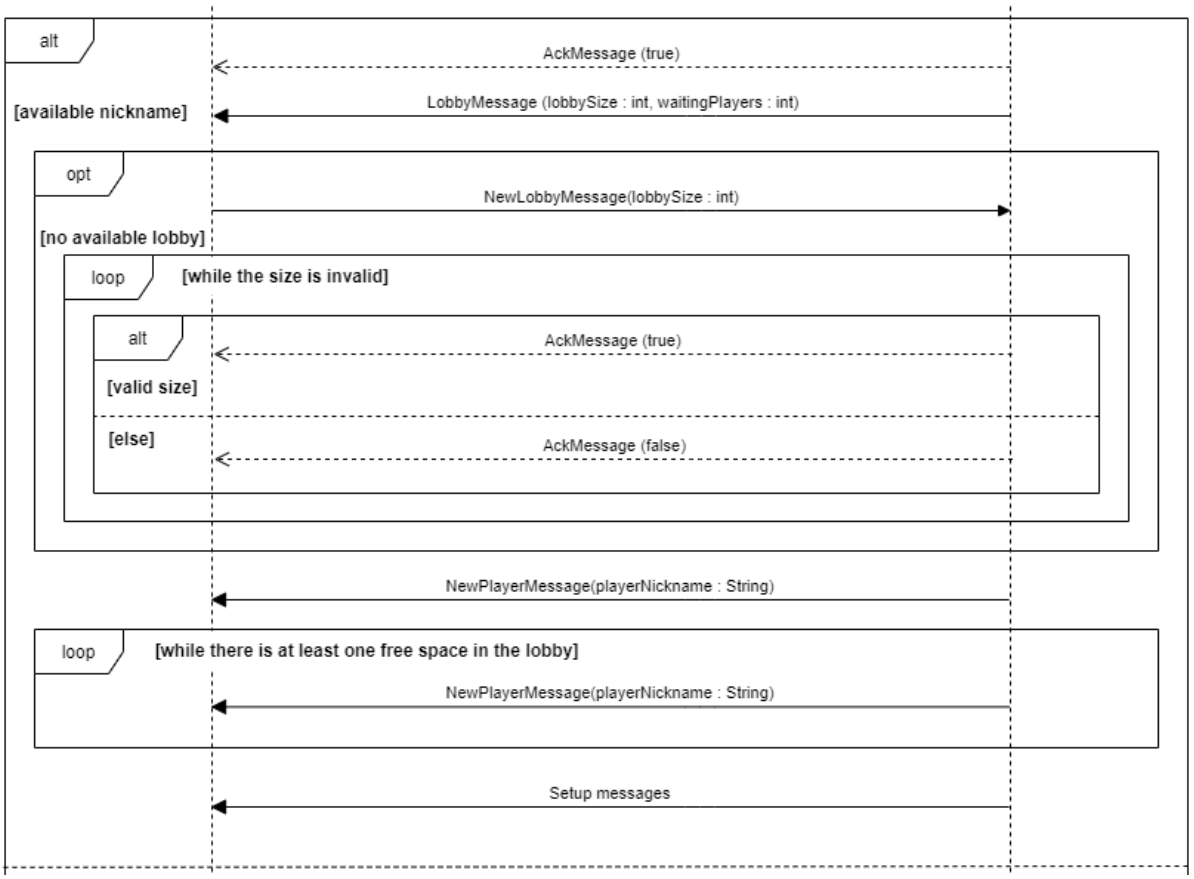
SEQUENCE DIAGRAMS

1. Connection to the server and game setup



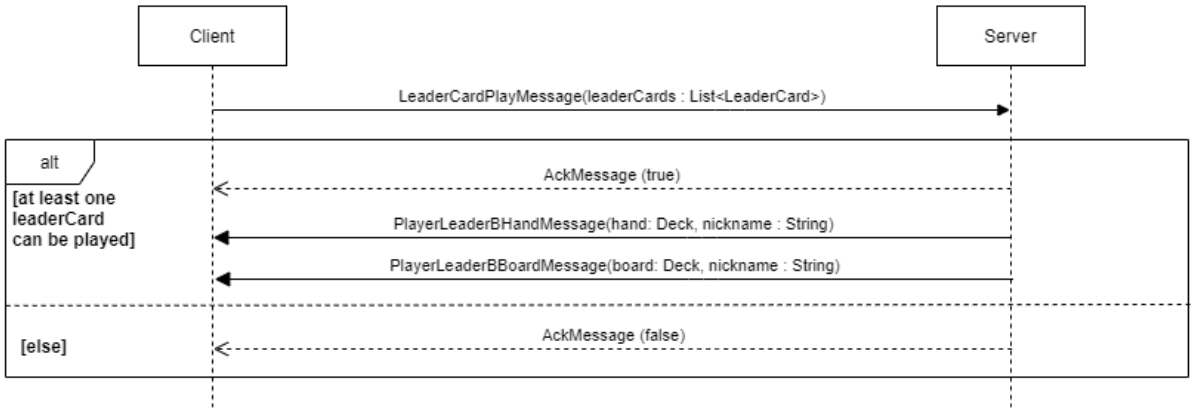
The setup phase starts with a client connecting to the server, sending their nickname.
If the nickname is already taken or contains invalid characters the server answers with an *AckMessage* to notify the error, otherwise the client is sent a *LobbyMessage* containing information about the current lobby which is still waiting for at least one other player, if there isn't an available lobby the player has to create a new one and specify its size to the server with a *NewLobbyMessage*.
When the player is added to the lobby (either new or already existing), the server sends a NewPlayerMessage containing the player's nickname to everybody in the lobby, and this action is repeated for every new player that joins the lobby, this is done so that every client can keep track of the current lobby status.
When the number of waiting players reaches the lobby maximum size, the server creates the game locally and proceeds to send a set amount of setup messages to every client in order to start the game.

The setup messages are respectively:

- a *MarketMessage* to set up the Market component
- a *DevelopmentDecksMessage* to set up the DevelopmentCard decks
- a *FaithTrackMessage* to set up the FaithTrack component
- a certain number of *PlayerBoardSetupMessage* messages for every player

