OBLIGATORIO 1 - SLASHER

Programación de Redes – octubre 2018

Universidad ORT Uruguay – Facultad de Ingeniería
Francisco Pollio - 192448
Andres Della Cella - 193949

Luis Barrague & Roberto Assandri

Índice

Francisco Pollio – 192448
Andrés Della Cella – 193949

1. Objetivo

El obligatorio tiene como objetivo simular el juego “Slasher”, mediante la construcción de dos aplicaciones de consola en la cual una actúa como cliente y otra como servidor.

La aplicación del cliente procesara los pedidos de los clientes que en este caso serán los jugadores, y podrán acceder a todas las funcionalidades del menú, así como el juego.

La aplicación del lado del servidor procesará todos los pedidos de los clientes conectados y les proveerá con una respuesta manejando la lógica del juego.

Francisco Pollio – 192448
Andrés Della Cella – 193949

2. Requerimientos

2.1 Cliente

RF1. Conectarse y desconectarse al servidor.

El cliente deberá ser capaz de conectarse (y desconectarse) a un servidor.

RF2. Dar de alta un jugador.

El sistema permitirá dar de alta jugadores, los jugadores tienen un nickname (sobre nombre) y un avatar (foto).

RF3. Permitir al jugador conectarse al juego.

El jugador enviará su nickname y en caso de que alguien más (otro cliente) no lo esté usando se sumará al juego (no a la partida activa).

RF4. Permitir a un jugador unirse a la partida activa.

Luego de que un jugador se haya conectado al juego podrá intentar conectarse a la partida activa, en caso de que en el servidor exista una se pasará a la selección y luego a la partida, en caso de no haber una partida activa el sistema le informará de esto al jugador.

RF5. Permitir al jugador seleccionar un rol antes de iniciar la partida.

El jugador puede elegir dos roles, monstruo o sobreviviente, el monstruo tiene cien puntos de vida y un poder de ataque de diez. El sobreviviente tiene veinte puntos de vida y un poder ataque de cinco. El monstruo sólo podrá atacar a sobrevivientes y monstruos, el sobreviviente puede sólo atacar a los demás monstruos.

RF6. Permitir al jugador realizar acciones durante la partida activa.

Como se detalló en el RF5 del cliente los jugadores pueden realizar acciones en la partida, se pueden mover o atacar, pero no hacer las dos cosas a la vez (tank controls).

RF7. Mostrar el resultado de la partida activa cuando termina.

Una partida puede terminar en diferentes escenarios, el manejo de esta será exclusivo del servidor y aquí sólo mostraremos el resultado

Francisco Pollio – 192448
Andrés Della Cella – 193949

2.2 Servidor

RF1. Aceptar pedidos de conexión de un cliente.

El servidor debe ser capaz de aceptar pedidos de conexión de varios clientes a la vez.

RF2. Dar de alta un jugador.

El sistema permitirá dar de alta jugadores, los jugadores tienen un nickname (sobre nombre) y un avatar (foto). En caso de que un usuario quiera dar de alta un usuario existente no será posible hacer esto y se le informará al usuario.

RF3. Mostrar los jugadores registrados.

El servidor deberá mostrar una lista de todos los jugadores registrados en el mismo.

RF4. Mostrar los jugadores conectados actualmente al sistema.

El servidor deberá mostrar una lista (que se debe actualizar automáticamente) de los jugadores que están actualmente conectados al sistema.

RF5. Iniciar partida.

El servidor deberá poder iniciar una y sólo una partida activa, la misma durará tres minutos. No deben permitirse acciones de interacción con el sistema servidor mientras una partida está en juego.

RF6. Finalizar partida.

Aquí tenemos diferentes escenarios para terminar una partida:

- ♣ El tiempo terminó y hay sobrevivientes vivos, sobrevivientes ganan.
- ♣ Quedan sólo sobrevivientes, sobrevivientes ganan.
- ♣ Queda sólo un monstruo vivo, gana el jugador que tenía dicho monstruo.
- ♣ El tiempo terminó y sólo hay monstruos vivos, nadie gana.
- ♣ Si un jugador muere no puede seguir jugando y deberá esperar a una nueva partida para volver a jugar.

