

SKILL

- 1 CC2.0 VERIFY Function Skill
 - 1.1 Planned Capabilities
 - 1.2 Planned Input/Output
 - 1.3 Implementation Timeline
 - 1.4 References

1 CC2.0 VERIFY Function Skill

Function: VERIFY - Quality Verification & Testing **Category Theory:** Applicative Functor (verification composition) **Purpose:** Verify implementations meet quality standards through automated testing **Status:** 📅 Planned (Phase 2, Weeks 4-6)

1.1 Planned Capabilities

- ✅ **Automated Testing:** Unit, integration, e2e test execution
 - 📊 **Quality Metrics:** Code coverage, complexity, security scans
 - 🎯 **Verification Strategies:** Functional, property-based, mutation testing
 - 🔍 **Bug Detection:** Static analysis, runtime validation
 - ✅ **Acceptance Criteria:** Pass/fail verdict with evidence
-

1.2 Planned Input/Output

Input (from CREATE):

```
{
  "implementation": {
    "code": "...",
    "tests": "..."
  },
  "criteria": {
    "minCoverage": 0.80,
    "maxComplexity": 15,
    "securityScan": true
  }
}
```

Output:

```
{
  "verdict": "pass" | "fail",
  "evidence": {
    "coverage": 0.87,
    "complexity": 12,
    "securityIssues": []
  },
  "confidence": 0.92
}
```

1.3 Implementation Timeline

Phase 2: Weeks 4-6 (after Foundation Release) **Trigger:** CREATE function stable, real testing needs identified **Estimated Effort:** 4 hours (following established patterns)

1.4 References

- Specification: `~/cc2.0/functions/verify/FUNCTION.md`
 - Implementation Plan: `~/cc2.0/PHASE-1-FOUNDATION-SPEC.md` (Phase 2 section)
-

Status: Deferred to Phase 2 - Learn from actual usage patterns first