Programming Languages (LP):
The team's proficiency in the programming languages required for the project. The higher the proportion of members with knowledge in each required language, the better the team's ability to efficiently divide tasks, minimize bottlenecks, and ensure code quality—even in situations involving the absence of a member or redistribution of responsibilities.

Programming Languages	
Value	Description
VL	**POOR** **Characteristics:** ● **No Proficiency:** The team has no practical knowledge of required languages, resulting in non-functional or error-prone deliverables. ● **Low Code Quality:** Code is unoptimized, poorly written, and fails to meet standards for readability or maintainability. **Example:** For a project requiring Python and Go, the team produces poorly written scripts that fail to execute correctly.
L	**BELOW AVERAGE** **Characteristics:** ● **Limited Knowledge:** The team has basic skills in only one or two required languages. ● **Inconsistent Quality:** Code is functional but unoptimized, with frequent bugs and poor documentation. **Example:** For a Web project requiring JavaScript and TypeScript, the team writes JavaScript but fails to leverage TypeScript's type-checking features.

M	**AVERAGE** **Characteristics:** ● **Proficient in Basics:** The team is proficient in most languages but may struggle with advanced implementations. ● **Good Quality:** Code meets acceptable quality standards but may lack advanced optimizations. **Example:** In a Backend project requiring Python and SQL, the team writes functional code but struggles with complex query optimizations.
H	**ABOVE AVERAGE** **Characteristics:** ● **Advanced Expertise:** The team demonstrates strong command over required languages. ● **High-Quality Code:** Deliverables are efficient, well-documented, and maintainable. **Example:** For a Microservices project requiring Java and Go, the team produces modular, well-structured code that supports scalability and maintainability.
VH	**PERFECT** **Characteristics:** ● **Complete Mastery:** The team writes optimized, production-grade code in all required languages. ● **Exemplary Quality:** Code is a benchmark for maintainability, scalability, and performance. **Example:** For a Distributed Systems project requiring Python, Go, and C++, the team delivers cutting-edge solutions with minimal debugging and rework.