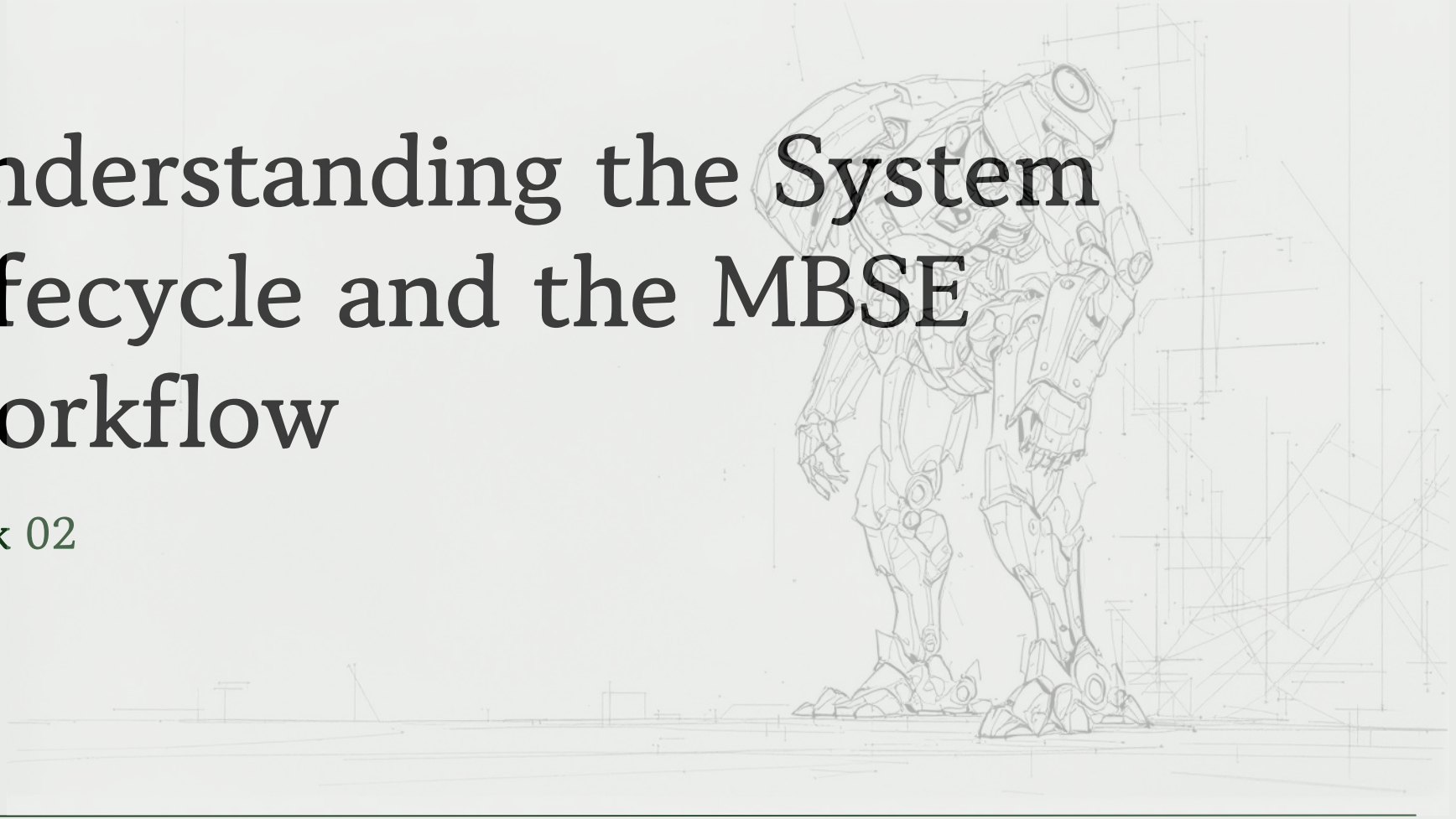
Understanding the System Lifecycle and the MBSE Workflow

