SpeakUp

A Systems-Engineering Demonstration

Bruce Dombrowski

December 22, 2025

Repository: `https://github.com/brucedombrowski/SpeakUp`

Generated: December 22, 2025 16:09

About This Document

Purpose: This briefing is designed for asynchronous review by managers and customers. It can be read independently without a presenter.

What SpeakUp Is:

- A response to organizational calls for constructive employee input
- A response to customer requests for process improvement ideas
- A demonstration of systems-engineering discipline applied to knowledge work
- Vendor-neutral at the requirements level—no specific tool is proposed

Repository: All artifacts, verification evidence, and this briefing are available at:

`https://github.com/brucedombrowski/SpeakUp`

Problem Statement

The current operating environment has systemic constraints that limit effectiveness:

Constraint	**Impact**
Fragmented workflows	Mobile ideation, desktop development, and execution environments are disconnected
Limited AI in trusted boundaries	Forces workflow degradation or excessive abstraction to stay compliant
Broadcast email as work proxy	Reduces signal-to-noise ratio and interrupts deep effort
Untracked coordination systems	Billable work in untracked systems limits traceability and auditability
Knowledge attrition risk	Legacy decommissioning, budget reduction, and personnel transition threaten institutional knowledge

Governing Principle

Core Principle

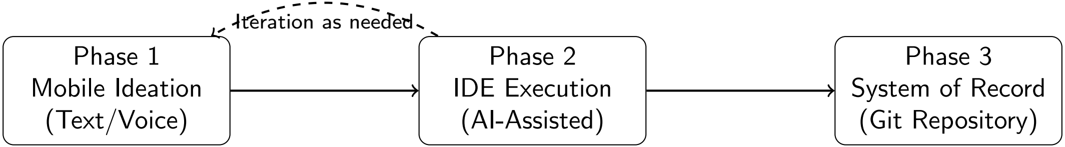
Thinking is necessary and expected.
Accountable work begins when thinking is captured.

This principle guides the proposed workflow. Work performed in structured, tracked systems maximizes:

Principle	**Benefit**
Work in structured systems	Enables automation support and reduces manual overhead
Capture in tracked systems	Provides traceability for audits and reviews
Git as system of record	Creates authoritative, version-controlled history
Email for notification only	Preserves email for time-critical coordination, not as work artifact

Proposed Workflow Model

The workflow has three phases that can iterate:



Phase 1: Ideation

- Smartphone-based reasoning
- Text input always available
- Voice input when possible
- No controlled data required

Phase 2: Execution

- Modern IDE environment
- AI assistance (modular)
- Within trust boundaries
- Produces artifacts

Phase 3: Record

- Git version control
- Captures history
- Captures rationale
- Authoritative source

Functional Requirements (Solution-Agnostic)

These requirements define *what* is needed, not *how* to implement it:

ID	Type	Requirement
FR-1	Mandatory	Mobile ideation capability (smartphone, text/voice input)
FR-2	Mandatory	IDE-centric execution with integrated, replaceable AI assistance
FR-3	Mandatory	Git-based system of record capturing artifacts, history, and rationale
FR-4	Mandatory	Identity and trust boundary alignment (security at identity and device)
FR-5	Recommended	High-signal communication model (email for notification only)

Security and Compliance

SpeakUp maintains existing security posture—no rules are relaxed:

Trust Boundary Alignment

- Security enforced at authenticated identity
- Security enforced at managed device
- AI operates in-boundary as assistive tool
- Classification and handling rules unchanged

Information Handling (This Project)

- No sensitive PII included
- No CUI included
- No proprietary information included
- No classified information included
- Verified by inspection (see repository)

Verification evidence: `verification/Compliance-Statement.md`

Value Proposition

The Core Point

With the right environment, I can produce significantly more software.

AI-assisted development with proper tooling dramatically multiplies output. The constraint is not capability—it is environment.

Capability	**Current State**	**Proposed State**
Work capture	Fragmented, untracked	Structured, version-controlled
AI assistance	Outside boundary or unavailable	In-boundary, modular
Knowledge preservation	At-risk	Durable artifacts
Automation readiness	Limited	Maximized
Auditability	Manual effort	Built-in traceability

Implementation Approach

This project demonstrates the pattern by being the pattern:

- **Concrete enough to execute**
  - Working repository with all artifacts
  - Defined outputs and verification evidence
  - Reproducible workflow documented in `artifacts/Workflow-Log.md`
- **Abstract enough to remain vendor-neutral**
  - Requirements specify *what*, not *how*
  - Implementation choices documented separately
  - Alternative tools can satisfy same requirements
- **Self-demonstrating**
  - This briefing was created using the proposed workflow
  - Ideation on mobile, execution in IDE, artifacts in Git

Example project using this workflow:
`https://github.com/brucedombrowski/OpenSourceHouseProject`

Repository Contents

All project artifacts are available for review:

File	Purpose
README.md	Authoritative requirements and project spe
briefing/SpeakUp-Briefing.pdf	This document
briefing/SpeakUp-Briefing.tex	Source for this document (version-controlle
verification/Compliance-Statement.md	Information handling verification evidence
verification/Requirements-Traceability.md	Requirements to evidence mapping
artifacts/Workflow-Log.md	Execution workflow documentation

https://github.com/brucedombrowski/SpeakUp

Verification Summary

This project produces verification evidence as first-class artifacts:

Method	**Application**	**Evidence**
Inspection	Document review for completeness	Compliance-Statement.md
Analysis	Compliance assessment against requirements	Requirements-Traceability.md
Demonstration	Working repository as proof of concept	This repository

Key Compliance Findings:

- All information handling requirements: **COMPLIANT**
- All functional requirements: **ADDRESSED** or **DEMONSTRATED**
- All expected outputs: **COMPLETE**

Recommendation

Adopt the SpeakUp workflow model as a pattern for:

- Converting thinking into durable, reviewable artifacts
- Preserving institutional knowledge as personnel transition
- Enabling automation and reducing manual audit effort
- Maintaining security and trust boundaries while using AI assistance
- Improving signal-to-noise in organizational communication

This pattern is applicable to:

- Engineering work
- Analytical work
- Knowledge work generally

Next Steps

1. **Review this briefing** and the repository contents
2. **Identify a pilot application area** where the workflow could be applied
3. **Establish repository and workflow** for the pilot
4. **Iterate** between ideation and execution phases
5. **Measure and refine** based on results

This briefing was produced using the SpeakUp workflow model it describes.

Repository: `https://github.com/brucedombrowski/SpeakUp`

Contact: Bruce Dombrowski (Creator)