AI-Driven Development Task 2

Part A Theory (Short Answers)

1. Nine Pillars Understanding

a) **Why is using AI Development Agents better for your growth as a system architect?**
AI Development Agents handle repetitive setup, configuration, and boilerplate tasks, which saves mental energy and time. This lets me focus on architecture, structure, problem-solving, and system thinking instead of doing small routine steps. Over time, this builds my mindset as a system architect rather than just a coder.

b) **How do the Nine Pillars help a developer become an M-Shaped Developer?**
The Nine Pillars combine testing, specs, agents, workflows, evaluation, and automation into one ecosystem. Because of this, a single developer works across multiple areas—architecture, testing, automation, documentation, and planning—eventually developing deep skills in multiple fields. This is what makes a developer M-Shaped.

2. Vibe Coding vs Specification-Driven Development

a) **Why does Vibe Coding create problems after one week?**
Vibe coding has no clear plan or structure. You write whatever feels right in the moment, so after a week the code becomes messy, inconsistent, impossible to extend, and very hard to debug.

b) **How would Specification-Driven Development prevent those problems?**
SDD defines clear, written, executable instructions before coding starts. These specs keep the structure consistent, avoid confusion, make the system easy to maintain, and ensure everyone follows the same rules.

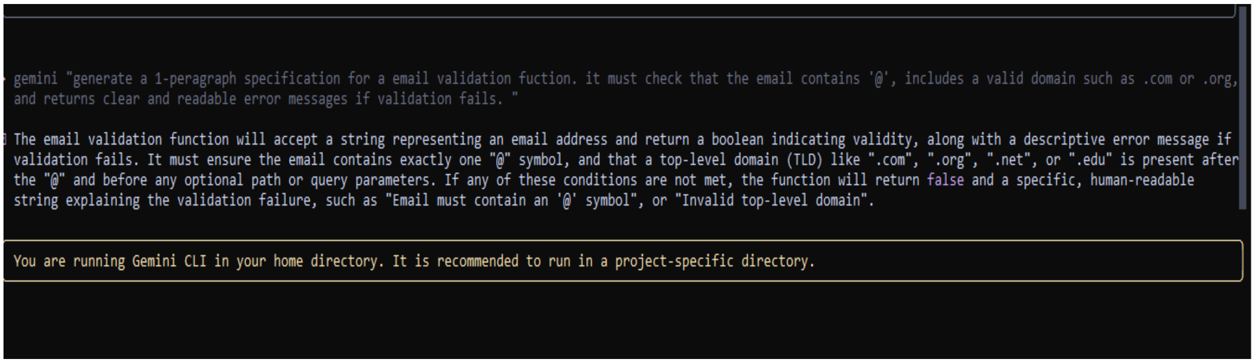
3. Architecture Thinking

a) **How does architecture-first thinking change the role of a developer in AIDD?**
Architecture-first thinking turns the developer into a planner and system designer.

Instead of only writing code, the developer now shapes the system, defines components, creates specs, and works with AI agents to implement solutions.

b) Why must developers think in layers and systems instead of raw code?
Layers and systems ensure scalability, clarity, and reusability. Raw code is isolated and hard to maintain, but layered architecture helps AI and humans understand the structure, extend the project, and fix issues easily.

Part B Practical Task



```
gemini "generate a 1-peragraph specification for a email validation fuction. it must check that the email contains '@', includes a valid domain such as .com or .org,
and returns clear and readable error messages if validation fails. "

The email validation function will accept a string representing an email address and return a boolean indicating validity, along with a descriptive error message if
validation fails. It must ensure the email contains exactly one "@" symbol, and that a top-level domain (TLD) like ".com", ".org", ".net", or ".edu" is present after
the "@" and before any optional path or query parameters. If any of these conditions are not met, the function will return false and a specific, human-readable
string explaining the validation failure, such as "Email must contain an '@' symbol", or "Invalid top-level domain".

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

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

Answer 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

Answer 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

Answer 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

Answer 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

Answer C) Deep skills in multiple related domains

Reflection

This task helped me understand that modern development is no longer about jumping straight into code. Architecture, specifications, and AI agents shape the direction before a single line is written. The real skill now is to think like a system designer and let AI help with execution. This mindset shift is what truly defines the AI-Native developer.

Name: MOIZ QURESHI QURESHI

Roll No: 00307188