Week 02



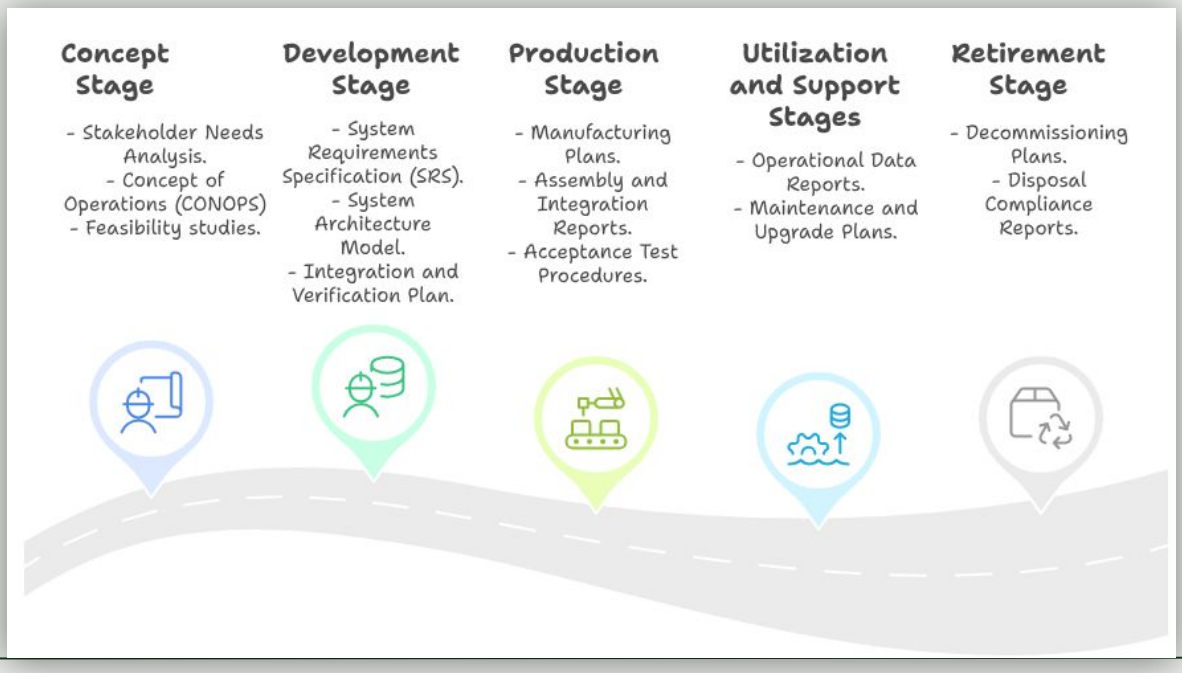
Review of Systems Engineering Lifecycle

Review of Systems Engineering Lifecycle (Expanded View): The Systems Engineering Lifecycle spans multiple interacting stages, each producing key deliverables and decisions that shape the system's success.

- **Concept Stage**: Define needs, explore solutions, assess feasibility.
- **Development Stage**: Define architecture, design components, integrate subsystems.
- **Production Stage**: Manufacture, assemble, and validate system units.
- **Utilization Stage**: Operate the system, collect performance data, adapt to needs.
- **Support Stage**: Maintain, upgrade, and sustain system capabilities.
- **Retirement Stage**: Decommission system responsibly and manage transitions.

Key Lifecycle Artifacts and Milestones

Each stage of the Systems Engineering Lifecycle produces critical artifacts and passes major decision milestones.



ConOps vs OpsCon

The Concept Phase: Define the Why and How of the System
Every system starts with a purpose. The Concept Phase focuses on **understanding the mission**, the **users**, and the **operational context** — before any technical design begins.

ConOps (Concept of Operations):