RF7. Permitir a un jugador unirse a la partida activa.

Cuando el servidor se encuentra en modo partida activa es posible que se conecten jugadores no conectados al sistema y se unan a la partida activa.

RF8. Mostrar resultado de partida.

Cuando una partida termina el servidor deberá enviar un mensaje con el resultado a todos los jugadores conectados actualmente al sistema. Ahí se debe volver permitir ver las opciones marcadas en los RF 3 y 4 o poder iniciar otra partida.

2.3 Anexo: Reglas del Juego

- El juego deberá ser un terreno de ocho por ocho.
- Cada espacio podrá ser sólo ocupado por un jugador.
- Si un jugador llega a los límites del terreno este no podrá moverse para afuera de los mismos.
- El sistema deberá informarle a un jugador si otro jugador se encuentra cercano a este (el estar cercano es un radio de uno) y el tipo de jugador que es la otra persona (monstruo o sobreviviente).

Francisco Pollio – 192448
Andrés Della Cella – 193949

- El jugador se puede mover hasta dos pasos por turno.
- El jugador puede atacar si otro jugador (o jugadores) se encuentra próximo a él.
- Monstruo puede atacar a cualquiera, sobreviviente puede atacar sólo a monstruo.

Francisco Pollio – 192448
Andrés Della Cella – 193949

3. Diseño

El equipo decidió agrupar el sistema en 4 paquetes buscando siempre la alta cohesión y el bajo acoplamiento.

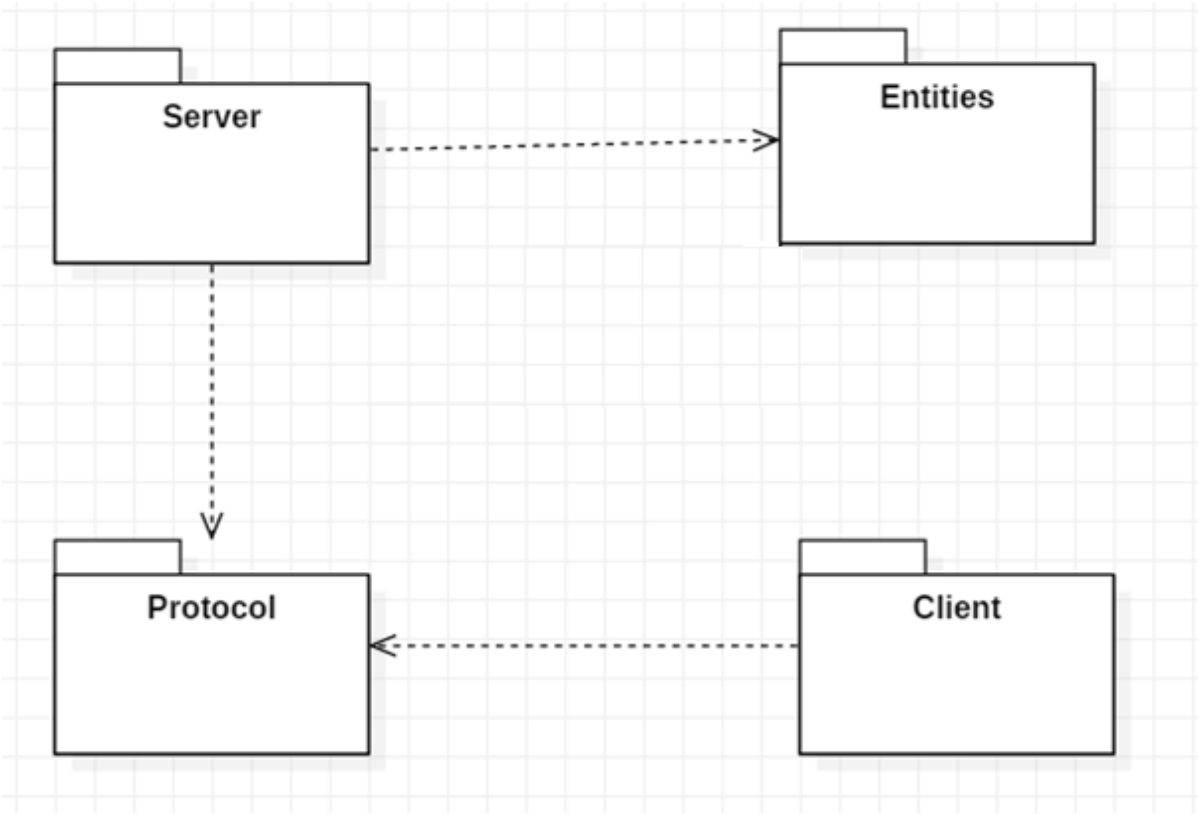
Client: Encargado de manejar el sistema que se encarga de recibir los pedidos del cliente y enviarlos para que sean atendidos por el servidor.

