# Department of Computing

**CS213: Advanced Programming**

**Class: BSCS – 6C**

# Lab 4: Development of a Multithread Client Server Application

**Date: October 05, 2018**

# Time: Friday 2:00 PM – 5:00 PM

# Instructor: Dr. Abdul Ghafoor

# 

# Lab 3: Development of Multithread Client Server Application

## Introduction

In this lab, students have to design, develop, and test a multithreaded client server application. The client will read a file line by line and then will upload on the server. The server will create a dedicated thread to handle client requests and save received data into a file.

## Objectives

* Develop a multithreaded client server application
* Handling of multiple users through multi-threading.
* Exception Handling

## Tools/Software Requirement

* Solutions should be implemented using Java.
* **Do not use any external library for reading files and creating client server.**

**Description**

Each student must, individually build the complete application on their own. Students must upload their solutions on LMS to qualify for evaluation.

* Any exceptions or errors leading to non-execution of submitted code.
* Failure to upload the solution on LMS.
* Failure to submit original code.
* Failure to explain the submission, during viva.

**Lab Task**

Develop a java based multithreaded client server application in which your client application will accept a file name from the user and then will read it line-by-line. Before uploading data, client application will send name of the file. Once the file name is sent, then it will start sending actually data which will carry a line in each message. When whole file is uploaded then the client will send end message and the server will reply with done as shown in the figure 1.

At the server side, when it will accept connection request then it will create a dedicated thread which will receive name of file from user and then will create a file on local directory. After that it will start appending received data into the file. When it will receive an end message it will send done message as a reply and then will close the file and corresponding socket. Format of the messages are given in the protocol diagram.

name:file\_name

data:aLine

data:aLine

data:aLine

end

done

client

server

## Deliverables

* Each submission is individual with the following composition:
  + Source Code
  + README.txt (Introduction, Approach, How to Run)
* Convert your submission files into a zip folder and name it as given below, finally upload the zip folder to LMS.
  + Name – Registration No. – Section

## Grade Criteria

This lab is graded. Min marks: 0. Max marks: 10.

|  |  |  |
| --- | --- | --- |
| **Activity** | **Minimum** | **Maximum** |
| Documentation with clearly defined understanding of the lab task and approach | 0 | 2 |
| Code clarity with clean, formatted and commented code. | 0 | 3 |
| Functionality | 0 | 3 |
| Viva | 0 | 2 |
| **Total** | **0** | **10** |