Models

User:   
Id – int  
Name – string  
ActiveSession – int (Id)

Friends:  
Id – int  
FriendId – int (id of the user that received the friend request)  
Status – FriendStatus (enum: value 0 – waiting, value 1 – accepted, value 2 – denied, value 3 – removed)  
DateAdded – datetime

GameSession:  
Id – int  
Users – List<int> (User ids)  
ActiveUntill - DateTime  
Data - GameServerData

GameServerData:  
Name – string  
IP – string  
Port – ushort  
Password – string (used for allowing access to a private server)  
Identifier – string (used for joining the server)

Message:

Id – int  
From – int (UserId)  
To – int (UserId)  
Status – bool (static MessageStatus class for )  
Content – string (The message itself)

**Communications**

Client To Server:

User Login(string username, string password);  
User Signup(string username, string password, string name);  
List<Friend> GetFriends();  
Friend AddFriend(int userId);  
void ChangeFriendStatus(int friendId, FriendStatus status);  
bool SendMessage(int toUserId, string message);  
// string returned is the identifier for this game session  
string CreateGameSession(string name, string ip, string port, string password);  
bool KeepGameSessionAlive(int sessionId, string identifier);  
List<GameSession> GetActiveGameSessions();  
void JoinGameSession(int sessionId, string identifier);

Server To Client:

Void MessageAccepted(int messageId);  
void NewMessage(Message message);  
void FriendStatusChanged(Friend friend, FriendStatus status);