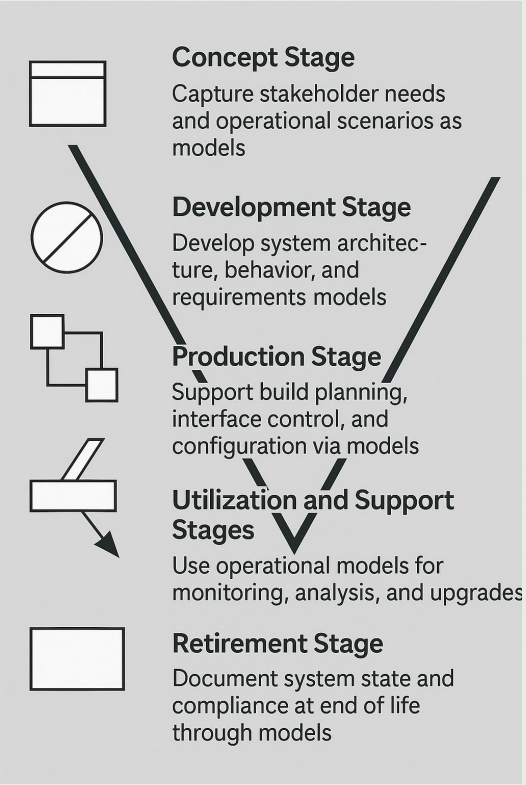
- High-level vision: *What is the system expected to do, and why?*
- Written from the **stakeholder perspective**
- Answers: Who uses it? What problems does it solve?

OpsCon (Operational Concept):

- Describes **how** the system will behave in its environment
- Includes **scenarios**, **interactions**, **external systems**, **constraints**
- Often modeled using **Use Cases** and **Operational Scenarios** in SysML

Both guide early **feasibility**, **trade studies**, and **stakeholder validation**

How MBSE Connects to the Lifecycle



MBSE activities are aligned with each Systems Engineering Lifecycle stage, supporting better consistency, traceability, and decision-making.

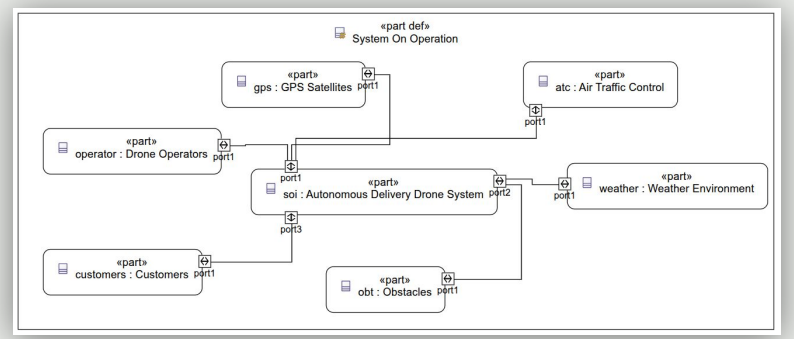
Throughout all phases, MBSE offers:

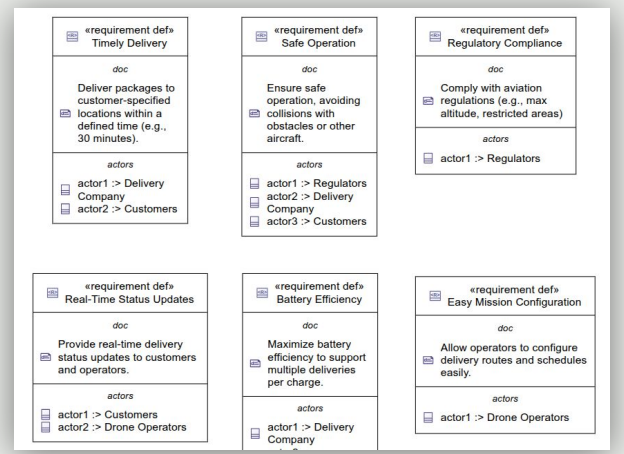
- **Consistency**: Models stay alive and updated across the lifecycle.
- **Traceability**: Changes and decisions are always linked to needs, requirements, and verification.
- **Collaboration**: Different teams work from a common source of truth.
- **Agility**: Rapid adaptation to new needs or conditions is easier when models are coherent.

MBSE in Concept Stage

During the Concept Stage, MBSE captures stakeholder needs, operational scenarios, and early system goals in structured, traceable models.

- Identify and model **stakeholders and their needs**.
- Define **operational use cases** and **mission scenarios**.
- Model early **system boundaries** and **interactions**.
- Support **trade studies** and **feasibility analysis** using models.
- Lay foundation for requirements refinement and system design.

