Policy: Hybrid Access Model

1. Permanent Cross-Department Communication

Who approves: Admin only
Scope:

- Dept-to-dept rules
- Specific dept roles (e.g., IT → all)
- Long-term or organization-wide collaboration
  Behavior:
  Once approved, any user in those departments can email each other without asking again.

Example:
✔ IT ↔ All
✔ HR ↔ Legal (permanent)

---

2. Temporary Access / One-Off Conversations

Who approves: Department Manager
Scope:

- Single user ↔ another dept user
- Time-limited or message-limited
- Auto-expire rules
  Behavior:
  Users request → Manager approves → Access only for that user and limited duration.

Example:
Employee in HR needs to email someone in Finance
→ HR Manager approves
→ Access valid for e.g. 24 hrs or 3 messages

---

3. Logging & Audit (Mandatory)

Every action must be logged:

- Who requested contact
- Who approved (manager/admin)
- Duration or permanence
- Timestamp + reason

Admins and auditors can monitor everything.

System Design Implications

Roles Required

- **Admin**
- **Manager**
- **User**
- (Optional) **Auditor**

🗂️ DB Tables Needed

You'll need something like:

departments
id | name

users
id | name | dept_id | role

communication_rules
id | dept_a | dept_b | type (temporary/permanent) | approved_by | expiry_datetime | user_specific? | requester_id

message_logs
id | sender | receiver | timestamp | status

Workflow Example (Temporary Access)

1. User tries emailing outside dept → blocked
2. System offers “Request Access”
3. Request saved with:
    - sender, target dept/user, reason
4. Manager receives request
5. If approves → temporary rule created with expiry
6. Sender can send email until expiry
7. Logs everything

Workflow Example (Permanent Access)

1. Manager or user cannot create it
2. Only Admin dashboard has this power
3. Admin sets: "Dept A ↔ Dept B (permanent)"
4. Rule active until removed’

DATABASE

Tables:

Depts:
Id(will work as primary and foreign key to connect)
name(just naming)

Roles:
Id
name(admin/manager/auditor)(maybe more we will see)

Users:
Id
Name
email(company generated)
Pass hash
dept_id (FK → departments)

role_id (FK → roles)

public_key / encrypted_private_key

Comm rules:
Id

dept_a_id	**FK to departments**
dept_b_id	**FK to departments**
rule_type	**'temporary' or 'permanent'**
requester_id	**FK to users (nullable for permanent)**
approved_by_id	**FK to users**
expiry_timestamp	**Nullable for permanent rules**

user_specific	**Boolean (if temporary access is user-to-user only)**

Middleware/Logic for Enforcing Message Sending Permissions

Before building UI or endpoints, the system needs to check:

- **Sender's department**
- **Recipient's department**
- **Whether a valid rule exists in `communication_rules`**
- **Whether the rule is still valid (temporary/permanent)**
- **Whether user-specific or dept-wide**

This places the control at a central point.

Add Request/Approval Workflows

After the foundation, we can enable:

- **Temporary message access request (user-initiated)**
- **Manager dashboard for approvals**
- **Admin dashboard for permanent links**

Add Messaging + Encryption on Top