

10-Day AI-Assisted Development Sprint

Outcomes

- **Scale Engineering Throughput:** Hardcode AI efficiencies directly into the team's daily developer routine to eliminate manual overhead.
- **Implement Spec-Driven Development (SDD):** Move cleanly from product requirements (PRDs/User Stories) to bulletproof technical specifications using AI as an architectural partner.
- **Automate Quality Gates:** Deploy AI utilities to automatically generate complete test plans (Unit, Integration, E2E), ensuring velocity doesn't break code quality.
- **Secure the AI Pipeline:** Establish strict guardrails for context management, data privacy, and code ownership risks.

Agenda Timeline

Day 1–2: Prep, Setup & Pipeline Audit

- **Objective:** Understand the current tech stack, development workflow, and biggest engineering bottlenecks.
- **Activities:**
 - Audit the existing delivery pipeline (from PRD to Deployment).
 - Align with the CTO/designated R&D leader on boundaries, goals, and key performance metrics.
 - Configure the team's baseline AI setup (IDE extensions, LLM access, context window limitations).
- **Deliverable:** Clear sprint boundaries and team alignment.

Day 3–4: Core Execution Sprint (Hands-On Workshop)

- **Objective:** The core interactive session (2 hours to a half-day depending on team size) where the team transitions to an AI-augmented workflow.
- **Session Structure:**
 - **Module 1: AI-Powered Product & Requirements Alignment**
 - Leveraging AI to parse PRDs, map out complex User Stories, and uncover edge cases instantly.

- **Module 2: Spec-Driven Development (SDD) & Architecture**
 - Transitioning the team to an SDD framework—generating rigid technical specs and architectural plans before writing code.
- **Module 3: Maximum Throughput Coding & Infrastructure**
 - Advanced contextual prompting techniques, Pair Programming with AI, automated Code Reviews, and smart Refactoring.
- **Module 4: Automated Test Generation**
 - Using AI to automatically draft comprehensive test plans covering Unit, Integration, and End-to-End (E2E) testing.

Day 5–7: Hardcoding the Efficiency & Guardrails

- **Objective:** Solidify the workflows built in the workshop and lock down security.
- **Activities:**
 - Implement AI Agent utilities to automate repetitive pipeline tasks and pipeline overhead.
 - Set up data privacy and context safety guardrails (protecting proprietary code, managing ownership and security risks).
 - Review early code written under the new SDD framework.

Day 8–10: Follow-up & Momentum Check

- **Objective:** Measure early velocity gains and ensure adoption sticks.
- **Activities:**
 - Review metrics on engineering throughput and test coverage.
 - Conduct a follow-up check-in with the developers to unblock friction points in their new daily routines.
 - Deliver the final **AI-Driven Execution Blueprint** to the leadership team for sustained scaling.