# Work Breakdown Structure v3.0

**Prepared By:** Win Phyo & Thomas Robinson & Nathan Quai Hoi

**Date:** 28/05/2025

**Project Name:** Linux Network Performance Evaluation

1. Requirement Analysis
   1. Entry Criteria
      1. Project Brief document
      2. Client Requirements Identified
   2. Conduct Kick-off meetings
      1. Team Kick-off Meeting
      2. Client Kick-off Meeting
      3. Recording Meeting minutes
   3. Requirement Gathering
      1. Functional requirements
      2. Non-functional requirements
      3. Out-of-scope
   4. Feasibility Study
      1. Skills Analysis Matrix
   5. Develop Project Proposal
      1. Project Charter
      2. Scope Statement
      3. Stakeholder Management Strategy
      4. Risk Register
      5. Communication Management Plan
      6. Change Management Plan
      7. Initial Project Schedule & Milestones
      8. Estimated Costs
         1. Labour Breakdown
      9. Quality Assurance Plan
      10. Proposal Document Preparation
      11. Proposal Submission and Presentation
   6. Develop Requirement Traceability Matrix (RTM)
   7. Develop Tool Assessment
      1. Evaluation on iPerf and D-ITG
   8. **Milestone 1 – Project Proposal**
   9. Exit Criteria
      1. Completion of Requirement Traceability Matrix
      2. Client approval for project proposal and project charter
2. Test Planning
   1. Entry Criteria
      1. Approved Project Proposal
      2. Completion of Requirement Analysis
   2. Researching And Upskilling
      1. Research Test tools (D-ITG)
      2. Practice OS Routing Setup Practice
      3. Studying Protocol (IPv4, IPv6, TCP, UDP)
   3. Define evaluation environment setup requirements
   4. Develop Test Plan Document
      1. User Test Case Development
      2. Define Testing objectives
         1. Identify Hardware/Software/Network needs
         2. Specify OS version and configuration
         3. Document environment setup prerequisites
      3. Outline Data logging and validation approach
      4. Define Re-test and success/ failure criteria
      5. Map Requirements to test scenarios
      6. Assign Roles and Responsibilities (IPv4 and IPv6 Team)
      7. Develop Timeline and milestones
      8. Document Risk and Mitigation
   5. Develop QA Checklists
      1. Evaluation
   6. Review and finalise
   7. Exit Criteria
      1. Environment requirements clearly defined
      2. Test Plan Approved
3. Test Case Development
   1. Entry Criteria
      1. Completion of Test Plan document
   2. Define Detailed Test Cases
      1. Specify input parameters (protocols, packet sizes, rates, time duration, etc.)
      2. Complete detailed step-by-step on running evaluations for each test case
         1. Setting up OS
      3. Maintain Traceability Between Test Cases and Requirements
   3. Develop Test Scripts/ Configurations
      1. Bash scripts to automate using ITGSend for each test case
      2. Bash scripts to automate decoding the logs using ITGDec
      3. Scripts to automate logging average performance metrics data into Excel Spreadsheet
      4. Review and Validate Evaluation Test Cases
         1. Peer-review of test cases and scripts configuration
   4. Exit Criteria
      1. Test cases review and validated internally
      2. Client approval on defined test cases and scripts
4. Test Environment Setup
   1. Entry Criteria
      1. Test cases reviewed and validated
      2. Defined Environment Requirements
      3. All required hardware/software resources available
   2. Setting up OS and Network Topology for Each OS

*Note: Testing conducted sequentially on Ubuntu, Fedora, and Kali due to hardware constraints*

* + 1. Installing Linux OS distros (Ubuntu, Fedora, Kali) on routers
    2. Run OS updates
    3. Installing SSH server for connection between routers
    4. Assigning Static IP addresses and subnet masks for all machines
    5. Setting up routing tables for all machines
    6. Verifying connectivity (ping)
  1. Installing required software
     1. D-ITG installation
     2. Chrony NTP server Installation and configuration
  2. Deactivating/ Deleting unused software/package (to avoid interference)
  3. Validate Environment Setup
  4. Exit Criteria
     1. Fully functional evaluation environment

1. Test Execution
   1. Entry Criteria
      1. Fully functional evaluation environment setup
      2. Approved Test Plan and Test cases
   2. Execute Test Cases for each OS Sequentially (Ubuntu, Fedora, Kali)
      1. Run scripts for sending packets
      2. Log raw performance data
      3. Run scripts for decoding logs (ITGDec)
      4. Run scripts for exporting data into excel spreadsheets
   3. Validate Test Results
      1. Create visual graphs of the data
      2. Compare and analyse the test results
      3. Identify whether result fall inside of the determined confidence interval
   4. Re-evaluating (if needed)
      1. Updating scripts or configurations
      2. Re-run Test cases if fall outside of the determined confidence interval
   5. **Milestone 2** – Mid-term progress Review
      1. Status Report
      2. Team Portfolio
      3. Progress Presentation
   6. Exit Criteria
      * 1. All planned test cases executed for every OS
        2. All results documented and validated internally
        3. Client approval on test execution summary results
2. Test Closure
   1. Entry Criteria
      1. Completion of Test Execution
      2. All test data collected and logged
   2. Analysing All Results
      1. Compiling performance metrics for all test evaluations
      2. Comparing results between the OS.
   3. Test completion Report
      1. Summary of executed tests results and anomalies found
      2. Finalising Requirement Traceability Matrix
   4. **Milestone 3** - Test Closure Meeting
      1. Final Poster
      2. Presenting Testing Summary to Client and Mentor
      3. Client Approval for Test summary report
   5. Exit Criteria
      1. All test deliverables completed and submitted to client
      2. Document closure report
      3. Client feedback and review