1.6 Standards

To ensure the reliability, maintainability, and scalability of the CodeCodez system, a variety of **industry-standard software development and AI/ML best practices** have been adopted. These standards serve as guiding principles for the architecture, code generation, documentation, testing, and interoperability of the platform. This section outlines the relevant international and de facto standards being followed in the project.

1. Programming Standards and Style Guides

- **PEP8 (Python Enhancement Proposal 8):**
  Python-based modules generated by the system adhere to the PEP8 style guide, ensuring uniformity in indentation, naming conventions, line length, and code readability.
- **Airbnb JavaScript Style Guide:**
  For frontend development (React/JavaScript), the system ensures conformity to the Airbnb style guide, promoting clean syntax, variable scope control, and error handling consistency.
- **ESLint and Prettier Integration:**
  Automated formatters and linters are used for both Python and JavaScript/TypeScript to enforce coding standards at generation time.

2. Software Engineering Standards

- **IEEE 830: Software Requirements Specification Standard:**
  All user-defined input specifications are internally converted into structured formats that conform to SRS principles outlined in IEEE 830.
- **IEEE 1016: Software Design Description (SDD):**
  Modular code is accompanied by markdown-based auto-generated design documents describing architecture, logic flow, inputs/outputs, and dependencies.
- **IEEE 829 / ISO/IEC/IEEE 29119: Testing Standards:**
  The automated test generation process aligns with these standards by producing unit, integration, and regression tests along with coverage validation.