

# Assignment 3 – Music Server

## Contributes 6% to overall mark

**This is an individual assignment.** The work must be done on your own.

**Submissions will be via ESP**

Due Date: Sunday May 25 at Midnight (Just before Monday!)

### **Submission Detail**

Full copy of source code and compiled classes with documentation all written in the Java in one zip file.

### **In Class Assessment**

You will need to demonstrate your assignment to your tutor in week 12.

## **Qualities**

This is a code writing assignment and it is expected that a significant amount of the code you write will be sourced from the work of others. The work sourced from others **MUST BE** acknowledged by the use of inline comments in your source code.

You may not submit other students work. You may not ask someone to write part or all of the code or pay someone to write part or all of the code.

The main criteria to pass are;

- You develop working programs
- You provide code that is meticulously documented
  - Uses javadoc
  - Complex areas are explained with comments.
  - Includes references to all sources
- You understand the work submitted and can explain how it works to your tutor
- You successfully demonstrate the code in class to your tutor.
- All projects run on **either** the virtual or the real network in EN305.

## **Project Overview**

You are to develop music sharing software that includes a central server and a number of peers.

The role of the central server is to

- Keep track of who is online
- Receive information about who is online
- Keep a list of songs that each peer has
- Save list of songs to file
- Receive lists of songs that a peer has
- Provide a list of peers that have a song and are online

The tasks a peer is to perform include

- Register with the server to say I'm on line
- Send updates to song held to the server
- Make requests to the server for who has a song
- Request songs from other peers (using TCP)
- Supply songs to other peers (Using TCP)

## Example behaviours

- When a peer comes on line it registers with the central server. The central server will keep track of a set of peers that are online.
- Peer adds a new song to its list and notifies the music server.
- Peer deletes a song from its list and notifies the server
- Peer asks server for a list of currently online IP's that have a song.
- Peer requests another peer to supply a song
- Peer receives request for a song from another peer and supplies the song.
- Server receives a request for a song, looks up which peers are on line and have the song and returns the list to the peer making the request.
- Server receives a revised song list and updates its records of who has what song

## Minimum testing Environment

- 3 peers
- 1 server
- Each peer has 5 songs
- Same song on at least three different peers ( \*2)
- At least one unique song on two peers

May be a GUI or a console Application or both.  
All files will be in mp3 format.