Technical Report

Multi-Threaded Web Server with IPC and Semaphores

[Student Name 1] (ID: [XXXXX]) Student Name 2>Student Name 2 (ID: [YYYYY])

Operating Systems - University of Aveiro

December 5, 2025

Abstract

This report describes the implementation of a concurrent web server in C, utilizing a multi-process and multi-thread architecture. The system implements complex synchronization mechanisms (POSIX semaphores, mutexes, condition variables) and inter-process communication via shared memory. Technical details of each module, major concurrency challenges faced, and a quantitative performance analysis under load are presented.

Contents

1 Introduction

2 Implementation Details

2.1 System Architecture

2.2 Process and Thread Management

2.3 Synchronization and IPC

2.4 Resource Management (Cache and Files)

2.5 HTTP Processing

3 Challenges and Solutions

3.1 Race Conditions

3.2 Memory Management and Leaks

3.3 Log Synchronization

3.4 Zombies and Signals

4 Testing Methodology

4.1 Functional Tests

4.2 Concurrency and Stress Tests

5 Performance Analysis

5.1 Test Environment

5.2 Results: Throughput

Concurrency	Req/s (No Cache)	Req/s (With Cache)
10	1500	4000
100	2000	8500

Table 1: Throughput comparison with and without cache

5.3 Results: Latency

5.4 Discussion of Results

6 Conclusion