

# /plan-with-team

**Source:** `~/ .claude/skills/plan-with-team/SKILL.md`

---

name: plan-with-team description:  
Create implementation plans with builder/validator agent teams. Use for major features, refactoring, or system components. argument-hint: "[what to build]"

## Plan With Team

Create an implementation plan with builder/validator agent teams.

## Related Skills (pick the right one)

- `/plan-with-team` (this skill) — Multi-expert planning for complex features, refactors, system components.
- `/build-plan` — Execute the plan produced here. Run AFTER plan is approved.
- `/hop-build <task_id>` — For a SINGLE task (not a multi-task plan).
- `/build` — Toggle session into Build Mode.
- `/prime-build` — Load lightweight build context into session.

# Instructions

You are creating a detailed implementation plan that assigns work to specialized agent teams.

## Step 1: Research (MANDATORY)

Before planning, research:

1. **If building a company/org:** Find 2-3 existing examples, analyze their structure
2. **If building software:** Explore the codebase, understand existing patterns
3. **If building process:** Find industry standards and best practices

Use Glob, Grep, Read, WebSearch as needed. Document findings.

## Step 2: Analyze

Based on research:

- What patterns should we copy?
- What should we adapt for our needs?
- What are the key components?

## Step 3: Define Team

For each major task, assign:

- **Builder** — Creates/implements
- **Validator** — Verifies the work

Reference agents from `~/claude/agents/`:

- `builder.md` — Implementation agent
- `validator.md` — Verification agent (read-only)

## Step 4: Create Plan

Write plan to `~/system/specs/<name>-plan.md` with this structure:

```
# Plan: [Name]
```

```
## Research Summary
```

[What we learned from existing examples]

## ## Objective

[1-2 sentences]

## ## Team Orchestration

### ### Team Members

ID	Name	Role	Agent Type
B1	[name]-builder	Build [what]	builder
V1	[name]-validator	Validate [what]	validator

### ### Step-by-Step Tasks

#### #### Phase 1: [Name]

**Task 1:** [Description]

- Owner: B1
- BlockedBy: none
- Acceptance:
  - [ ] Criterion 1
  - [ ] Criterion 2

**Task 2:** Validate [above]

- Owner: V1
- BlockedBy: 1
- Acceptance: [criteria]

## ## Validation Commands

[How to verify the work]

# Step 5: Validate Plan

Self-validate the plan:

- All tasks have acceptance criteria
- Dependencies make sense (validators blocked by builders)
- No circular dependencies

**MANDATORY CHECK — plan is INCOMPLETE without both:**

- Validation task** exists — owner: Proveo/Angie Jones, end-to-end test with real evidence (not dry-run), BlockedBy all build tasks
- Documentation task** exists — owner: Skillforge, BookStack page for every system built or changed, BlockedBy validation task

If either is missing → add them before presenting to CEO. Do not ask. Just add.

## Step 6: Present for Approval

Show the user:

1. Research summary
2. Plan overview
3. Number of tasks and phases
4. Ask: "Approve plan? Then run `/build-plan` to execute."

## Output

The plan file path: `~/system/specs/<name>-plan.md`

Ready for execution with `/build-plan`.

\$ARGUMENTS

---

Revision #8

Created 2026-02-18 08:39:48 UTC by John

Updated 2026-06-21 20:00:42 UTC by John