DIGITAL TWIN (SIEMENS NX + TIA PORTAL) & FMU /FMI

1. INTRODUCTION

This document specifies the structure and mandatory content for the DIT (Digital Twinning) course report. The report must clearly address two major deliverables: (1) the development, integration, and testing of a Digital Twin using Siemens NX and TIA Portal, and (2) the development and validation of a Functional Mock-up Unit (FMU) compliant with the FMI standard.

The report requires a clear separation between the Group Assignment (system-level) and the Individual Assignment (unit-level).

Objective: The goal is to produce a focused, readable document that demonstrates comprehensive coverage of all technical aspects. Clarity, disciplined modelling, structured code, and thorough testing are essential and will be reflected in the final evaluation.

Note on Submission: The overall deliverable will consist of two documents: a Group Report covering sections 2, 4.3 (Integration), and the System-Level Testing, and an Individual Report covering sections 3, 4 (FMU), and the Unit-Level Testing. The content of both reports must be mutually consistent and traceable.

2. GROUP ASSIGNMENT REQUIREMENTS (SYSTEM-LEVEL)

2.1 PROJECT BACKGROUND AND SCOPE

- Describe the industrial process or automation scenario being modelled.
- Motivate and justify the necessity of decomposing the overall process into discrete Units (subsystems) to manage complexity.
- State the system boundaries, stakeholders, assumptions, and constraints.
- Include a concise problem statement and measurable objectives.

2.2 SYSTEM-LEVEL FUNCTIONAL REQUIREMENTS

- Define comprehensive **functional (FR)** and **non-functional (NFR)** requirements for the entire system.
- Requirements must be **uniquely identified**, **testable**, and supported by a clear rationale.
- Coverage must include performance, reliability, safety, interoperability (e.g., PLC interfaces, HMIs), and data exchange.
- The report must include documentation (e.g., a table) illustrating the **traceability** linking requirements to design elements and subsequent tests.

2.3 SYSTEM ARCHITECTURE & PROCESS DESCRIPTION (MANDATORY DIAGRAMS)

- Present a high-level system architecture showing the interaction between the NX Digital Twin (mechanical/kinematic model), TIA Portal (PLC logic, I/O mapping), and communication links.
- Mandatory Diagrams: The explanation of the system, its components, and the overall process must be supported by diagrams based on relevant standards (e.g., ISA-88 for process hierarchy, UML/SysML for component structure or behavior).
- Hardware Architecture: Include diagrams describing the physical Hardware Architecture (control devices, sensors, actuators, network topology).

- Process Description: Provide diagrams detailing the process flow or state changes (e.g., using a state diagram or ISA-88 Procedure/Process Model).
- Define interface contracts, I/O lists, and communication protocols.

2.4 GROUP TESTING & VALIDATION

- Plan and report comprehensive System Integration Tests that verify all the group's system-level functional requirements.
- Define clear test procedures, expected results, pass/fail criteria, and sufficient evidence (screenshots, logs).
- Testing must cover nominal operation, edge cases, safety interlocks, and failure handling.
- Summarize test outcomes.

---

3. INDIVIDUAL ASSIGNMENT REQUIREMENTS (UNIT-LEVEL)

3.1 UNIT SCOPE & REQUIREMENTS

- Each student must clearly define their assigned Unit/Use-Case from the group's decomposition.
- Derived Requirements: Derive Unit-Specific Requirements from the group's system-level requirements, maintaining traceability.
- Mandatory Unit Diagram: The unit's structure and function must be explained using a diagram related to the chosen standard (ISA-88 Unit Module, UML Component/Class Diagram, etc.).
- Document the unit's specific role, interfaces (inputs/outputs), and constraints.
- Provide a unit requirement specification with unique IDs and clear testability criteria.

3.2 NX DIGITAL TWIN DEVELOPMENT (UNIT)

- Explain the detailed modelling approach used in Siemens NX for the specific unit.
- Clearly specify the definition and application of the following elements in the simulation model:
    - Rigid Bodies, Mass properties, and Material properties.
    - Joints (e.g., Sliding Joints, Hinges) and associated kinematic chains.
    - Constraints and mechanical limits.
    - Collision Bodies and contact settings.
- Include annotated screenshots, model hierarchy, and relevant parameter tables.
- State any simplifications made and provide justification (e.g., why certain bodies are rigid).

3.3 CONTROL/AUTOMATION CODE (UNIT)

- Present the well-structured control code (PLC, SCL, structured text) developed for the unit.
- Structured Code: Demonstrate adherence to good programming standards (modularity, clear naming, effective commenting).
- The code structure must be explicitly aligned with supporting diagrams (e.g., state machines, Process Hardware diagrams). *Alignment with diagrams is critical and will be reflected in assessment.*

3.4 UNIT TESTING & VERIFICATION

- Define comprehensive **Individual Test Cases** that specifically cover **all unit requirements** (Section 3.1).
- Include clear input conditions, expected outputs, and pass/fail criteria.
- Provide detailed **test evidence** (simulation captures, logs) and summarize any defect fixes implemented.
- Ensure complete **traceability** from requirements to tests and results.

---

4. FMU DEVELOPMENT (FMI STANDARD)

Recommended to only watch to implement the FMI 2.0.

Don't forget to add references to the sources

4.1 TECHNICAL BACKGROUND

- Provide a clear and detailed technical explanation of the FMU concept under the FMI standard.
- Describe the fundamental difference and application of Model Exchange versus Co-Simulation.
- Explain the structure of the FMU (e.g., modelDescription.xml, binaries, resources).

4.2 FMU IMPLEMENTATION & STAND-ALONE TESTING

- Document how the FMU was developed (tools/languages used, interface variables, parameters).
- Describe the model that has been implemented
- Test the FMU model.
- The developed FMU must be tested stand-alone.

4.3 FMU INTEGRATION & PORTABILITY

- Describe potential integration pathways for the FMU within the group's Digital Twin system.
- Mandatory Requirement: Demonstrate the FMU's export capability to at least one external FMI 2.0-compliant environment (e.g., a different simulation tool). Provide clear evidence (screenshots/logs) to validate this portability.

---

5. REPORT STRUCTURE & WRITING GUIDELINES

While there is no fixed page limit, the document must be easily readable, well-structured, and focused. Avoid including irrelevant details or simply dumping large files of data without analysis.

The document must be easily readable, well-structured, and focused. Avoid irrelevant details or simply dumping large files of data without analysis.

Section No.	Report Section Heading	Responsibility
Title Page	Title, Authors, and Unit Assignments	Group
1.	Introduction	Group
2.	Project Background & Scope (Group)	Group
3.	System-Level Requirements & Architecture (Group)	Group
4.	Unit Specification & NX Modelling Details (Individual: Unit)	Individual
5.	Control/Automation Code Overview (Individual: Unit)	Individual
6.	Testing: Comprehensive Coverage	Mixed
	6.1 Unit Testing & Verification (Individual: Unit)	Individual
	6.2 System Integration Testing & Validation (Group)	Group
7.	FMU Background, Implementation, Tests, and Integration (Individual: FMU)	Individual
8.	Conclusions & Lessons Learned	Group
9.	References/Appendices	Mixed