

Assignment 3

Task - 4

Diagramming Environment

Choice of Environment: PlantUML

- **Comprehensive Features:** It offers a comprehensive set of features for modelling, including support for class relationships, attributes, methods, and annotations.
- **Text-Based Representation:** PlantUML uses a simple text-based syntax, which is easy to write and version control using plain text files.
- **Integration with Markdown:** Integration with Markdown allows seamless inclusion of state chart diagrams in the documentation.
- **Open Source:** Being an open-source tool, PlantUML is accessible and easily integrated into various environments.
- **Code Generation Capability:** PlantUML is not inherently designed for code generation but primarily focuses on visual representation.
- **Observations:** No code is directly generated from PlantUML Charts. Instead, these charts serve as a visual aid for understanding and communicating the system's behaviour
- **Weaknesses:** The lack of direct code generation is not a weakness but a design choice. The expectation is to use PlantUML Charts as a documentation and communication tool rather than a code generation tool.
- **Differences:** There are no expectations for code generation from PlantUML, so there are no differences in this context.
- **Intended Use:** PlantUML is intended for visual representation and communication, and any disparities are more likely related to misunderstandings or misinterpretations during the communication process rather than code generation issues.