**1)Prerequisites:**  
a) Install updated version of **cmake** utility (> 3.5 – current newest version is 3.9.X):   
***$ sudo apt-get install cmake***

NOTE: if the gcc version on the machine is lower than 4.8.x then an upgrade might be needed for using the cmake.

The following packages/libraries are needed to run the unit tests & component tests for that project, so they are not mandatory – but I think it is desirable to run them prior to running the executable:  
  
b) Install Python requests library:  
***$ sudo apt-get install python-requests***

c) Install python-pip utility:  
***$ sudo apt-get install python-pip***

d) Installing termcolor:  
***$ sudo pip install colored***

e) Increase the number of open files for all users:  
e1) Add the two following lines to the ***/etc/security/limits.conf*** file (using your favorite editor):  
\* soft nofile 6000  
\* hard nofile 6000  
e2) reboot the system:  
***$ reboot***  
e3) Make sure the settings took place by verifying that the following command outputs 600  
***$ ulimit -n***

**2) Building the project:**  
a) Extract the compressed folder and go to the “main folder”  ***ts-task***.  
b) Run the build script with a single argument which is the full path to the ts-task folder (inclusive – it is where the “main” cmake file located at).  
The build script is under ts-task/scripts (buildHttpServerEntireProject.sh), for example: ***$ ./scripts/buildHttpServerEntireProject.sh /home/guya/Desktop/Guy/ts-task/***  
Upon successful build the output should look as follows:  
-- Configuring done

CMake Warning in CMakeLists.txt:

The build directory is a subdirectory of the source directory.

This is not supported well by Eclipse. It is strongly recommended to use a

build directory which is a sibling of the source directory.

-- Generating done

-- Build files have been written to: /home/guya/Desktop/Guy/ts-task/build  
c) Go to build folder and run make:  
***$ cd build (now you should be at : /home/guya/Desktop/Guy/ts-task/build)***  
***$ make***  
Upon successful build you should see the progress of cmake up to 100%, terminating like so:

[ 97%] Building CXX object unitTests/googletest-release-1.8.0/googlemock/CMakeFiles/gmock\_main.dir/src/gmock\_main.cc.o

[100%] Linking CXX static library ../../../build/libgmock\_main.

From this stage on it is not mandatory, YET it is desirable to run these checks to verify proper behavior of the server.  
d) Run the unit tests:  
***$ go back to the main folder (ts-task)***  
***$ ./scripts/runUnitTests.sh***  
NOTE: all tests should pass

e) Run component tests (tests which are written in Python that sends requests towards the server).  
e1) Functionality tests:  
***$ go back to the main folder (ts-task).  
$ ./tests/runFunctunalityComponentTests.sh***  
NOTE: all tests should pass.  
  
e2) Load test:  
Here I tried to send to the server as many requests as possible. When I ran the load test, the maximum requests I was able to send to it were ~ 1500 in total – otherwise I got errors (I guess) indicating   
***$ go back to the main folder (ts-task).  
$ ./tests/runLoadComponentTests.sh***  
**3) Running the executable:**The executable requires two parameters:  
a) The full path to the DB txt file, i.e.- /full/path/to/DBfile/ts-task/textFiles  
NOTE: within ts-task/textFiles folder, there are several text files, one of them is the words\_clean.txt provided.  
b) Number of worker threads – when I tested the server I used 2, I guess that there could be more fine tuning about that.  
NOTE: It is located within the ts-task/build/build folder.  
  
For example:  
$ move to the ts-task/build/build folder, and from there run the following:  
***$ ./serverExe /home/guy/Desktop/ts-task/textFiles/words\_clean.txt 2***