Application Domain (DA):
The degree of the team's knowledge and familiarity with the areas or types of technology directly related to the project's objectives, such as Cloud, Artificial Intelligence, Mobile, or Web. The more team members possess relevant expertise or experience in these domains, the greater the team's ability to meet project needs, maintain delivery continuity, and adapt to changes.

Application Domain	
Value	Description
VL	**POOR** **Characteristics:** ● **Critical Gaps:** The team has no practical knowledge of the required technologies. Progress is nearly impossible without external training. ● **Quality Issues:** No understanding of how the technology impacts project outcomes, resulting in poor-quality deliverables that fail to meet industry standards. ● **Unacceptable Outcomes:** Low maintainability, frequent defects, and rework. **Example:** For a Cloud-based AI project, the team has no experience with cloud platforms or AI frameworks, leading to poor implementation and inability to meet functional requirements
L	**BELOW AVERAGE** **Characteristics:** ● **Minimal Knowledge:** Basic familiarity with at least one required technology, but with significant gaps in critical areas. ● **Inconsistent Quality:** The team struggles with implementing best practices (e.g., scalable architecture or optimization) due to lack of expertise. ● **High Risk:** Deliverables are prone to defects, technical debt accumulates, and the overall product is below acceptable quality benchmarks. **Example:** For a Web project requiring front-end performance optimization using React, the team has only rudimentary React skills, producing slow, buggy interfaces.

M	**AVERAGE** **Characteristics:** ● **Adequate Understanding:** The team has moderate expertise in most required technologies, with minor gaps in specialized areas. ● **Satisfactory Quality:** The team produces functional deliverables that meet minimum project standards, but improvements in maintainability and scalability are needed. ● **Occasional Rework:** Missing advanced knowledge may lead to inefficiencies or errors in complex scenarios. **Example:** In an IoT project requiring MQTT protocols, the team can implement basic functionality but struggles with ensuring robust device communication.
H	**ABOVE AVERAGE** **Characteristics:** ● **Strong Proficiency:** The team has advanced expertise in most required technologies, with only minor gaps. ● **High-Quality Output:** The team applies best practices (e.g., design patterns, optimization, and scalable architecture), delivering solutions that exceed quality standards. ● **Efficient Delivery:** Minimal rework is required, and deliverables are well-documented, maintainable, and aligned with industry benchmarks. **Example:** For a Cloud project requiring AWS Lambda, the team delivers serverless architectures with highly optimized and scalable implementations.
VH	**PERFECT** **Characteristics:** ● **Comprehensive Expertise:** The team demonstrates mastery in all required technologies. ● **Exceptional Quality:** Deliverables are robust, scalable, and adhere to the highest industry standards. The team anticipates challenges and proactively ensures long-term maintainability. ● **Innovation-Driven:** The team sets benchmarks for quality, often exceeding stakeholder expectations. **Example:**

	For a Distributed Systems project requiring Kubernetes, Python, and Kafka, the team delivers high-performance systems with seamless integration and fault tolerance.