

## Task #2

### Part A

#### Nine Pillars Understanding

**Q1: Why is using AI Development Agents (like Gemini CLI) for repetitive setup tasks better for your growth as a system architect?**

Ans: I think this is because it saves a lot of time for developers and helps them focus on the important, high-level parts of the system. They don't need to write the same code again, and they don't have to remember every small syntax rule. For example, when a web developer makes an app, they must create all the files by hand, fix small errors, and set up everything on their own. This can even take a whole month. But an AI Development Agent can do the same setup in just a few hours, and it usually makes the structure cleaner and more organized. This way, the developer can use their energy to plan the right infrastructure and design a good-quality app, while the AI handles the repetitive work.

**Q2: Explain how the Nine Pillars of AIDD help a developer grow into an M-Shaped Developer?**

Ans: The M-Shaped Developer is someone who maintains deep expertise across multiple complementary domains. The Nine Pillars of AIDD help a developer grow into an M-Shaped Developer because AIDD removes the burden of writing every piece of code manually. This means the developer does not spend all their time thinking about syntax or repetitive tasks.

Instead, they focus on understanding the problem, planning the right solution, and thinking across multiple areas like architecture, workflows, data flow, AI tools, and system design. By using AIDD, a developer learns to think more broadly and deeply at the same time, which is exactly what makes them an M-Shaped Developer.

# Vibe Coding vs Specification-Driven Development

## **Q: Why does Vibe Coding usually create problems after one week?**

Ans # Vibe Coding initially works smoothly because there is no complexity. But when we add something complex, it creates problems because complex work needs more thinking and planning. For example, we made a simple e-commerce website, and it ran perfectly. But when we added more features like Stripe or tracking using APIs, it became complex, caused conflicts, and created errors.

## **Q # How would Specification-Driven Development prevent those problems?**

Ans # In specification-driven development, the developer gives strict specifications about the architecture and structure. Step by step specifications make problem solving easier. Therefore, we can say that specification-driven development helps solve the problems that appear in Vibe Coding.

# Architecture Thinking

## **Q: How does architecture first thinking change the role of a developer in AIDD?**

Ans: Even though AI can generate code, it can't think like a human. AI cannot replace human thinking, planning, and problem solving. In architecture-first thinking, the developer designs the structure and decides step by step how to solve the problem. This changes the

role of the developer from just writing code to planning the system, guiding AI, and making high-level decisions. Using this approach, developers get better results, avoid errors, and create well-structured, scalable solutions.

**Q: Explain why developers must think of layers and systems instead of raw code.**

A # Because in complex tasks, writing only raw code can cause conflicts and errors. Thinking in layers and systems helps the developer organize the work step by step, reduce complexity, and prevent mistakes. It also makes the code easier to maintain, scale, and update. By focusing on systems, developers can see the big picture, plan better, and guide AI tools effectively.

## **Part B**

### **Task:**

**Using any AI CLI tool, generate a 1-paragraph specification for an email validation function.**

**Requirements:**

**Must contain “@”**

**Must contain a valid domain (e.g., .com, .org)**

**Should return clear error messages**

### **Submitted screenshot**



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

```
> /model

[Gemini CLI update available! 0.15.3 → 0.15.4
 Installed with npm. Attempting to automatically update now...]

[Update successful! The new version will be used on your next run.

> generate a 1-paragraph specification for an email validation function.its is specify that it must contain "@" , a valid domain (e.g., .com, .org) and
Should return clear error messages
+
The email validation function should accept a string as input and return a boolean indicating whether the string is a valid email address, along with a
clear error message if validation fails. A valid email address must contain a single "@" symbol separating the local part from the domain, and the domain
must include at least one dot (e.g., ".com", ".org") with characters before and after it, ensuring it's not just "@" or "@.". The function should provide
specific error messages for common validation failures, such as missing "@", invalid domain format, or empty input.]
```

## 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

### **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

**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

**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

**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
- C. Deep skills in multiple related domains**