

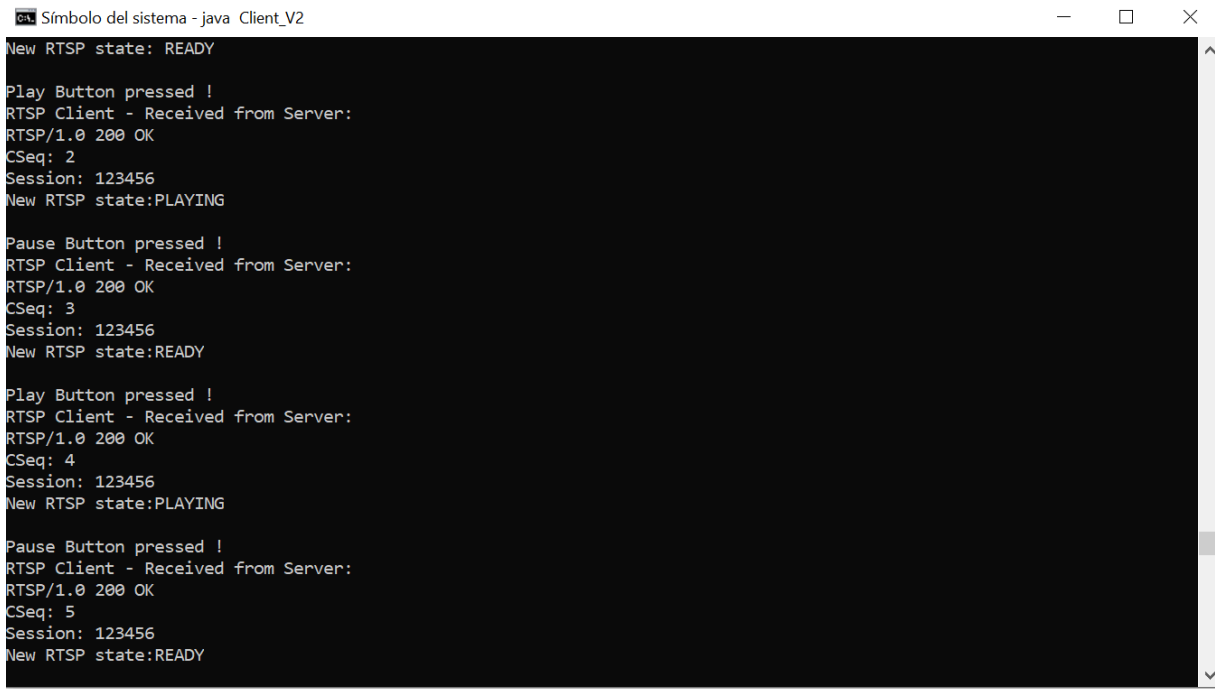
Trabajo Laboratorio: Sistemas Multimedia- Streaming Media - (Real Time Streaming Protocol – Cliente +)

En el laboratorio anterior implementamos un servicio de vídeo streaming basado en RTSP.

Las aplicaciones que se apoyan en este protocolo, además de garantizar una comunicación eficiente, pueden identificar y ofrecer al usuario información actualizada del modo en que se está realizando dicha comunicación.

La monitorización del estado de la comunicación supone un valor añadido a la aplicación que ofrece un servicio video streaming, en la medida que permite la posibilidad de tomar decisiones basadas en el análisis de los valores monitorizados.

Comprobaremos en primer lugar la información que monitoriza la aplicación Client_V1 que acabáis de implementar en la práctica anterior. Veremos que la información es la siguiente:



```
Símbolo del sistema - java Client_V2
New RTSP state: READY

Play Button pressed !
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 2
Session: 123456
New RTSP state:PLAYING

Pause Button pressed !
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 3
Session: 123456
New RTSP state:READY

Play Button pressed !
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 4
Session: 123456
New RTSP state:PLAYING

Pause Button pressed !
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 5
Session: 123456
New RTSP state:READY
```

En esta práctica, queremos añadir a esta información, la siguiente: tamaño de cada paquete RTP; número de secuencia de cada paquete; valor del time stamp de cada paquete; y código del códec utilizado en cada paquete, para conseguir la siguiente presentación:

```

Símbolo del sistema - java Client
Play Button pressed !
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 6
Session: 123456
New RTSP state:PLAYING
PayloadLength = 5085

Received RTP packet with SeqNum # 14 and TimeStamp 1400 ms, of type 34
10000000 00100010 00000000 00001110 00000000 00000000 00000101 01111000
PayloadLength = 5358

Received RTP packet with SeqNum # 15 and TimeStamp 1500 ms, of type 34
10000000 00100010 00000000 00001111 00000000 00000000 00000101 11011100
PayloadLength = 5685

Received RTP packet with SeqNum # 16 and TimeStamp 1600 ms, of type 34
10000000 00100010 00000000 00010000 00000000 00000000 00000110 01000000
PayloadLength = 6010

Received RTP packet with SeqNum # 17 and TimeStamp 1700 ms, of type 34
10000000 00100010 00000000 00010001 00000000 00000000 00000110 10100100
PayloadLength = 6334

Received RTP packet with SeqNum # 18 and TimeStamp 1800 ms, of type 34
10000000 00100010 00000000 00010010 00000000 00000000 00000111 00001000

Pause Button pressed !
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 7
Session: 123456
New RTSP state:READY

