriptions, impacted emails, and policy references.   
5. The Administrator reviews the Compliance Report and chooses an action (e.g., approve, modify, or delete the report).   
6. If modifying, the Administrator updates the report's details (e.g., adding comments, updating the status of violations).   
7. If deleting, the Administrator confirms the deletion or selects an archive option if applicable.   
8. The system validates the action against the Policy rules.   
9. The system updates or deletes the Compliance Report accordingly.   
10. The system updates the Compliance Report list and logs the action.   
11. The system displays a confirmation message indicating the success of the action.   
  
Alternative Flow:   
1. If the Administrator does not select a valid Compliance Report, the system displays an error message and prompts for a valid selection.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized modification or deletion of a Compliance Report), it displays an error message and prevents the action.   
3. If the system fails to retrieve the Compliance Report due to technical issues, it displays an error message and logs the failure for troubleshooting.   
4. If the selected Compliance Report is already archived or deleted, the system displays a message indicating that it is no longer available for modification.   
5. If the Administrator cancels the operation, the system returns to the Compliance Report management screen without making any changes.  
  
Use Case Name: Manage User   
Use Case ID: UC-46   
Actors: Administrator, User, Email Account, Shared Account, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the User entity configured to manage user profiles and access rights.   
3. The Administrator has the necessary permissions to manage Users.   
4. The system has access to Email Accounts and Shared Accounts for integration with user roles and permissions.   
5. The system has a Policy entity to enforce rules for user management.   
  
Postconditions:   
1. The User is either created, updated, or deleted, based on the Administrator's action.   
2. The system updates the list of Users and their associated roles, Email Accounts, or Shared Accounts.   
3. The system logs the management action for auditing and tracking purposes.   
4. If applicable, the system sends a notification to the User regarding account creation or modification.   
  
Main Flow:   
1. The Administrator navigates to the User management section of the system.   
2. The Administrator selects an action (e.g., create, edit, or delete a User).   
3. If creating, the Administrator enters the User details, such as username, full name, role, and associated Email Account or Shared Account.   
4. The system validates the input data and checks for compliance with the Policy rules.   
5. The system creates the User and assigns the specified roles and access rights.   
6. The system updates the User list and displays a confirmation message.   
7. The system may send a notification to the User via email to inform them of their account creation or modification.   
  
If editing, the Administrator selects an existing User, modifies their details (e.g., role, access rights, or associated accounts), and clicks the "Save" button.   
If deleting, the Administrator selects a User and confirms the deletion. The system removes the User from the system and updates the User list accordingly.   
  
Alternative Flow:   
1. If the Administrator does not provide valid or complete User information, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized user deletion or role assignment), it displays an error message and prevents the action.   
3. If the selected User is associated with active Email Accounts, Shared Accounts, or workflows, the system displays a warning and prompts the Administrator to resolve the issue before proceeding.   
4. If the system fails to update or delete the User due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the operation, the system returns to the User management screen without making any changes.  
  
Use Case Name: Manage Administrator   
Use Case ID: UC-47   
Actors: Administrator, User, Policy   
  
Preconditions:   
1. The Administrator is logged into the system.   
2. The system has the User entity configured to distinguish between regular users and administrators.   
3. The Administrator has the necessary permissions to manage other Administrators.   
4. The system has a Policy entity configured to define rules for administrator management.   
5. The system supports creating, updating, or deleting administrator roles and access rights.   
  
Postconditions:   
1. The Administrator is either created, updated, or deleted based on the Administrator's action.   
2. The system updates the list of Administrators and their associated roles and permissions.   
3. The system logs the management action for auditing and tracking purposes.   
4. If applicable, the system sends a notification to the affected administrator regarding their role change or deletion.   
  
Main Flow:   
1. The Administrator navigates to the Administrator management section of the system.   
2. The Administrator selects an action (e.g., create, edit, or delete an Administrator).   
3. If creating, the Administrator enters the new Administrator's details, such as username, full name, role, and permissions.   
4. The system validates the input data and checks for compliance with the Policy rules.   
5. The system creates the new Administrator and assigns the specified roles and access rights.   
6. The system updates the Administrator list and displays a confirmation message.   
7. The system may send a notification to the new Administrator via email to inform them of their account creation.   
  
If editing, the Administrator selects an existing Administrator, modifies their details (e.g., role, permissions, or associated accounts), and clicks the "Save" button.   
If deleting, the Administrator selects an Administrator and confirms the deletion. The system removes the Administrator from the system and updates the Administrator list accordingly.   
  
Alternative Flow:   
1. If the Administrator does not provide valid or complete information for creating or editing an Administrator, the system displays an error message and prompts for correction.   
2. If the system detects a violation of the Policy rules (e.g., unauthorized administrator deletion or role assignment), it displays an error message and prevents the action.   
3. If the selected Administrator is associated with active Email Accounts, Shared Accounts, or workflows, the system displays a warning and prompts the Administrator to resolve the issue before proceeding.   
4. If the system fails to update or delete the Administrator due to technical issues, it displays an error message and logs the failure for troubleshooting.   
5. If the Administrator cancels the operation, the system returns to the Administrator management screen without making any changes.