## Project Title: *Python Digital Logic Simulator with Quartus Export*

**Scope:**
A desktop application written in Python that simulates basic digital circuits using logic gates (AND, OR, NOT, NAND, NOR, XOR, XNOR). Users can build circuits through a GUI, toggle inputs, and observe real-time outputs. The simulator will also generate truth tables automatically and include an export feature so that designed circuits can be tested on Quartus Prime.

### 1. Objectives

*Individual Objectives*

Member 1:
Implement Python logic gate classes (AND, OR, NOT, etc.).
Build the circuit evaluation engine.

Member 2:
Design the GUI in Python (Tkinter or PyQt) for drag-and-drop gates and wire connections.
Implement Verilog export functionality for Quartus.

Member 3:
Develop truth table generation.
Handle testing and validation of circuits.

*Team Objectives*

- Build a Python-based simulator with GUI support for circuit design.
- Implement core logic gates with real-time evaluation.
- Provide automatic truth table generation.
- Enable Verilog export for Quartus FPGA testing.
- Document the full project and manage code with GitHub.

### 2. Technologies and Tools

- Programming Language: Python
- Libraries:
- Quartus Prime: For testing and FPGA synthesis
- Version Control: GitHub
- Documentation: Markdown → PDF

### 3. Project Timeline

Week 1–2:

- Set up GitHub repo, finalize scope, research GUI frameworks.

Week 3–4:

- Implement gate classes and evaluation logic.
- Create a basic GUI layout with gate placement.

Week 5–6:

- Integrate backend logic with GUI.
- Add a truth table generator.

Week 7–8:

- Implement Verilog export feature.
- Test sample circuits in Quartus.

Week 9–10:

- Debug and polish GUI.
- Write documentation and finalize PDF.
- Upload final project to GitHub.