Ítem 2 : Funcionamiento/Reglas del juego y protocolo de Funcionamiento/Reglas del juego y protocolo de comunicación comunicación Describe cuáles son las reglas del juego:

1. Cuál es la situación inicial en el juego

Hay un número de lobos(hasta 3) y un numero de pueblos (el resto)

1. Cómo evoluciona

Hay dos turnos, turno de dia y turno de noche, en el turno de dia todos los jugadores deben votar a otro, una vez todos los jugadores han votado, el jugador más votado muere y se hace de noche, durante el turno de la noche, el chat se deshabilita para los jugadores que no son lobos y los lobos deben votar a que jugador quieren matar. En caso de empate no hay muerto.

1. Cuáles son las condiciones de finalización y quién es el ganador

La partida termina cuando todos los lobos han muerto (victoria de los Villagers), o cuando todos los villagers mueren (victoria de los lobos)

PROTOCOLO DE MENSAJES CLIENT-SERVER

|  |  |  |
| --- | --- | --- |
| Estructura mensaje | Sentido | Acción asociada |
| CMSJ<mensaje,idSender> | Client -> Server | Envía un mensaje al servidor para que este procese si lo puede enviar y en caso afirmativo lo reenvia a todos los jugadores que corresponde |
| SMSJ<mensaje> | Server->Client | Envia un mensaje a todos los jugadores para que les aparezca en el chat |
| DEAD\_<players.alive,role> | Server->Client | Envia la información de vida o muerte de todos los players, si alguno esta muerto manda también el rol |
| ENDTURN\_<turn,mensaje> | Server->Client | Envia el turno al que pasa el juego y envía un mensaje informativo |
| DISCONNECT\_<id,role> | Server->Client | Envia la información importante (id y rol) de un jugador que se ha desconectado y por tanto esta muerto |
| VOTE\_<votingId,votedId> | Client->Server | Envia una intención de voto de un jugador a otro |
| ALERTDISCONNECT<id> | Server->Server | Este código se lo manda a si mismo el Server para recordarle que debe informar a los jugadores de que alguien se ha desconectado |

Protocolo Comunicación PEER-PEER

|  |  |  |
| --- | --- | --- |
| Estructura mensaje | Sentido | Acción asociada |
| NAME\_<userName> | Peer->BSS | Envía al BSS su nombre |
| INFOS\_<role,id,peersSize,peersIps,  peersPorts,peersIds,peersNames,  peersRoles | BSS->Peer | Envia a un peer toda la información de los peers que se han conectado antes que el |
| INFOP\_<id,username,role> | Peer->Peer | Envia a un peer al que se ha conectado su información clave |
| VOTE\_<votingId,votedId> | Peer->Peer | Envia a un peer la intención de votar a otro peer |
|  |  |  |
|  |  |  |
|  |  |  |