Instructions to run the code

1. Install the jsoncpp library for reading .json files in C++ using the command `sudo apt-get install libjsoncpp-dev` in ubuntu.
2. Edit the configurations.json file.
   Following are the parameters in the configurations.json:
   - Number_of_runs : Total runs in the simulation
   - Number_of_cores : Total cores in the web server
   - Number_of_threads : Threads count in the web server
   - Buffer_length : Maximum length of the request buffer
   - Scheduling_type: Scheduling algorithm type 1 - FCFS, 2- Round robin algorithm.
   - Quantum_time: Time slice in round robin scheduling algorithm
   - Context_switch_overhead: Overhead time for doing context switch from one process to another process
   - Number of users: Total users in the request-response loop
   - Total_delays: Total number of requests processed to stop the simulation in each run
   - Mean_service_time: Average service time of a request
   - Service_time_distribution: 1 - constant, 2 - uniform, 3 - exponential
   - Thinktime: Time a user waits to send a request after receiving a response or timeout
   - Thinktime_distribution: 1 - constant, 2 - uniform, 3 - exponential
   - Timeout_time: variable component of the timeout
   - Timeout_minimum: Minimum time out for each request
   - Timeout_distribution: 1 - constant, 2 - uniform, 3 - exponential
3. To compile the C++ program use the command `g++ -o main main.cpp -ljsoncpp`
4. To run the code use the command `./main`

Links of google sheets for Response time confidence intervals calculation

ResponsetimeM10

ResponsetimeM20

ResponsetimeM30

ResponseTimeM40

ResponsetimeM50

ResponsetimeM60

ResponsetimeM70

ResponsetimeM80

ResponsetimeM90

ResponsetimeM100

ResponsetimeM110

ResponsetimeM120

ResponsetimeM130

ResponsetimeM140

ResponsetimeM150

ResponsetimeM160

ResponsetimeM170

ResponsetimeM180

ResponsetimeM190

ResponsetimeM200