**Untitled Sandbox Server**

[There should be these dumb things, ya know]

This is repository with the source code and releases of Untitled Sandbox Server.

What is this, actually? It is a server for my game Untitled Sandbox, check it out on Google Play (sorry for apple users)

Below you can read the instructions about using this.

**Instructions**

For normal people:

* Download the latest server release
* Extract downloaded ZIP file
* Open the main executable file
* Select “Settings” option and change anything in the way that you want.

They will not be reset on closing the program. More about settings is after this paragraph.

* Exit settings, and select “Server” option.
* Done! Server is running and you can connect by your IP adress.

**Settings**

Settings are:

* Authification

Enables or disables authification. Player will must register or login.

* Use banlist

Disables or enables ban function.

* Chat enabled

Disables or enables the chat.

* Anti cheat

In development. Currently does nothing.

**Additional**

For programmers: Below you can find there are all sorts of packets, that are being sended by the server or client. Also for debugging code you can select “Client” option.

For normal people: There is nothing below for you.

**Packets**

Packets arguments are separated with a comma.

Groups of packets are prefixes. Example: Disconnect,Banned

After sending some packets client will get a respnose by a packet with a similar name.

* Disconnect
  + ThisNameIsUsed (Server)

Sends if player with this name is connected already.

* + WrongPacket (Server)

Sends if player sended wrong packet.

* + IllegalPacket (Server)

Sends if player sended packet, that shouldn’t be sended by him.

* + Kicked (Server)

Sends if player was kicked.

* + Banned (Server)

Sends if player was banned.

* + IncorrectPassword (Server)

Sends if player entered wrong password.

* + RestrictedName (Server)

Sends if player tried to log in with ”Server” nickname.

* + CorruptedDatabase (Server)

Sends if players database is corrupted, and player tried to login or register.

* + LoginTimeout (Server)

Sends if player didn’t login in 1 minute period.

* Connect
  + Successfully (Server)

Sends if player connected successfully

* Player
  + GetPlayerData (Client)

Get player position and rotation.

Returns: ReturnData.

* + ReturnData (Server)

Packet with player’s position and rotation.

Arguments: PosX, PosY, PosZ, RotX, RotY, Rot Z.

* + SetPos (Client)

Sets player’s position.

Arguments: X, Y, Z.

* + SetRot (Client)

Sets player’s rotation.

Arguments: X, Y, Z.

* + GetHP (Client)

Get player HP.

Returns: ReturnHP packet with HP value.

* + ReturnHP (Server)

Packet with player’s position.

Arguments: HP.

* + SetHP (Client)

Sets player’s HP.

Arguments: HP.

* + Death (Server)

Triggers player’s death.

* + Respawn (Client)

Respawns player if dead. If not, sends Disconnect,IllegalPacket.

* + Join (Client)

Triggers player initialization.

Arguments: Name.

* + GetPlayers (Client)

Gives players list.

Returns: ReturnPlayers

* + ReturnPlayers (Server)

Packet with players list.

Arguments: Players list.

* + GetPlayerData (Client)

Gives player’s position and rotation.

Arguments: Player’s name.

Returns: ReterunPlayerData.

* + ReturnPlayerData (Server)

Packet with player’s position and rotation.

Arguments: PosX, PosY, PosZ, RotX, RotY, RotZ.

* Chat
  + SendMessage (Client)

Send message in chat.

Arguments: Message.

* + Receive (Server)

Packet with message.

Arguments: Player, Message.

* + Disabled (Server)

Packet with message that chat was disabled.

* + PlayerJoined (Server)

Packet with message that player joined.

Arguments: Player.

* + PlayerLeaved (Server)

Packet with message that player leaved.

Arguments: Player.

* + PlayerKicked (Server)

Packet with message that player was kicked.

Arguments: Player.

* + PlayerBanned (Server)

Packet with message that player was banned.

Arguments: Player.

* + LoginNow (Server)

Packet with message that player must login.

* Authification
  + Login (Client)

Log into an account.

Arguments: Password.

* + Register (Client)

Register an account.

Arguments: Password.

* + ChangePassword (Client)

Change password of an account.

Arguments: Old Password, New Password.

* GameObjects
  + Instantiate (Client)

Sends Instantiated packet to everyone.

Arguments: Prefab, PosX, PosY, PosZ, RotX, RotY, RotZ

Returns: Instantiated.

* + Instantiated (Server)

Packet with message that object was instantiated.

Arguments: ID, Prefab, PosX, PosY, PosZ, RotX, RotY, RotZ

* + GetObjectInstance (Client)

Gives object instance.

Arguments: ID.

Returns: ReturnObjectInstance.

* + ReturnObjectInstance (Server)

Packet with object instance.

Arguments: Object.

* + GetObjects (Client)

Gives objects IDs list.

* + ReturnObjects (Server)

Packet with objects IDs list.

Arguments: IDs.

* + UpdateObject (Client)

Update object’s data by ID.

Arguments: ID, PosX, PosY, PosZ, RotX, RotY, RotZ.

* + UpdatedObject (Server)

Packet with message that object was updated.

Arguments: ID, PosX, PosY, PosZ, RotX, RotY, RotZ.

* + RemoveObject (Client)

Removed object by ID.

Arguments: ID.

* + RemovedObject (Server)

Packet with message that object was destroyed.

Arguments: ID.

**TODO List**

* Implement anti-cheat
* Make world saving