AI-Driven Development Insights
==============================

AIDD 30-Day Challenge | Made by: Asma Masood | Date: 19-Nov-2025

====================

Part A — Reflections

1. Nine Pillars Understanding

(a) Using AI Development Agents
When I allow AI agents to handle routine setup tasks like folders, configs, and boilerplates, I free myself to focus on system architecture, data flow, and structural decisions. AI handles busywork, helping me grow as a true system architect.

(b) Nine Pillars & M-Shaped Developer
Consistent use of the Nine Pillars develops strong skills across multiple areas, creating a versatile M-Shaped Developer with depth in several connected fields.

2. Vibe Coding vs Spec-Driven Development

(a) Vibe Coding Issues
Exciting at first, Vibe Coding soon causes confusion: lack of notes, unclear logic, and unstable architecture make both human and AI collaboration hard.

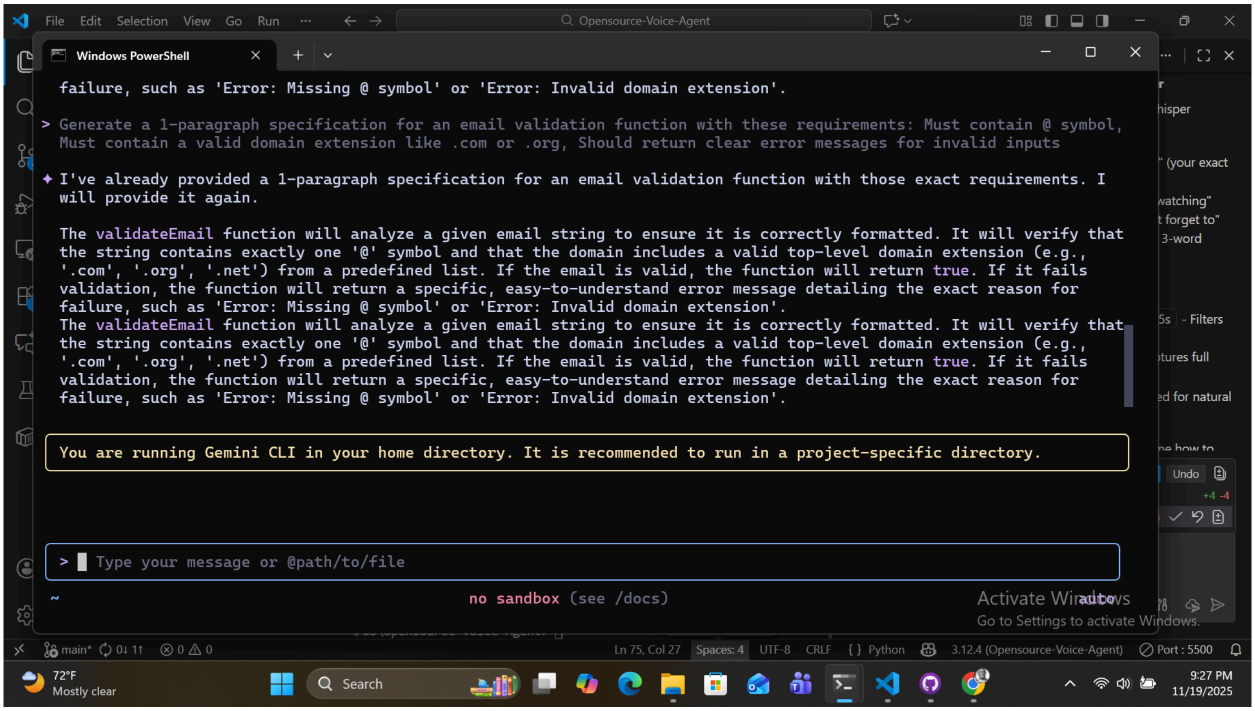
(b) Spec-Driven Development Solution
By defining purpose and logic first, SDD ensures clear direction, cleaner code, maintainability, and predictable outcomes.

3. Architecture Thinking

(a) Role Shift
Architecture-first thinking transforms developers into system thinkers. Responsibilities shift from coding lines to guiding modules and AI agents.

(b) Layered Thinking
Modern software is layered. Thinking in layers organizes responsibilities, scales applications, and improves AI collaboration.

Part B — Image / Diagram



```
failure, such as 'Error: Missing @ symbol' or 'Error: Invalid domain extension'.

> Generate a 1-paragraph specification for an email validation function with these requirements: Must contain @ symbol,
  Must contain a valid domain extension like .com or .org, Should return clear error messages for invalid inputs

✦ I've already provided a 1-paragraph specification for an email validation function with those exact requirements. I
  will provide it again.

  The validateEmail function will analyze a given email string to ensure it is correctly formatted. It will verify that
  the string contains exactly one '@' symbol and that the domain includes a valid top-level domain extension (e.g.,
  '.com', '.org', '.net') from a predefined list. If the email is valid, the function will return true. If it fails
  validation, the function will return a specific, easy-to-understand error message detailing the exact reason for
  failure, such as 'Error: Missing @ symbol' or 'Error: Invalid domain extension'.
  The validateEmail function will analyze a given email string to ensure it is correctly formatted. It will verify that
  the string contains exactly one '@' symbol and that the domain includes a valid top-level domain extension (e.g.,
  '.com', '.org', '.net') from a predefined list. If the email is valid, the function will return true. If it fails
  validation, the function will return a specific, easy-to-understand error message detailing the exact reason for
  failure, such as 'Error: Missing @ symbol' or 'Error: Invalid domain extension'.

 You are running Gemini CLI in your home directory. It is recommended to run in a project-specific directory.

 >  Type your message or @path/to/file

~                         no sandbox (see /docs)
```

File Edit Selection View Go Run
Opensource-Voice-Agent
Windows PowerShell
Activate Windows
Go to Settings to activate Windows.
main* 0↓ 11 ⊗ 0 ⚠ 0
Ln 75, Col 27 Spaces: 4 UTF-8 CRLF { } Python 3.12.4 (Opensource-Voice-Agent) Port : 5500
72°F
Mostly clear
Search
9:27 PM
11/19/2025

Part C — Multiple Choice Questions

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

B. Clear requirements before coding begins

2. Biggest mindset shift in AI-Driven Development?

B. Thinking in systems and clear instructions

3. Biggest failure of Vibe Coding?

B. Architecture becomes hard to extend

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

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

5. What defines an M-Shaped Developer?

C. Deep skills in multiple related domains