🤖 AI-Driven Development: Task 2 Submission

Part A: Theory (Short Questions)

1. Nine Pillars Understanding

Why is using AI Development Agents (like Gemini CLI) for repetitive setup tasks better for your growth as a system architect?

Using **AI Development Agents** for routine tasks is better for growth because it conserves your **mental capacity** that would otherwise be wasted on trivial, cyclical activities (like environment setup or templating). This frees you to focus on **higher-level functions** such as system architecture, design, strategic decision-making, and orchestrating complex workflows. By offloading the routine work, your skills are refined toward designing systems and developing the **instinct of an architect** instead of just a coder.

Explain how the Nine Pillars of AIDD help a developer grow into an M-Shaped Developer.

The **Nine Pillars** provide an integrated system of tools, agents, tests, and workflows that significantly **reduce the friction** of learning new domains. Because **AI covers knowledge gaps**, a developer can effectively work across many interrelated fields (e.g., architecture, testing, DevOps, and coding). This multi-domain exposure builds a strong foundation in several complementary areas, leading to a developer becoming an **M-Shaped Developer**—someone with deep expertise in multiple domains.

---

2. Vibe Coding vs. Specification-Driven Development (SDD)

Why does Vibe Coding usually create problems after one week?

Vibe Coding fails quickly because it lacks organization, documented choices, and consistent strategies. Since it relies only on intuition, the logic quickly becomes erratic, features break, and developers often forget the rationale behind previous decisions. Without a clear **source of truth** (no specification), corrective measures become extremely difficult, causing significant problems as the project scales.

How would Specification-Driven Development prevent those problems?

Specification-Driven Development (SDD) prevents these issues by requiring a detailed specification document to be completed *before* any coding begins. This document acts as a **guiding star**, formally recording every function and behavior to eliminate ambiguity. Outlining everything upfront ensures the work is predictable, testable, and extensible, maintaining the original intention and preventing chaotic development.