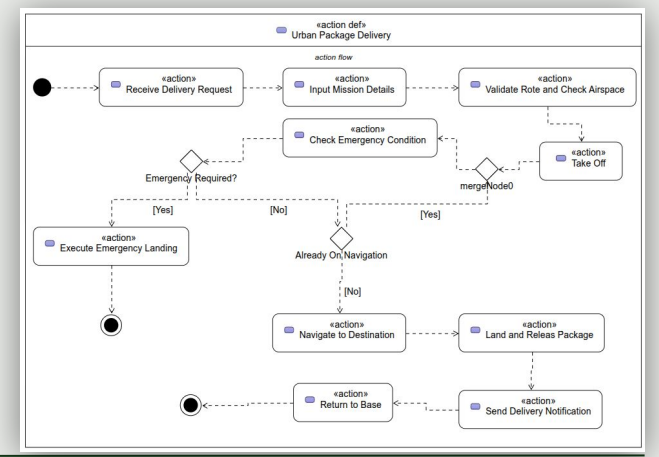




MBSE in Development Stage

During the Development Stage, MBSE defines system requirements, architecture, behavior, constraints, and verification models — forming the authoritative system description.

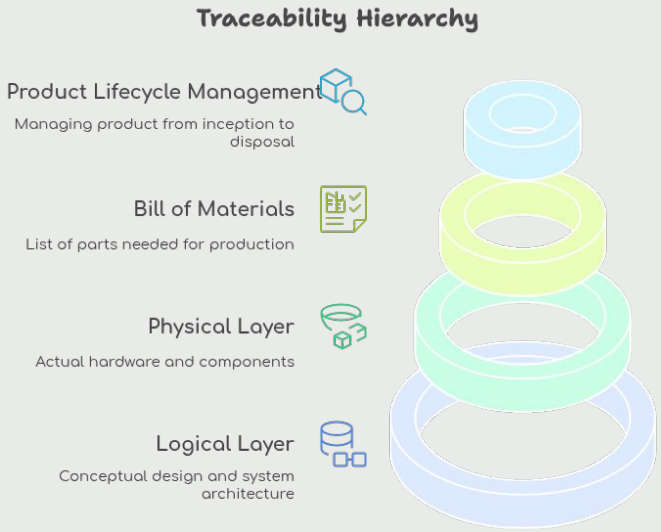
- Refine stakeholder needs into **formal system requirements**.
- Model **system architecture** (structure, interfaces, allocations).
- Model **system behavior** (use cases, activities, state transitions).
- Define **constraints and parameters** for engineering trade-offs.
- Plan and model **verification and validation** strategies.



MBSE in Production Stage

During the Production Stage, MBSE supports manufacturing, integration, configuration management, and production verification by maintaining authoritative models.

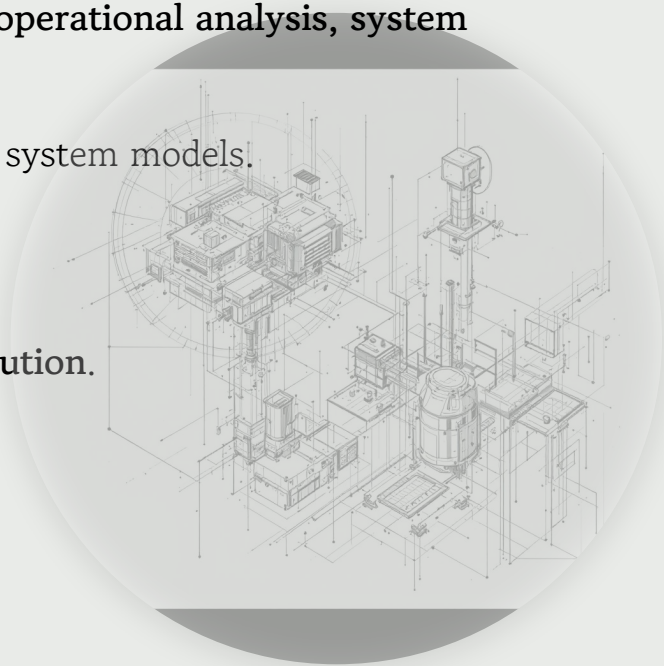
- Ensure **build-to-model** manufacturing consistency.
- Support **interface verification** and **assembly integration**.
- Manage **configuration baselines** through models.
- Assist in planning and executing **acceptance testing**.
- Reduce errors and rework through **model-driven validation**.