Server: Encargado de manejar el sistema del servidor que recibe todos los pedidos de los distintos clientes, los procesa y les da su correspondiente respuesta. Maneja a su vez la lógica del juego.

Protocol: Encargado de manejar las conexiones, generar las tramas correctas y realizar sus envíos correspondientes. Ya sea de cliente a servidor o de servidor a cliente.

Entities: Encargado de manejar las entidades del sistema.

3.1 Diagrama de Paquetes

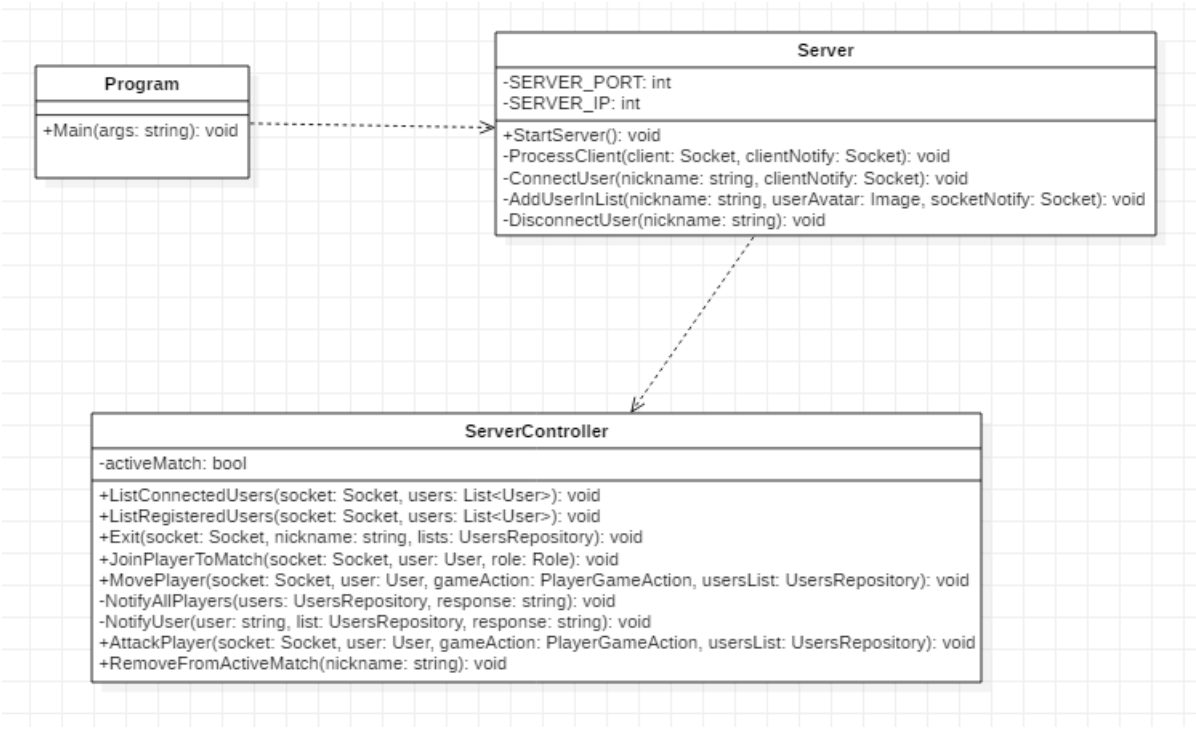


Francisco Pollio – 192448
Andrés Della Cella – 193949

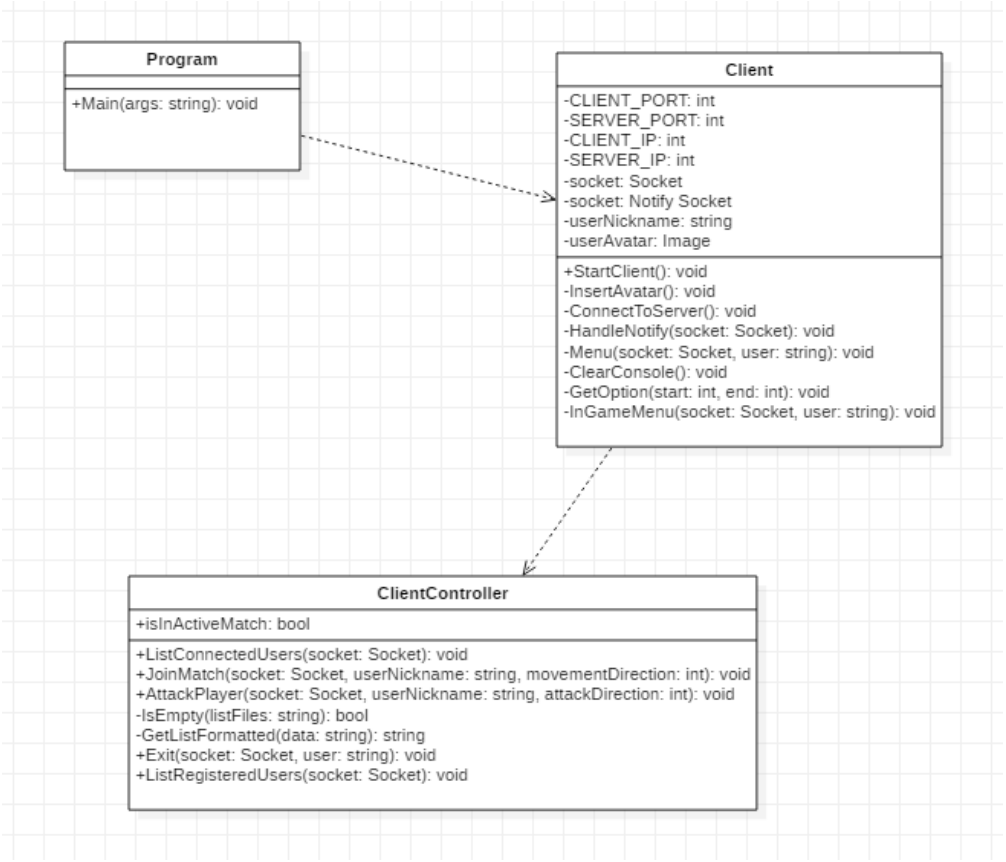
3.2 Diagramas de Clases

Francisco Pollio – 192448
Andrés Della Cella – 193949

3.2.1 Server

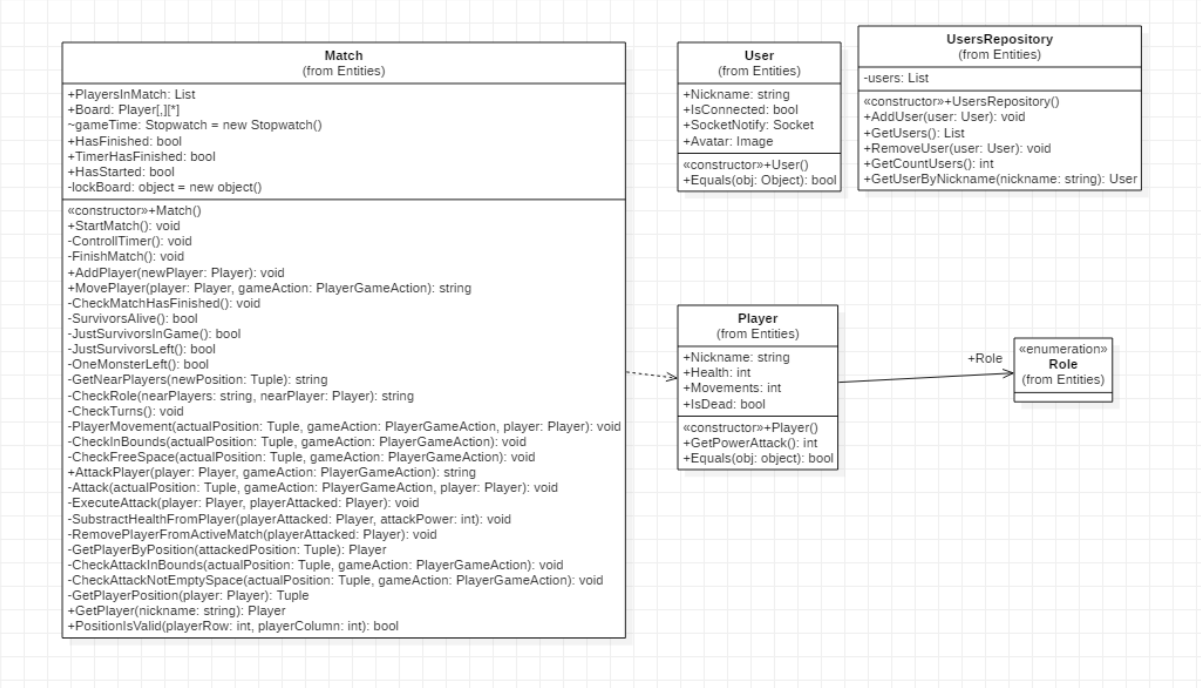


3.2.2 Client

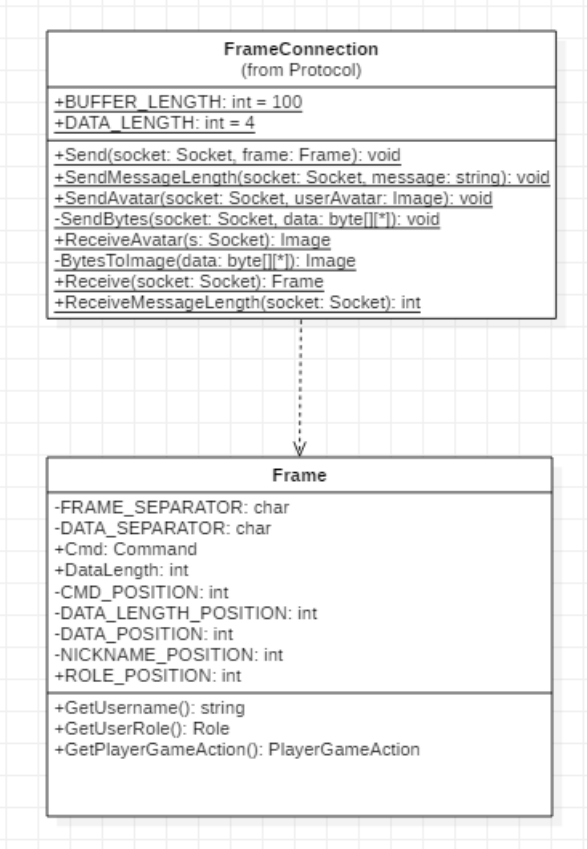


Francisco Pollio – 192448
Andrés Della Cella – 193949

3.2.3 Entities

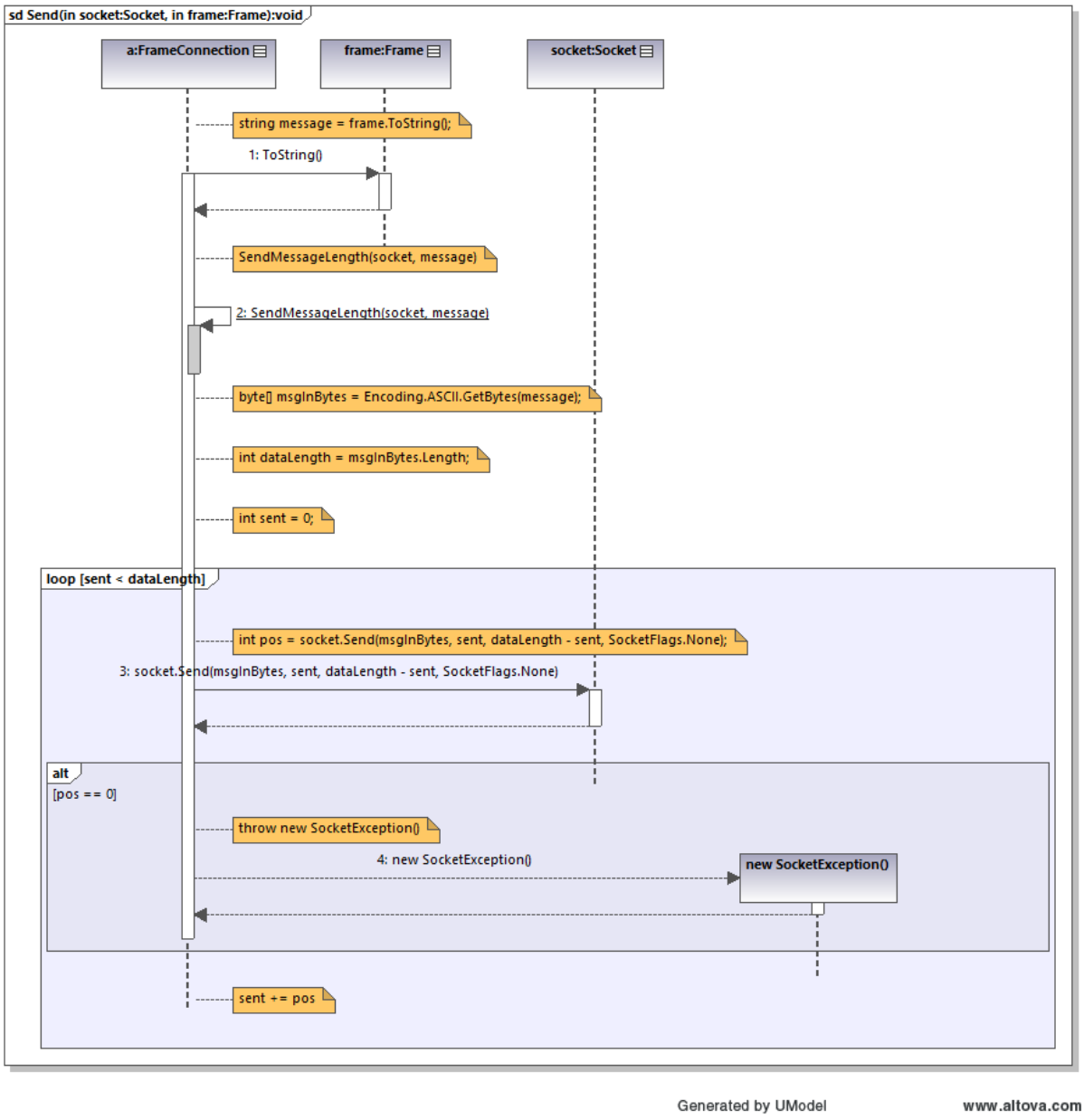


