



## CSC13002 – LẬP TRÌNH ỨNG DỤNG JAVA

### ĐỒ ÁN MÔN HỌC

#### ĐỒ ÁN #2 – A CLIENT MONITORING SYSTEM

##### I. Thông tin chung

Mã số bài tập:	DAMH#2
Thời lượng dự kiến:	20 h
Deadline nộp bài:	Moodle
Hình thức:	Bài tập cá nhân
Hình thức nộp bài:	Nộp qua Moodle môn học
GV phụ trách:	Nguyễn Văn Khiết (LT), Hồ Tuấn Thanh (TH)
Thông tin liên lạc với GV:	htthanh@fit.hcmus.edu.vn

##### II. Chuẩn đầu ra cần đạt

Bài tập này nhằm mục tiêu đạt được các chuẩn đầu ra sau:

- G1.1: Thành lập, tổ chức, vận hành và quản lý nhóm
- G1.3: Phân tích, tổng hợp và viết tài liệu kỹ thuật theo mẫu cho trước theo cá nhân hoặc cộng tác nhóm
- G4.2: Vận dụng kiến thức lập trình giao diện đồ họa Swing
- G4.4: Vận dụng kiến thức lập trình mạng
- G5.2: Vận dụng kiến thức lập trình đa tiểu trình
- G5.8: Vận dụng Quản lý ngoại lệ trong bài toán đơn giản
- G6.1: Hình thành bài toán ban đầu
- G6.2: Phân tích bài toán đưa ra
- G6.3: Phân rã bài toán thành các tác vụ và lựa chọn công nghệ Java phù hợp để cài đặt
- G7.1: Sử dụng các IDE hỗ trợ lập trình như Visual Studio Code, IntelliJ IDEA, Eclipse, Netbeans ...

##### III. Mô tả bài tập

Sinh viên được yêu cầu xây dựng một ứng dụng giám sát tập tin thư mục trên các máy thành viên (client), sử dụng ngôn ngữ lập trình Java, đồ họa (GUI), xử lý mạng và đa tiến trình với các chức năng sau.

1. **(5đ)** Đầu tiên, người quản lí start server. Server sẽ hiển thị IP, port để người quản lí có thể gửi IP, port này cho các người dùng thành viên ở client để kết nối vào server.
2. **(5đ)** Người dùng thành viên start client. Người dùng thành viên điền vào IP, port để kết nối với server. Thông tin kết nối phải được thông báo ở server và client. Nếu kết nối bị lỗi, ứng dụng cho phép người dùng thành viên thực hiện kết nối lại.
3. **(5đ)** Người quản lí server xem được danh sách các client đang kết nối. Có thể tìm kiếm trên danh sách này, dựa vào IP của client.
4. **(5đ)** Khi client kết nối vào server, server mặc nhiên sẽ được giám sát 1 thư mục mặc định trên máy client. VD: folder C:/ClientMonitoringSystem/Data
5. **(10đ)** Người quản lí server có thể browse để xem danh sách cây thư mục ở trên máy client. Từ đó, người quản lí chọn thay đổi sang 1 folder khác để giám sát.
6. **(30đ)** Tất cả các thao tác của client trên folder được chọn, đều được giám sát và ghi log ở phía server. Một số cột cần hiển thị trên JTable như: STT, thời điểm, action, IP client, diễn giải cho action. Ngoài ra, cho phép người quản lí server có thể tìm kiếm / filter theo thời điểm, action, client....
  - Các thao tác cơ bản như: RENAME thư mục đang giám sát, CREATE một thư mục con, tập tin con, RENAME một thư mục con, tập tin con, UPDATE nội dung một tập tin con, DELETE thư mục con, tập tin con... Lưu ý, tất cả các thao tác thay đổi này được thực hiện trên các ứng dụng thông thường, chẳng hạn người dùng vào Windows Explorer rename thư mục, người dùng soạn thảo file Word và save as vào thư mục đang giám sát, người dùng dùng Total Commander copy data vào thư mục đang giám sát, người dùng dùng các chương trình nén để nén tập tin, giải nén tập tin vào trong thư mục đang giám sát. Ngoài ra, các thao tác LOG-IN khi client kết nối vào server và thao tác LOG-OUT khi client thoát khỏi server cũng cần được giám sát.
7. **(10đ)** Tất cả các thao tác của client đều được hiển thị ở phía client. Tương tự, một số cột cần hiển thị trên JTable của client như: STT, thời gian, action, diễn giải. Cũng cho phép người dùng tìm kiếm / filter trên các cột này.
8. **(20đ)** Tất cả thao tác giám sát được đều phải ghi log thành tập tin log trên máy server, để khi tắt server và mở lại server, người quản lí có thể load lại các file log cũ và xem lại.
9. **(10đ)** Tương tự, ở phía client, cũng cần ghi log thành tập tin, những gì đã được giám sát.
10. **(Tối đa ko quá 20đ)** Các chức năng khác mà SV tự đề xuất, có ý nghĩa sử dụng thực tế.



## Yêu cầu chung

1. Cần xây dựng GUI app, có xử lý mạng, xử lý đa tiểu trình. Nếu không, 0đ bài tập.
2. Sử dụng Git (GitHub/Bitbucket/GitLab) để quản lý source code. Khi nộp bài, cần nộp kèm hình chụp các lần commit. Phải có ít nhất 10 commit, nếu ít hơn sẽ 0đ. Các commits phải phân bố đều trong các ngày, tránh tình trạng toàn bộ các commit nằm trong 30 phút cuối deadline.

## IV. Các yêu cầu & quy định chi tiết cho bài nộp

1. Đặt tên bài nộp có dạng **MSSV\_SoDiemTuDanhGia.zip**.
2. Số điểm tự đánh giá, theo thang 100đ.
3. Nếu sinh viên có dưới 10 commits và trải dài trong 5 ngày, thì điểm đồ án này là 0đ.
4. Nếu sinh viên copy bài từ người khác, hoặc từ internet, thì 0đ toàn bộ phần thực hành và báo GVLT để xem xét 0đ môn học.
5. Trường hợp có sử dụng nguồn từ người khác, hoặc từ internet, thì phải đánh dấu lại trong code phần tham khảo và cho biết tổng phần trăm tham khảo trong toàn bộ đồ án.
6. Quay 1 video, upload lên Youtube bao gồm: mô tả ngắn gọn về cách thức xử lý, cho biết phần trăm tham khảo và demo toàn bộ chức năng. Video demo cần có ít nhất 1 server và 2 client.
7. Bài nộp bao gồm **source code, file jar, file data, hình chụp commit, file txt có phần chấm điểm cho từng chức năng và có chứa link video trên Youtube và đặt tên đúng qui định**. Nếu ko, sẽ 0đ.
8. Chức năng 6, hiển thị giám sát ở phía server là chức năng bắt buộc, nếu không có thì toàn bộ bài tập là 0đ.



## **V. Cách đánh giá**

1. Kiểm tra tên bài nộp, nội dung bên trong.
2. Kiểm tra giống bài.
3. Giáo viên chạy thử chương trình, kết hợp xem video để kiểm tra các chức năng.
4. Kiểm tra các mã nguồn quan trọng.

## **VI. Tài liệu tham khảo**

Slide bài giảng của GVLT

## **VII. Các quy định khác**

Không có