MBSE in Utilization and Support Stage

During Utilization and Support, MBSE provides the foundation for operational analysis, system upgrades, and maintenance management.

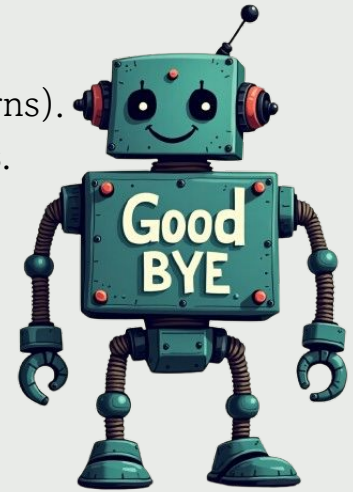
- Support **real-time monitoring** and **performance analysis** using system models.
- Drive **issue diagnosis** and **failure root-cause analysis**.
- Plan and validate **system upgrades and patches**.
- Enable **maintenance planning** through digital models.
- Extend system life and performance through **model-based evolution**.



MBSE in Retirement Stage

During the Retirement Stage, MBSE provides traceable system knowledge to support safe decommissioning, compliance, and transition activities.

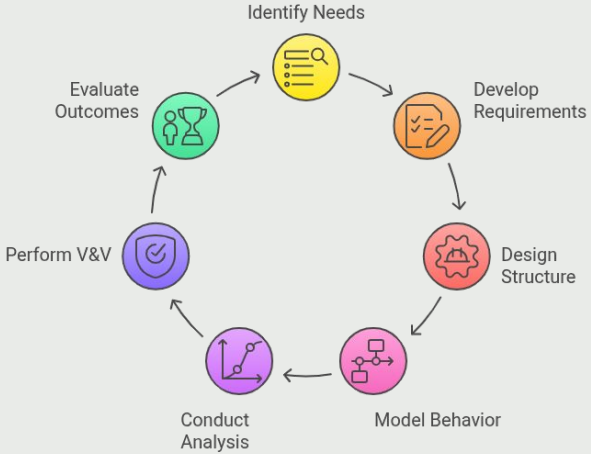
- Maintain **traceability of system configurations and changes**.
- Plan and model **safe system decommissioning procedures**.
- Provide **evidence for regulatory and environmental compliance**.
- Support **transition to successor systems** (reuse of architecture and design patterns).
- Preserve system knowledge for **future learning and lessons-learned repositories**.



Practical MBSE Workflow Overview

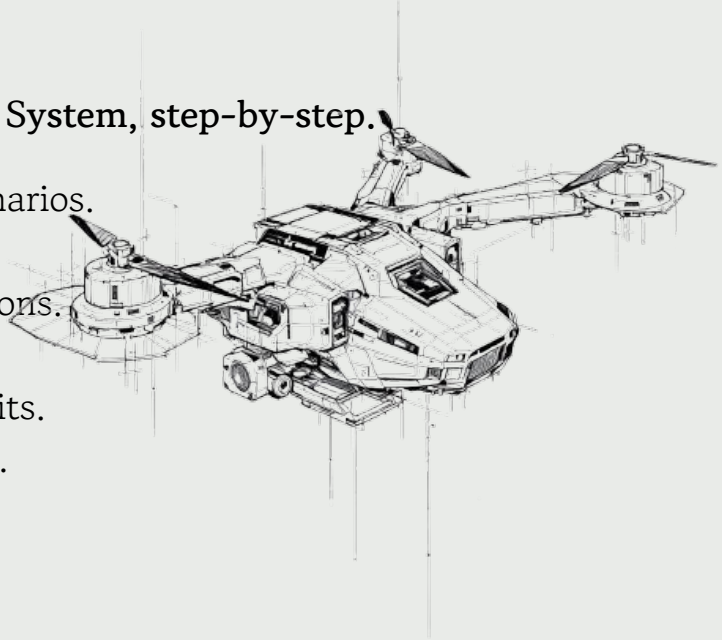
The MBSE Workflow connects system lifecycle stages into a continuous, structured modeling process.