3.2.4 Protocol

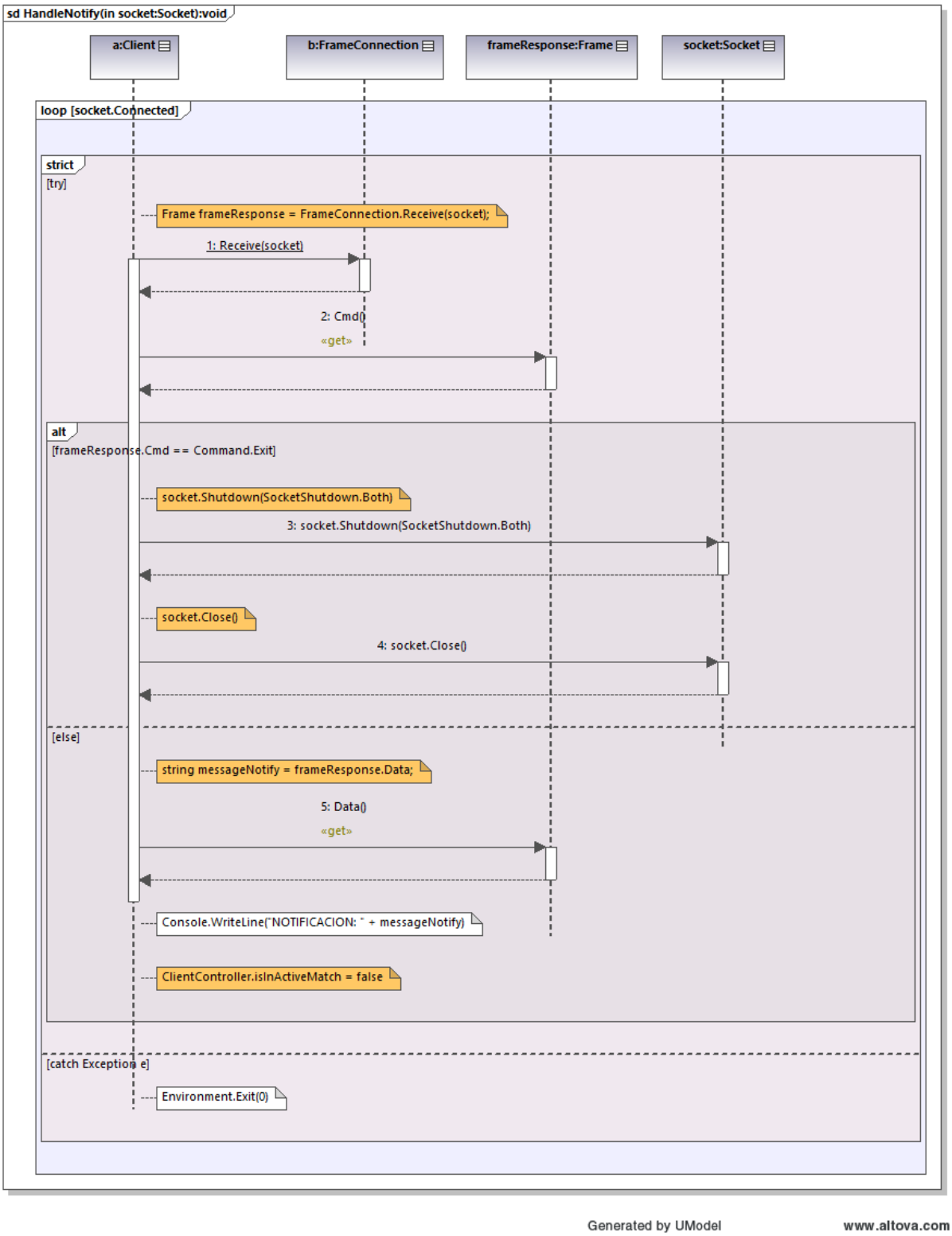


Francisco Pollio – 192448
Andrés Della Cella – 193949

3.2 Diagramas de Interacción



Francisco Pollio – 192448
Andrés Della Cella – 193949



Francisco Pollio – 192448
Andrés Della Cella – 193949

4. Protocolo

Se decidió que el protocolo proporcionado por la letra no iba a ser el mas adecuado para nuestro sistema, por lo tanto, se decidió adaptarlo para nuestras necesidades.

El formato resultante es el siguiente:

Campo	Cmd	Largo	Data
Valores	0-99	Entero	Variable
Largo	2	4	Variable

El campo acción hace referencia al tipo “Command” que en el alcance de esta entrega puede tener los siguientes valores:

- `ConnectToServer`
- `ListConnectedUsers`
- `ListRegisteredUsers`
- `JoinMatch`
- `Exit`
- `MovePlayer`
- `AttackPlayer`

Cada una de estas opciones esta asociada a los pedidos del cliente.

5. Decisiones y Supuestos

- Si un jugador realiza un pedido para unirse a una partida y no hay ninguna partida activa, se crea una nueva partida y se une al jugador automáticamente.
- Se le pide el Rol a un jugador cuando se va a unir a una partida.
- Si un cliente intenta conectarse con el nickname de un usuario ya registrado, se lo asocia con el mismo, si este no esta conectado.

Francisco Pollio – 192448
Andrés Della Cella – 193949