# Software Engineering Concepts

**SE - Lab 02**

*Start date:* *31 October, 2025.*
*Due date:* *10 November, 2025.*

# Lab Overview

This lab is helps you understand the Prototyping phase within the Software Development Life Cycle (SDLC) by developing a simple attendance system using Java.

# Practical Example: Attendance System Using Java

Let's apply prototyping to develop an attendance system using Java:

## Prototyping Cycles for Attendance System

### Cycle 1: Initial Prototype

**Goal:** Create a basic attendance system with core functionality

**Initial Prototype Features:**

- Simple console-based interface (CLI)
- Add student to system (e.g.: attendance add student "Some Student")
- Mark attendance (Present/Absent)
- View attendance records

**Figma Design:**

- Create a Figma design for the attendance system interface

**User Feedback:**

- "Interface is too basic and hard to use, need a GUI"
- "Need to see attendance by date"
- "Want to calculate attendance percentage"

### Cycle 2: Enhanced Prototype

**Goal:** Improve user interface and add date tracking

**Enhanced Features:**

- JavaFx interface
- Date-based attendance tracking
- Attendance percentage calculation

- Better data organization

**Figma Design v2:**

**User Feedback:**

- "Great improvement! Love the JavaFx interface"
- "Need to generate reports for teachers"
- "Want to export data to files"
- "Add validation for invalid inputs"

### Cycle 3: Final Prototype

**Goal:** Add reporting features and data persistence

**Final Features:**

- Input validation
- Report generation
- File export functionality
- Error handling
- Student ID system

**Figma Design v3:**

**Final User Feedback:**

- "Perfect! This meets all our requirements"
- "The validation prevents errors"
- "Export feature is very useful for record keeping"
- "Ready for production use"

## Key Prototyping Benefits Demonstrated:

1. **Iterative Feedback:** Each cycle incorporated user suggestions
2. **Risk Reduction:** Early identification of user needs and technical challenges
3. **User Involvement:** Continuous user feedback shaped the final product
4. **Flexible Development:** Easy to modify and enhance features
5. **Early Validation:** Confirmed requirements before full development