

# SCOPE DOCUMENT

**Product Name:** (TBD – AI Planning Assistant)

**Document Type:** Scope / Vision / MVP / V1 / Roadmap

**Version:** Draft 1

**Team:** ATC – Team 2

## 1. Product Overview

The system connects to a user's communication channels (Gmail, Slack, WhatsApp, Messages, etc.), captures any actionable information (tasks, events, commitments, follow-ups), prioritizes them, and automatically plans the user's day by syncing everything to their calendar.

The goal is to reduce stress, eliminate mental load, and provide a reliable external system that handles planning without requiring constant user input.

## 2. Product Vision

To build a personal AI executive assistant that continuously listens to the user's digital world, understands what needs to be done, and organizes their time intelligently.

The system should create "mind like water": no clutter, no mental overhead, no missed commitments. Users focus on doing the work, not on planning it.

## 3. Philosophy / Guiding Principles

The system follows a simple decision-making loop inspired by GTD and real executive assistants:

**Capture → Clarify → Organize → Prioritize → Schedule → Notify**

- **Capture:** Extract tasks/events from all communication channels.
- **Clarify:** Identify if the item is actionable, time-bound, or informational.
- **Organize:** Group by context (project, person, deadline).
- **Prioritize:** Score items based on urgency, importance, sender, context.
- **Schedule:** Block time on the user's calendar automatically.
- **Notify:** Alert the user about trigger events, changes, and critical deadlines.

The system should feel:

**Simple, predictable, helpful, and trustworthy.**

## 4. Problem We Are Solving

Users deal with:

- Scattered information across email, Slack, WhatsApp, SMS, and meetings.
- Tasks and deadlines hidden inside conversations.
- Manual planning and reprioritization that take 30–90 minutes daily.
- High cognitive load: “What should I work on next?”
- Missed follow-ups and forgotten commitments.
- Calendars that do not understand context or importance.

No single system unifies all communication → priority → planning → scheduling.

## 5. Scope

### In Scope

- Reading communication channels (permissions-based)
- Detecting tasks, events, deadlines, follow-ups
- Categorizing and interpreting actionable items
- Priority scoring
- Automatic daily planning
- Writing time blocks and events to user’s calendar
- Real-time updates when new high-priority messages arrive
- User configuration: which chats/threads/apps can be accessed
- Basic security and trigger-based data capture

### Out of Scope (for now)

- Sending messages on behalf of the user
- Negotiating meeting times
- Multi-user/team workflows
- Deep project management
- Third-party integrations beyond primary channels (later phases)
- Offline/desktop “memory” like Rewind

## 6. MVP (Version 0)

The smallest functional version that demonstrates the core loop:

**multi-channel capture → prioritization → planning → calendar sync.**

### MVP Capabilities

- **Gmail & Slack integration** (read-only)
- **Trigger-based extraction:** deadlines, tasks, meeting requests
- **Basic priority engine:** urgency + deadlines + simple rules
- **Auto-scheduling:** insert tasks into free blocks in Google Calendar
- **Daily plan summary** (text-based)
- **Dashboard (simple):** tasks extracted, schedule created
- **User config:** choose which inboxes/chats are monitored
- **Basic notifications** for critical items

## MVP Goals

- Show that the system can *reliably extract tasks* from real messages
- Show that users save time planning their day
- Build user trust through accuracy and transparent scheduling
- Deliver one clear output: **a daily plan they can follow**

## 7. Version 1.0

Once MVP validates the workflow, V1 focuses on intelligence, reliability, and broader coverage.

### V1 Additions

- **WhatsApp & Messages ingestion**
- **Advanced NLU extraction:** implicit tasks (“Can you get this done today?”)
- **Smart reprioritization:** rearrange the day when new items appear
- **User preference learning:** preferred focus hours, meeting times, work patterns
- **Meeting prep summaries:** pull context from prior messages
- **Better dashboard:** timeline view, task overrides, explanations
- **Spam/Noise filtering** for chat-heavy channels
- **Higher-accuracy priority engine** (ML-driven)

### V1 Goals

- Deeper intelligence
- Faster adoption
- Trustworthy automation
- Expanded coverage of user workstreams

## 8. Roadmap

### Phase 1 — Foundations (0–2 months)

- Channel integration (Gmail + Slack)

- Basic extraction model
- Rule-based priority scoring
- Calendar time-blocking
- Simple dashboard
- Closed user testing

### **Phase 2 — Intelligence Layer (2–5 months)**

- WhatsApp integration
- ML-based extraction
- Preference learning
- Auto-rescheduling
- Real-time alerts
- Early meeting summaries

### **Phase 3 — Experience & Reliability (5–8 months)**

- Strong UI/UX for schedule viewing
- Override system (user controls)
- Noise filtering
- Mobile notifications
- Accuracy improvements

### **Phase 4 — AI Assistant Evolution (8–12 months)**

- Coordinating with other users
- Suggesting meeting times
- Full “AI Chief of Staff” workflow
- Integration with project tools
- Smart habits + weekly planning

## **9. Functional Requirements**

### **1. Ingestion Engine**

- Connect to communication channels
- Pull messages/threads (configurable)
- Identify trigger events to minimize privacy risk

### **2. Extraction Engine**

- Locate tasks, events, deadlines, follow-ups
- Clarify what is actionable vs informational

### 3. **Priority Engine**

- Score items based on urgency, importance, sender weight, deadlines

### 4. **Planning Engine**

- Convert tasks into time blocks
- Avoid conflicts
- Place high-priority items first
- Adjust when new items arrive

### 5. **Sync Engine**

- Write tasks/events to Google Calendar
- Update when changes occur

### 6. **Notification/Update System**

- Real-time alerts for critical items
- Daily plan summary

### 7. **User Preference Module**

- Save per-user rules, focus times, do-not-disturb hours

## 10. **Non-Functional Requirements**

### **Security & Privacy**

- Only capture data after explicit permission
- Allow user to choose specific threads/channels
- Trigger-based extraction to minimize data retention
- Encrypt data in transit and at rest

### **Performance**

- Extraction should occur within seconds of message arrival
- Calendar updates must be reliable and timely

### **Accuracy**

- MVP: ≥70% extraction accuracy
- V1: ≥90% accuracy (tasks + deadlines + meetings)

## **Reliability**

- System must not double-book or overwrite user's manual events
- Clear logic for conflict resolution

## **Transparency**

- Show users "why" an item was prioritized or scheduled