```

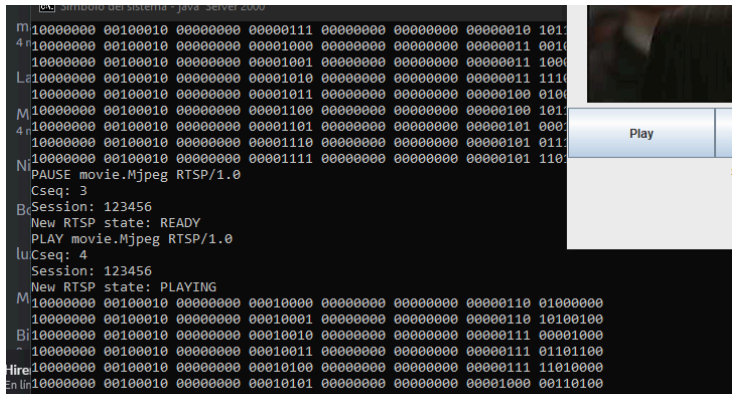
Se pide, sobre el código Client_V1.java:

1. Comprobar la información que monitoriza la aplicación Client_V1 de la práctica anterior.
2. Monitorizar la carga útil de cada paquete RTP (los bytes de datos de vídeo).
3. Monitorizar el número de secuencia de cada paquete.
4. Monitorizar el valor del time stamp de cada paquete.
5. Monitorizar el código del códec utilizado en cada paquete.
6. Monitorizar los bits de la cabecera RTP, sin el campo SSRC.

Estas tareas se programarán dentro de la clase: class timerListener implements ActionListener, que se encuentra en el fichero fuente: Client_V1.java.

Tarea 1: Comprobar la información que monitoriza la aplicación Client_V1 de la práctica anterior.

Screen-shot 1:



Breve descripción:

CSeq = 3

Session = 123456

State = Ready

Tarea 2: Monitorizar la carga útil de cada paquete RTP (los bytes de datos de vídeo)

Screen-shot 2:

```

Header: 10000000 00100010 00000000 00010110 00000000 00000000 00001000 10011000
PayloadLength = 7314
Received RTP packet with SeqNum # 23 and TimeStamp 2300 ms, of type 34
Header: 10000000 00100010 00000000 00010111 00000000 00000000 00001000 11111100
PayloadLength = 7459

Pause Button pressed!
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 5
Session: 123456
New RTSP state: READY

```

Breve descripción:

PayloadLength = 7459

Tarea 3: Monitorizar el número de secuencia de cada paquete.

Screen-shot 3:

```

Header: 10000000 00100010 00000000 00010110 00000000 00000000 00001000 10011000
PayloadLength = 7314
Received RTP packet with SeqNum # 23 and TimeStamp 2300 ms, of type 34
Header: 10000000 00100010 00000000 00010111 00000000 00000000 00001000 11111100
PayloadLength = 7459

Pause Button pressed!
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 5
Session: 123456
New RTSP state: READY

```

Breve descripción:

SeqNum = 23

Tarea 4: Monitorizar el valor del time stamp de cada paquete.

Screen-shot 4:

```
Header: 10000000 00100010 00000000 00010110 00000000 00000000 00001000 10011000
PayloadLength = 7314
Received RTP packet with SeqNum # 23 and TimeStamp 2300 ms, of type 34
Header: 10000000 00100010 00000000 00010111 00000000 00000000 00001000 11111100
PayloadLength = 7459

Pause Button pressed!
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 5
Session: 123456
New RTSP state: READY
```

Breve descripción:
TimeStamp = 2300ms

Tarea 5: Monitorizar el código del códec utilizado en cada paquete.

Screen-shot 5:

```
Header: 10000000 00100010 00000000 00010110 00000000 00000000 00001000 10011000
PayloadLength = 7314
Received RTP packet with SeqNum # 23 and TimeStamp 2300 ms, of type 34
Header: 10000000 00100010 00000000 00010111 00000000 00000000 00001000 11111100
PayloadLength = 7459

Pause Button pressed!
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 5
Session: 123456
New RTSP state: READY
```

Breve descripción:

Codec: MJPEG

1. **Tarea 6:** Monitorizar los bits de la cabecera RTP, sin el campo SSRC.

Screen-shot 6:

```
Header: 10000000 00100010 00000000 00010110 00000000 00000000 00001000 10011000
PayloadLength = 7314
Received RTP packet with SeqNum # 23 and TimeStamp 2300 ms, of type 34
Header: 10000000 00100010 00000000 00010111 00000000 00000000 00001000 11111100
PayloadLength = 7459

Pause Button pressed!
RTSP Client - Received from Server:
RTSP/1.0 200 OK
CSeq: 5
Session: 123456
New RTSP state: READY
```

Breve descripción:

Header: 10000000 01000110 00000000 00000001 00000000 00000000 01100100

Primer byte: 10000000

Versión: 10

Padding: 0

Extensión: 0

CC: 0000

Segundo byte: 01000110

Marcador:

Payload Type: 100110

Tercer y cuarto byte: 00000000 00000001

Num Seq: 1

Quinto al octavo byte: 00000000 00000000 00000000 01100100

Timestamp: 100
