Our project uses the **Software Testing Life Cycle (STLC)** methodology to systematically evaluate multiple operating systems. STLC focuses entirely on testing rather than development, ensuring a structured and reliable process. STLC’s systematic approach allows defects to be detected and reviewed early to save both time and cost (BrowserStack, 2025), which we applied as follows:

**Requirement Analysis:** Aspects such as which evaluations need to be performed (e.g. IPv4 TCP Delay, IPv6 UDP Throughput), evaluation methods (e.g. scripting for automation), and requirements for each of the evaluations (e.g. PC hardware, Linux ISOs, software tools) were defined. A crucial part of our analysis was understanding how to automate the evaluation process.

**Test Planning:** This phase goes into further detail with analyse of evaluation tools, and development of evaluation documentation. At this phase, the team finalised the choice of using D-ITG over iPerf, as per the client’s request.

**Test Case Development:** Development of the evaluation process is based on the evaluation documentation outlined in the previous phase. As per the STLC methodology, this phase had multiple iterations for each operating system to ensure accurate and reliable data.

**Test Environment Setup:** This involves setting up hardware and software - in our case, a daisy chain network topology of 4 computers, each with a Linux-based operating system installed. This phase was iterated three times, once per OS.

**Test Execution:** The actual process of conducting the evaluation, where all 4 computers were connected, scripts were executed to send traffic from the sender to receiver via the routers, and data was recorded in logs.

**Test Closure:** This wraps up the evaluation, with logged data exported to a readable format, then compared and plotted. Finally, the data was presented to the client, and the poster was presented for marking.