

AI-Driven Development - 30-Day Challenge Task-7

Name: Zoya Afzal

Slot: Friday 6–9 PM

◆ Task Overview

💡 **Task:** What is **Spec-Kit Plus**? – Write a short note.

📦 Students must download **Spec-Kit Plus** and review its 5 core concepts:

- 📝 /constitution
- 📋 /specify
- 🧠 /plan
- 🛠️ /tasks
- ⚙️ /implement

✨ What is Spec-Kit Plus (Short Note)

Spec-Kit Plus is an advanced toolkit for **Spec-Driven Development (SDD)**, where clear specifications are written before any code is generated. 🛠️

It builds on the original GitHub Spec Kit and provides a structured workflow for creating **AI-native applications**, including **multi-agent systems** 🤖.

In Spec-Kit Plus, specifications, architecture decisions, plans, tasks, and prompt histories are treated as **first-class project files**, ensuring clarity and consistency 📁.

The toolkit offers **templates, automation scripts, and AI-tool integrations** that guide you from writing specs to planning, task generation, and implementation 🧩.

It is installed via **Python** 🐍 and initialized with simple **CLI commands** 💻, making setup easy.

Overall, Spec-Kit Plus enforces **structured development**, reducing errors ⚠️ and producing more maintainable, reliable AI-generated software ✅.

🌟 Core Concepts of Spec-Kit Plus

🏛️ Constitution Phase

The **Constitution Phase** sets the **foundational rules** for a project 📜 by defining coding standards, testing guidelines, and documentation requirements in a single `constitution.md` file.

All future specs and code must follow these rules 🧱, ensuring consistency and quality. It acts as the **guiding framework** for the entire development workflow 📁.

📋 Specify Phase (Specification Phase)

The **Specify Phase** turns an idea into a **clear, structured set of requirements** 📝.

Using the `/sp.specify` command, you generate a `spec.md` file 📄 (plus requirement checklist) describing:

- Feature functionality ⚙️
- User stories 👤
- Acceptance criteria ✅
- Edge cases ⚡
- Constraints 🚧

🧠 Plan Phase

The **Plan Phase** translates the specification into a **technical roadmap** 🗺️.

Using `/sp.plan`, it generates files like `plan.md` and `data-model.md` 📁 that outline architecture, data models, APIs, and implementation details 🏗️.

This phase ensures all **project rules are followed** and **technical decisions clarified** ⚙️, providing a structured guide for development.

🔧 Tasks Phase

The **Tasks Phase** breaks the plan into **actionable work units** 🛠️ using `/sp.tasks`, generating a `tasks.md` file 📄.

Each task is **small**, **test-driven**, and clearly ordered 🧩, making implementation manageable and traceable ✅.

This phase ensures all work **aligns with the spec and plan**, ready for structured development 📌.

⚙️ Implement Phase

The **Implement Phase** turns the plan and tasks into **working code** 💻 using `/sp.implement`. It follows a **Test-Driven Development workflow**:

1. Write tests first (RED) 🔴
2. Code to pass them (GREEN) 🟢
3. Refactor (REFACTOR) ✨

Tasks are executed in order ⌚, producing **source code, tests, and updated progress** aligned with the specification and plan 📊.

```
C:\WINDOWS\system32\cmd.exe

GEMINI

tips for getting started:
1. Ask questions, edit files, or run commands.
2. Be specific for the best results.
3. /help for more information.

Using: 2 GEMINI.md files | 2 MCP servers

> /sp

sp.tasks      Generate an actionable, dependency-ordered tasks.md for the feature based on available design ar...
sp.specify    Create or update the feature specification from a natural language feature description.
sp.plan       Execute the implementation planning workflow using the plan template to generate design artifact...
sp.phr        Record an AI exchange as a Prompt History Record (PHR) for learning and traceability.
sp.implement  Execute the implementation plan by processing and executing all tasks defined in tasks.md
sp.git.commit_pr An autonomous Git agent that intelligently executes git workflows. Your task is to intelligently...
sp.constitution Create or update the project constitution from interactive or provided principle inputs, ensurin...
sp.clarify    Identify underspecified areas in the current feature spec by asking up to 5 highly targeted clar...

(1/13)
~\my-research-paper (main*)      no sandbox (see /docs)      auto
```