---

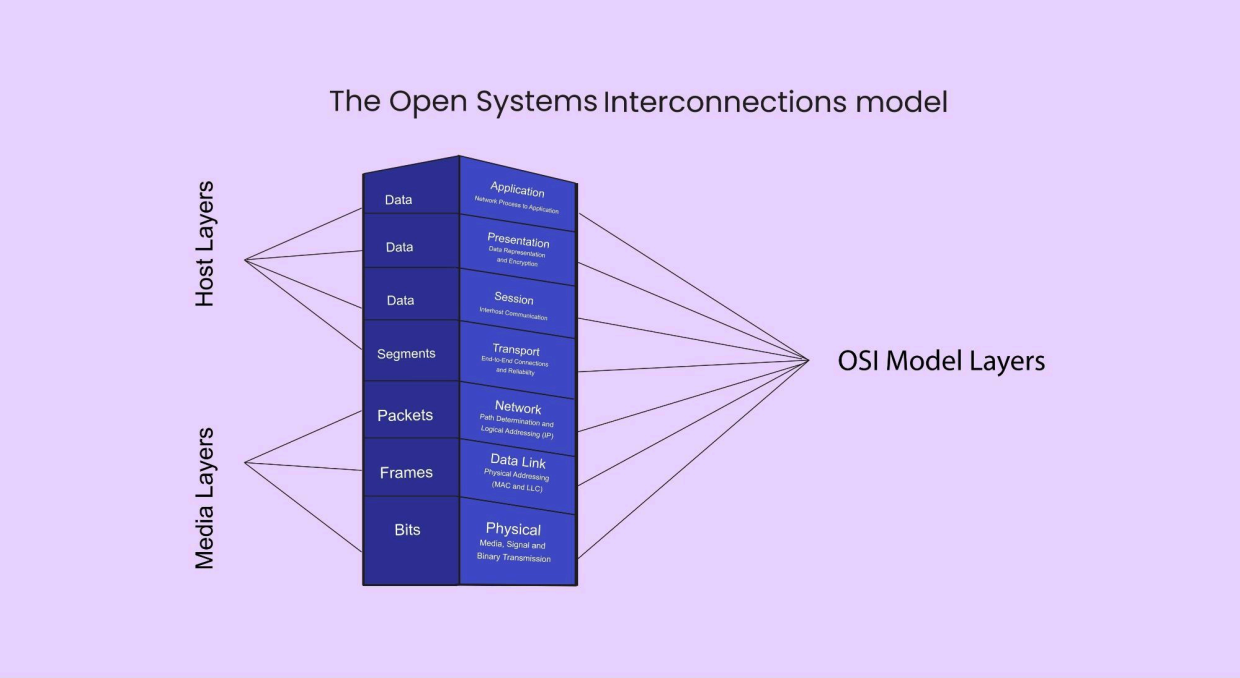
3. Architecture Thinking

How does architecture-first thinking change the role of a developer in AIDD?

Architecture-first thinking means thinking beyond just writing raw code; it means **designing the structure of the system**. The developer's role shifts to defining the components, their interactions, layers, and the division of responsibilities. Since AI tools handle the implementation details and take care of most of the coding, the developer is pivoted into a more **strategic position** rather than just manual coding.

Explain why developers must think in layers and systems instead of raw code.

Developers must think in **layers and systems** because this structure makes the project **scalable, maintainable, and easier for AI agents** to work through. Each layer has a clear, defined purpose, ensuring a **separation of concerns** and predictable behavior. Focusing purely on raw code often leads to a tangled, unmanageable structure that is hard to extend.



Shutterstock

Part B: Practical Task

Generate a 1-Paragraph Specification for an Email Validation Function

Requirements: Must contain "@" symbol, have a valid domain (e.g., .com, .org), and return clear error messages.

1️⃣ Your exact CLI prompt (Text)

Generate a one-paragraph specification for an email validation function. Requirements: must include "@", must have a valid domain like .com or .org, and return clear error messages for

invalid input.

2️⃣ The 1-paragraph specification generated by the CLI

The email validation function will accept a string as input and determine if it represents a valid email address. It must verify the presence of a single "**@**" symbol separating the local part from the domain. The domain part must contain at least one dot ("**.**") and end with a **Top-Level Domain (TLD)** consisting of at least two alphabetic characters. The function will return a boolean indicating validity and, if invalid, a specific error message detailing the reason for failure (e.g., "Missing '@' symbol", "Invalid domain format", "TLD too short").

Part C: Multiple Choice Questions

1. What is the main purpose of Spec-Driven Development?

- A. Make coding faster
- **B. Clear requirements before coding begins**
- C. Remove developers
- D. Avoid documentation

Option : **B. Clear requirements before coding begins**

2. What is the biggest mindset shift in AI-Driven Development?

- A. Writing more code manually

- **B. Thinking in systems and clear instructions**
- C. Memorizing more syntax
- D. Working without any tools

Option : **B. Thinking in systems and clear instructions**

3. Biggest failure of Vibe Coding?

- A. AI stops responding
- **B. Architecture becomes hard to extend**
- C. Code runs slow
- D. Fewer comments written

Option : **B. Architecture becomes hard to extend**

4. Main advantage of using AI CLI agents (like Gemini CLI)?

- A. They replace the developer completely
- **B. Handle repetitive tasks so dev focuses on design & problem-solving**
- C. Make coding faster but less reliable
- D. Make coding optional

Option : **B. Handle repetitive tasks so dev focuses on design & problem-solving**

5. What defines an M-Shaped Developer?

- A. Knows little about everything
- B. Deep in only one field
- **C. Deep skills in multiple related domains**
- D. Works without AI tools

Option : **C. Deep skills in multiple related domains**

Reflection

This task highlights how **AI-Driven Development (AIDD)** fundamentally elevates the developer's role from a coder to a **system designer**. The discipline enforced by concepts like Development Agents and Specification-Driven Development encourages thinking in **systems instead of syntax**. This shift enables professionals to scale as **M-shaped professionals** with strong competencies in several areas, driven by intelligent automation, rather than being limited by manual labor.