- **Capture Stakeholder Needs** ➔ Model Needs and Concerns
- **Define System Requirements** ➔ Formalize and Trace
- **Model Architecture** ➔ Structure, Interfaces, Allocations
- **Model Behavior** ➔ Use Cases, Activities, States
- **Model Constraints and Parameters** ➔ Support Trade Studies
- **Plan Verification and Validation (V&V)** ➔ Model Test Cases
- **Manage Model Evolution** ➔ Updates and Configuration Control



Example - MBSE Workflow steps for Drone System

Applying the MBSE workflow to the Autonomous Delivery Drone System, step-by-step.



- **Capture Needs ➔** Model Stakeholder Needs and Mission Scenarios.
- **Define Requirements ➔** Model Formal System Requirements.
- **Architect the System ➔** Model Structure, Interfaces, Allocations.
- **Model Behavior ➔** Define Use Cases, Activities, and States.
- **Define Constraints ➔** Model Endurance, Payload, Safety Limits.
- **Plan V&V ➔** Model Test Cases for Requirements Satisfaction.
- **Evolve and Trace ➔** Maintain model updates across changes.

Introduction to MBSE Tools Overview

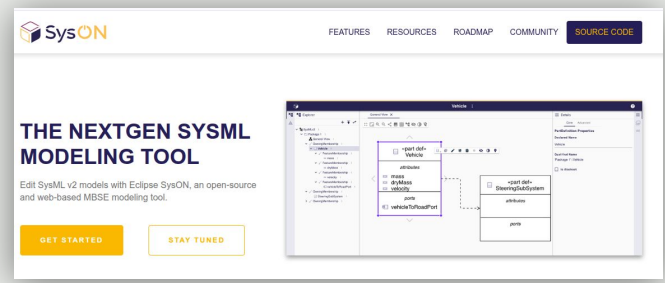
MBSE tools provide environments for building, managing, simulating, and analyzing system models using SysML v2.

- **Purpose of MBSE Tools**:
  - Create structured, consistent, machine-readable models.
  - Enforce SysML v2 language rules.
  - Support traceability, simulation, and validation.
- **Examples of MBSE Tools**:
  - **SysON** (Lightweight Open Source SysML v2 Web IDE) ✅ (Selected for this course)
  - MBSE IDE (SysML v2 Pilot Implementation)
  - CATIA Magic / Cameo Systems Modeler (Commercial MBSE tools)
- **Choosing a Tool**:
  - Depends on ease of use, SysML version support, collaboration needs, and project complexity.

SysOn Introduction

SysON is a lightweight, web-based open-source platform for creating and managing SysML v2 models.

- **Purpose**:
  - Provide an easy-to-use SysML v2 modeling environment.
  - Simplify setup using Docker-based deployment.
- **Key Features**:
  - Fully web-based interface (no manual installation required).
  - Supports SysML v2 textual and graphical editing.
  - Project creation, element editing, and model validation.
- **Ideal for**:
  - Students, educators, researchers, and MBSE beginners.
  - Rapid modeling workshops and hands-on learning.



https://mbse-syson.org/

SysOn Setup Instructions

Prepare your SysON modeling environment with two simple setup steps.

1. **Install Docker Engine**:

    - Download the Docker installer package.
    - Install Docker Desktop (for Windows/macOS) or Docker Engine (for Linux).

2. **Deploy SysON Using Docker Compose**:

    - Download the SysON Docker Compose YML file.
    - Launch SysON locally with a single command.
    - Access SysON through your web browser (localhost).

Summary of Week 2

This week, we take a step forward to deepen our understanding of MBSE and its relationship to the Systems Engineering Lifecycle. We will also cover an overview and setup of SysML v2, which we will use throughout this course.

- Understand the **full Systems Engineering Lifecycle** in depth.
- Learn **how MBSE supports each lifecycle stage**.
- Understand and apply the **MBSE Workflow** step-by-step.
- Become familiar with **SysON** for modeling SysML v2 systems.
- Successfully **install and verify the modeling tool** ready for Week 3.
- Conduct a **first simple hands-on exploration** of a system model inside SysON.

QUESTION!