

Business Strategy / Operational Systems

Create guidelines for internal communication (Slack, email, project management tools) to reduce notification overload.

Difficulty: Intermediate

Model: GPT-4 / Claude / Gemini

Use Case: Internal Communication, Notification Management, Team Collaboration

Updated: May 2026

Why This Prompt Exists

Most teams have communication chaos — Slack notifications all day, email buried, missed messages in project tools.

You get:

- notification overload (constant interruptions)
- missed important messages (buried in noise)
- no clear channel for different communication types
- decision paralysis (which tool for which message?)
- slower responses (can't prioritize)

But communication protocols are not rules.

They are guidelines that reduce noise and increase signal.

- Slack/Teams: quick questions, informal updates, real-time
- Email: external communication, formal decisions, async
- Project tools: task-specific, documented, accountable
- Wiki/Docs: permanent reference, processes, policies

Without protocols, important messages get lost in noise.

This framework forces AI to build communication guidelines.

The Prompt

Assume the role of a communication strategist who reduces notification overload.

Your task is to create communication protocols.

Generate:

1. CHANNEL PURPOSE STATEMENTS

- Slack/Teams: what goes here
- Email: what goes here
- Project management: what goes here
- Wiki/Docs: what goes here

2. SLACK/TEAMS GUIDELINES

- When to use (vs. email vs. project tool)
- Response time expectations
- @here and @channel rules
- Do not disturb hours

3. EMAIL GUIDELINES

- When to use email (vs. Slack)
- Response time expectations (24-48 hours)
- Subject line standards
- Internal email vs. external email

4. PROJECT MANAGEMENT TOOL GUIDELINES

- Task assignment and comments
- Status updates
- File storage
- @mention rules

5. ASYNC COMMUNICATION EXPECTATIONS

- No expectation of immediate response
- Default to async unless urgent
- What constitutes urgent

6. COMMUNICATION REVIEW PROCESS

- How often to review protocols
- Who can suggest changes

INPUTS:

Team Size:

[INSERT NUMBER]

Remote or In-person:

[REMOTE / HYBRID / IN-PERSON]

Current Tools:

[SLACK / TEAMS / EMAIL / ASANA / TRELLO / NOTION / OTHER]

Current Pain Points:

[E.G., "Too many Slack notifications," "Missed emails," "No one checks project comments"]

Time Zones (if remote):

[LIST]

RULES:

- Each channel has a clear purpose (no overlap)
- Slack: quick, informal, non-urgent
- Email: formal, external, async
- Project tools: task-specific, accountable
- Async default: no expectation of immediate response
- Urgent means truly urgent (not everything)

How To Use It

- Slack for quick questions, not decision documentation.
- Email for external communication and formal decisions.
- Project tools for task-specific communication.
- Async default: don't expect immediate replies.
- Review protocols quarterly as tools and team evolve.

Example Input

Team Size: 12 people (distributed across 4 time zones)

Remote or In-person: REMOTE

Current Tools: Slack, Gmail, Asana, Google Docs

Current Pain Points: "Slack notifications never stop," "People expect immediate answers," "Important decisions buried in Slack threads," "No one knows which tool to use for what"

Time Zones: PST, MST, CST, EST, GMT

Why It Works

Most teams have communication chaos.

This framework improves outcomes by forcing:

- channel purpose clarity (what goes where)
- response time expectations (reduce anxiety)
- async default (respect focus time)
- urgency definition (protect deep work)
- review process (continuous improvement)

Great communication protocols don't add rules — they reduce noise and increase signal.

Build Better AI Systems

Subscribe for advanced prompt engineering, AI business strategy tools, operational systems frameworks, and practical strategies for leaders and operators.

Carefully engineered prompts for people doing real work.

Share this:

- [Share on Facebook \(Opens in new window\) Facebook](#)
- [Share on X \(Opens in new window\) X](#)

See also [The Core Business Process Map](#)