



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2233 - Programación Avanzada
1 semestre 2015

Actividad 25

Networking

Instrucciones

Luego de pasar con éxito la primera prueba, la compañía *Guugul* le presenta su siguiente desafío para lograr ingresar a trabajar con ellos: elaborar un programa para compartir fotos, exclusivamente en formato JPG. Para ello, la compañía le provee de una interfaz gráfica. A la compañía no le interesa que pueda enviar mensajes: solo fotos.

Interfaz gráfica

Tendrá que trabajar sobre el archivo *main.py* (contenido en la carpeta de enunciado). En este se encuentran las clases *Servidor* y *Cliente*. Cada uno contiene un campo *chat*, el cual representa sus interfaces gráficas asociadas.

- *update_image(self, path)*: actualiza la foto indicada por el parámetro *path* en la interfaz gráfica asociada.

Requerimientos

- Implementar método *enviar(path)* de la clase *Cliente*: el método *Cliente.enviar(path)* se detona cada vez que se selecciona un archivo en la interfaz del *Cliente* a partir del botón **"Subir foto"** (ya está implementado). Recibe como parámetro el *path* del archivo seleccionado. En este método ud. se debe encargar de enviar la foto contenida en ese *path* al *Cliente*.
- Implementar método *escuchar(path)* de las clases *Servidor* y *Cliente*: en la clase *Servidor*, el campo *cliente* contiene la referencia al socket del cliente que está conectado. Ud. debe detectar cuando se recibe información de ese socket y actualizar la interfaz del Servidor.
En el caso de la clase *Cliente*, el campo *s_cliente* referencia a su propio socket (de cliente) y es por este medio que recibe información del servidor. Ud. debe detectar cuando se reciba información de ese socket y actualizar la interfaz del Cliente.

To - Do

- (2.50 ptos.) Método *Cliente.escuchar*.
- (2.50 ptos.) Método *Servidor.escuchar*.

- (1.00 ptos.) Método *Cliente.enviar*.
- BONUS (1.50 ptos.) Poder enviar imágenes tanto en JPG como en PNG.

IMPORTANTE

- Recomendación: mande los archivos como **binarios**.
- No intente entender el resto del código: **tiene que abstraerse de él**. Solo tiene que enfocarse en los métodos escuchar y enviar y saber qué parámetros reciben. Además, acuérdesse de usar el campo chat para actualizar la interfaz en estos métodos cuando corresponda.
- Verifique tener la imagen completa antes de intentar actualizar la interfaz gráfica de *Servidor* o *Cliente*. En los métodos *escuchar* debe tener en cuenta que la imagen no va a estar totalmente contenida en el retorno de `socket.recv()`, ya que es mucha información para ser recibida de una vez. Es decir, tendrá que recibir por partes la imagen. Para ello, le será útil verificar usando la función ***select.select([socket_donde_se_recibe_data], [], [], 0)*** si quedan datos en el buffer del socket por recibir. Si es que quedan, utilice la función *socket.recv()*. Si no quedan, guarde la imagen en algún lado (o puede ir generando la imagen a medida que recibe la información), recuerde el path donde lo guardó y actualice la interfaz gráfica.