

**POINTSHOP**



**TRADING  
SYSTEM**

**User Guide**

# Table of Contents

Pointshop Trading System .....	1
Installation .....	1
Configuration .....	2
Theme Customization .....	2
Core Settings .....	2
Language Strings .....	4
Developer Customization.....	4

## Pointshop Trading System

Pointshop Trading adds a great new way for players to swap, barter and exchange items complete with an invitation system, categorized inventory, drag and drop trade items, chat, a robust ready system and a huge set of configurable options to theme and customize trading for your server.

Best of all Pointshop Trading is compatible with Pointshop1, Pointshop2 **and now SH Pointshop servers** and works great across any gamemode.

This guide describes how to install, configure and customize every setting of the Pointshop Trading addon including detailed descriptions on each and every setting and developer examples for using the custom can trade function and logging hook.

### Installation

To install Pointshop Trading all you need to do is unzip the download (click Download on the Pointshop Trading System page after purchasing to get the zip file). Then you will need to upload the custom fonts to your FastDL server, there are two fonts located in: pointshoptrading/resource/fonts

# Configuration

To make the addon even easier to configure, all the settings can be managed in one file; `sh_tradingconfig.lua` In here you can manage the theme, core settings, language strings and notification sounds.

## Theme Customization

The first section of the config file allows you to customize a large number of theme settings, you can alter these to completely transform the look of your trade window.

Theme Setting	Details
<b>TRADING.Theme.WindowColor</b>	The trade window background color
<b>TRADING.Theme.ControlColor</b>	Color of controls within the trade window.
<b>TRADING.Theme.OutlineColor</b>	Color of outline for controls including buttons and the default item outline.
<b>TRADING.Theme.NotificationAccentColor</b>	Accent color for trade invitations.
<b>TRADING.Theme.NotificationErrorColor</b>	Accent color for trade errors and cancellations.
<b>TRADING.Theme.NotificationSuccessColor</b>	Accent color for trade sent message.
<b>TRADING.Theme.TradePointsIcon</b>	Icon used in the trade window to display traded points.
<b>TRADING.Theme.TradePointsOutlineColor</b>	Color of outline for traded points.
<b>TRADING.Theme.InventoryItemSize (125 Default)</b>	The maximum size of inventory and trade slots in the trade window (in pixels). This size will be scaled down for different resolutions.

## Core Settings

The second section of the config file allows you to customize the core settings and behaviour of the trading system addon.

Core Setting	Details
<b>TRADING.ChatCommands</b>	A table of chat commands that can be used to send a trade request to another player. If no name is specified the trade invitation window will be opened.
<b>TRADING.Settings.SelectionMenuFKey</b>	FKey setting to open the trade invitation window.
<b>TRADING.Settings.CursorFKey</b>	Useful for gamemodes with no cursor key, the cursor can be used to accept invitations if the accept and ignore FKeys are disabled.
<b>TRADING.Settings.AcceptLastTradeFKey</b>	FKey setting to accept the last active trade invitation.
<b>TRADING.Settings.IgnoreLastTradeFKey</b>	FKey setting to ignore the last active trade invitation.

<b>TRADING.Settings.NotificationSound (Default "ui/hint.wav")</b>	Sound effect to be played when a trade notification is displayed.
<b>TRADING.Settings.DefaultTradeSlots (Default 8)</b>	The default amount of empty trade slots to be displayed in the trade window. When the slots are nearly full, half of this amount will be added again.
<b>TRADING.Settings.TradeRequestCooldown</b>	Cooldown for sending new trade invitations (in seconds)
<b>TRADING.Settings.CanTradePoints</b>	Set to true or false to enable/disable trading points.
<b>TRADING.Settings.JoinNotification</b>	Set to true or false to enable/disable a join chat notification telling players how to send a trade request.
<b>TRADING.Settings.ExcludeCategories</b>	<p>A table of categories which are not allowed for trading. In Pointshop1 you must specify the category folder name e.g. {"weapons","trails"}.</p> <p>In Pointshop2 you must specify the category name as shown in the menu e.g. {"Player Models","Hats"}</p> <p>In SH Pointshop you must specify the category folder name or the category name in lower case e.g. {"weapons","trails"}.</p>
<b>TRADING.Settings.ExcludeItems</b>	<p>A table of items which are not allowed for trading. In Pointshop1 you must specify the item lua file name e.g. {"afro","texthat"}.</p> <p>In Pointshop2 you must specify the name as shown in the menu e.g. {"Bomber Hat","Steeze"}</p> <p>In SH Pointshop you must specify the Class Name of the item (you choose this when you create a new item) or the item lua file name for default items e.g. {"tophat","bucket"}.</p>
<b>TRADING.Settings.EnableBuiltInLogs</b>	<p>Set to true to enable the built in logging system. Detailed information about every trade will be output to data/pointshoptrading_logs/.</p> <p>Set to false if you want to use a third party logging system.</p>
<b>TRADING.Settings.MaximumItemsInTrade</b>	Set to 30 by default. Don't set this number much higher as larger trades can cause server issues.

## Language Strings

Pointshop Trading allows you to customize a large number of text strings that appear in the addon. By default, these strings are in English, in the last section of the config file you can customize the language in the addon by editing these text strings.

## Developer Customization

Pointshop Trading features a custom can trade function which developers can use to customize who can trade and which items can be traded based on their own code. An example of restricting trading to admins, superadmins and VIPs only is included in the config file.

<b>CustomCanTradeFunction</b>	<b>Use</b>
<b>TRADING.Settings.CustomCanTradeFunction = function(ply, tradingwith, item) end</b>	This function is called in two different places. When a trade invitation is being sent and when a new item is being added to the trade. The first two arguments pass the player and who the player is trading with (or sent a trade request to). The third argument is only given when a new item is being added and passes the item table from PS.Items in Pointshop1 and passes the item object in Pointshop2 from the players inventory. In SH Pointshop the item table from SH_POINTSHOP.Items is passed as the third argument.
<b>hook.Add("TradingLogs", "YourHookName", function(logtext, involvedplayers) end)</b>	The TradingLogs hook is called using hook.Run every time a log event occurs within Pointshop Trading. You can use this hook to bring Pointshop Trading logs into a third party or your own logging system.  Two variables will be passed to your hook function. The first is the full log text describing the players and their actions. The second is a table of the players involved in the action. The first player in this table being the one who initiated the action.

You can set  
TRADING.Settings.EnableBuiltInLogs =  
false when you use your own logging  
system to disable the built in log files.