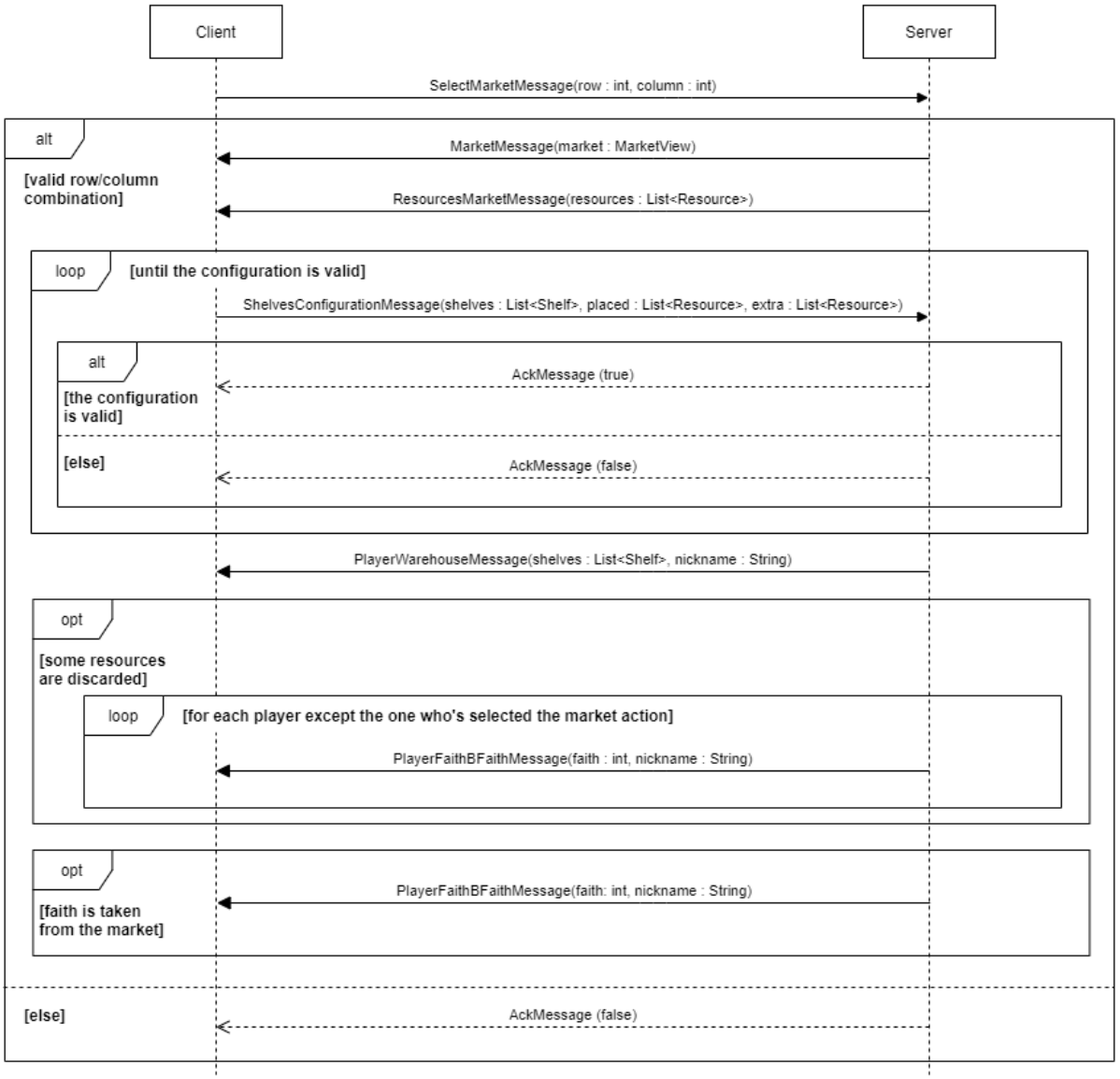
If the game is in single player mode (lobby size of 1), a *LorenzoFaithMessage* is also sent in order to create the LorenzoBoard.

2. Buying a leader card



When a player buys a LeaderCard, a *LeaderCardPlayMessage* containing the LeaderCard details is sent to the server, the server checks if the player meets the requirements to play the selected card and sends an *AckMessage*, either positive or negative.
If the player is able to play the card, after updating the game status, the server will send a *PlayerLeaderBoardMessage* relative to the player's LeaderBoard to every player in the game.

3. Acquiring resources from the Market



When a player selects a row or a column for the Market, a *SelectMarketMessage*, containing their choice, is sent to the server.
The server, after calling the designated method, will send a *MarketMessage* to every player in order to update the Market status and a *ResourcesMarketMessage* containing the selected resources to the calling player.
The player, after resolving eventual wildcards present in the resources, will send back a *ShelvesConfigurationMessage* containing the desired shelves arrangement for the Warehouse and an eventual list of discarded resources, the server will respond with an *AckMessage*; this operation is repeated until the client submits a valid configuration for the shelves, which will be met by a positive *AckMessage* from the server.

Once a valid configuration is submitted, the server updates the Warehouse by sending a *PlayerWarehouseMessage* relative to the calling player to every player.
If some resources were discarded during the process, the server updates all of the players' faith count and sends a number of *PlayerFaithBoardMessage* to each player, equal to the number of current player minus 1 (since the player that discards the resources doesn't get any faith) but if the list of resources acquired by the player contains faith, their faith counter is also updated and the total number of *PlayerFaithBoardMessage* sent to each player will be equal to the number of players in the game.