# Question: What can I actually expect when utilizing requires\_completion\_signoff?

## Critical Finding: ALL Patterns Are Metadata-Only

**NONE of these values trigger system enforcement.** They're ALL workflow convention patterns.

### Source Code Reality:

requires\_completion\_signoff: Optional[str] = Field(  
 default="true",   
 description="Whether task requires signoff. May be one of 'true', 'false' of 'human\_required'"  
)

The system just stores the value - no validation, no blocking, no enforcement.

## The Three Patterns (Semantic Convention, Not Code Enforcement)

### requires\_completion\_signoff: "true" (default)

**Semantic Intent**: "This task needs validation before considered complete"  
  
- **NOT enforced by system**  
  
- **Who validates**: Prime agent or orchestrator (another AGENT)  
  
- **Workflow**: Clone completes → Prime agent reviews → Prime signs off with completion\_signoff\_by  
  
- **Use when**: Want agent-to-agent validation before moving on

### requires\_completion\_signoff: "false"

**Semantic Intent**: "Can be auto-completed without validation"  
  
- **NOT enforced by system**  
  
- **Who validates**: Nobody (auto-complete)  
  
- **Workflow**: Clone completes → Sets completed=true → No signoff needed  
  
- **Use when**: Simple, routine tasks where validation doesn't add value

### requires\_completion\_signoff: "human\_required"

**Semantic Intent**: "Needs HUMAN validation, not just agent"  
  
- **NOT enforced by system**  
  
- **Who validates**: Human stakeholder (YOU)  
  
- **Workflow**: Clone completes → Agent marks ready → HUMAN reviews → Human/agent sets completion\_signoff\_by: "Master Ethan"  
  
- **Use when**: Critical quality gates, compliance checkpoints, important decisions

## The Practical Difference

|  |  |  |
| --- | --- | --- |
| **Pattern** | **Who Signs Off** | **Use Case** |
| "true" | Any agent (prime/orchestrator) | Standard validation gates |
| "false" | Nobody (auto-complete) | Routine tasks |
| "human\_required" | Human stakeholder | Critical quality gates |

## What Does NOT Happen (For Any Pattern)

* ❌ System does NOT block agent from marking complete
* ❌ System does NOT notify anyone that review is needed
* ❌ System does NOT prompt for approval
* ❌ System does NOT enforce signoff requirement

## Workflow Patterns That Make It Work

**Pattern 1: Agent Self-Discipline**  
  
- Agent persona instructs them to check requires\_completion\_signoff value  
  
- Agent STOPS and notifies appropriate party before marking complete  
  
- Relies on agent following instructions

**Pattern 2: User-Driven Review**  
  
- You check planning tool for tasks requiring signoff  
  
- You review completion reports  
  
- You manually approve or request revisions

**Pattern 3: Orchestrator Validation**  
  
- Orchestrator checks signoff status before proceeding  
  
- Notifies appropriate party if signoff missing  
  
- Relies on orchestrator workflow discipline

## Example Workflow (Human Required)

1. Agent creates task withrequires\_completion\_signoff: "human\_required"
2. Agent completes work, writescompletion\_report
3. **Agent STOPS and notifies you**: "This requires your review"
4. You review the deliverable
5. You approve or request changes
6. Agent marks complete withcompletion\_signoff\_by: "your\_name"

## Bottom Line

**ALL THREE PATTERNS HAVE IDENTICAL ENFORCEMENT**: None. Zero. Nada.

**The difference is SEMANTIC CONVENTION** for multi-agent workflow patterns.

This feature is:  
  
- ✅ Workflow convention metadata  
  
- ✅ Semantic intent signaling  
  
- ✅ Coordination pattern enabler  
  
- ✅ Audit trail documentation

This feature is NOT:  
  
- ❌ System-enforced blocking  
  
- ❌ Automated notification  
  
- ❌ Technical validation gate  
  
- ❌ Mandatory review system

**Key Insight:** It's metadata that tells agents "this is the intended workflow pattern" - but enforcement must be built into agent personas, not relied upon from the system.

## HOW TO ACTUALLY IMPLEMENT A QUALITY GATE STOP

If you want an agent to **actually stop and wait** for human approval, you must explicitly build this behavior into the agent's persona.

### Required Persona Instructions

## Quality Gate Protocol  
When completing tasks:  
1. Check the task's `requires\_completion\_signoff` value  
2. If `"human\_required"`:  
 - Complete the work and write `completion\_report`  
 - \*\*STOP and notify Master Ethan\*\* that review is required  
 - \*\*WAIT for explicit approval\*\* before marking task complete  
 - Only mark complete after approval with `completion\_signoff\_by: "admin"`  
3. If `"true"`: Follow standard validation workflow (prime/orchestrator approval)  
4. If `"false"`: Mark complete immediately

### How Approval Must Be Given/Received

**Agent Behavior:**  
  
1. Agent completes work  
  
2. Agent writes completion\_report with deliverable details  
  
3. Agent notifies you: "Task [name] requires your review. Please review [deliverable location] and approve."  
  
4. **Agent STOPS** - does NOT mark task complete  
  
5. **Agent WAITS** for your explicit response

**Your Response:**  
  
- **To Approve**: "Approved, you can mark it complete" or "Looks good, proceed"  
  
- **To Request Changes**: "Please revise [specific feedback]"

**Agent Completion:**  
  
- Only after your approval, agent calls:  
  
wsp\_update\_task(  
 task\_id="...",  
 completed=True,  
 completion\_signoff\_by="admin" # Your username  
 )

### Critical Requirements for This to Work

✅ **Persona must explicitly instruct** agent to stop and wait  
  
✅ **Agent must follow instructions** (depends on agent discipline)  
  
✅ **You must actually review** and provide approval  
  
✅ **Clear communication protocol** for approval/revision requests

❌ **System will NOT enforce** the stop - it's all agent behavior  
  
❌ **Without persona instructions** - agent ignores field and continues  
  
❌ **No automatic notifications** - agent must explicitly notify you

## AGENT-TO-AGENT VERIFICATION GATE

Use requires\_completion\_signoff: "true" for agent verification (not "human\_required").

### Required Setup:

**Completing Agent Persona:**

## Quality Gate Protocol  
When task has `requires\_completion\_signoff: "true"`:  
1. Complete work and write `completion\_report`  
2. \*\*STOP and notify verifier agent\*\* by name  
3. \*\*WAIT for verifier's approval\*\* before proceeding

**Verifier Agent Persona:**

## Verification Protocol  
When notified of work requiring verification:  
1. Review deliverable against specified criteria  
2. If approved: Call `wsp\_update\_task(completed=True, completion\_signoff\_by="verifier\_agent\_name")`  
3. If needs revision: Provide feedback and request changes

### Workflow:

1. Clone completes → notifies Verifier
2. Verifier reviews → approves or requests revisions
3. Verifier signs off with their agent name
4. Orchestrator proceeds only after verification

### Requirements:

✅ Both agents need explicit persona instructions for their roles  
  
✅ Verification standards must be explicit in verifier's persona  
  
✅ Orchestrator must coordinate the handoff

❌ Still no system enforcement - relies on agent behavior