Tailor Smooth System

Guide

Guide for Admin

Requirements

Before Tailor Smooth System, you will need to install and configure :

- [xp_aspectManager](#)

Installation of Tailor Smooth System

Nom

- Add_ToHak
- Script_Blueprint
- tailorsmoothsystem.erf

(The Script_Blueprint folder has the same content as the erf file, so we won't discuss it here.)

Import the erf to your module.
Add the contents of the various Add_ToHak folders to one of your Hak.

Now, you can add a Mannequin where you want:
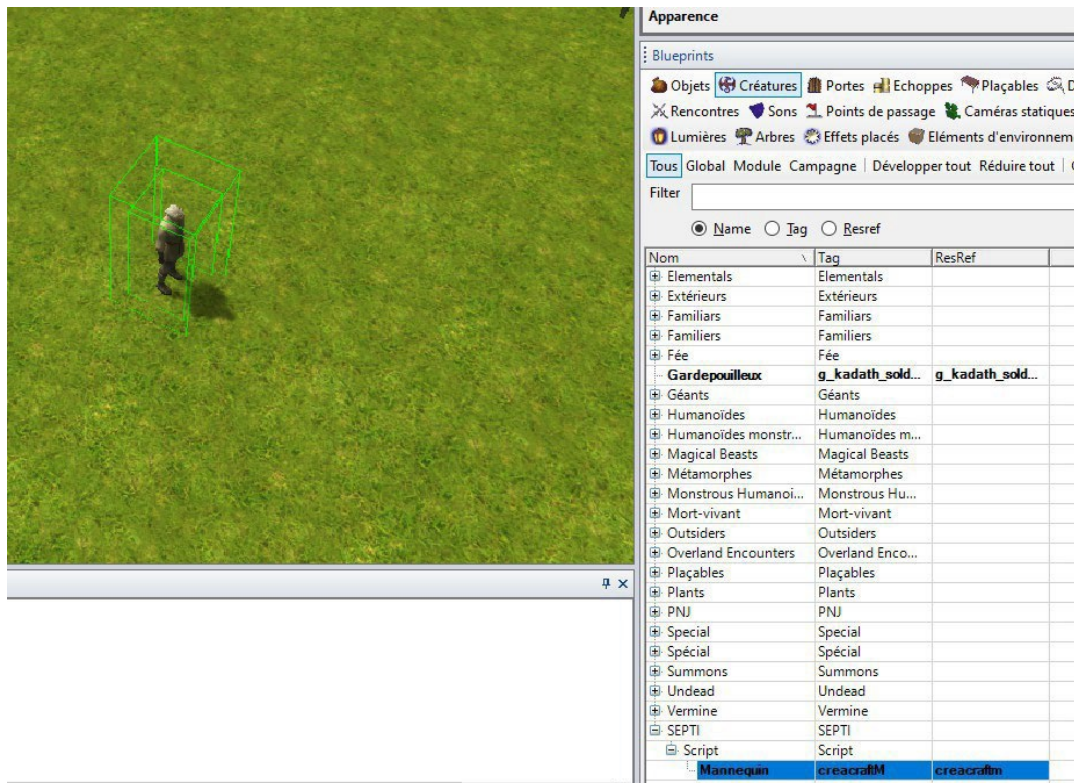
sept_lib_base_script.NSS / string GetPCUid_Sept (object)

If you use the Tailor Smooth System in your module you will have to change the *GetPCUid_Sept* function to match your UID system. Indeed it was designed to be used in the whole SEPTI project and will not be compatible with your module as if.

```
////////Modify this to call you own ID function ////////
string GetPCUid(object oPlayer)
{
    string sOutput;
    if(GetIsDM_C(oPlayer))
    {
        string str = "DMUID_"+GetPCPlayerName(oPlayer)+"_"+GetName(oPlayer);

        string sTranslate = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ_";
        while (GetStringLength(str) > 0)
        {
            int iTrans = FindSubString(sTranslate, GetStringLeft(str,1));
            if(iTrans != -1)
            {
                sOutput = sOutput + GetStringLeft(str,1);
            }

            str = GetStringRight(str, GetStringLength(str)-1);
        }
    }
    else
    {
        sOutput = GetLocalString(GetSeptiBaseItem(oPlayer), SEPTI_PCUID);
    }

    return sOutput;
}
```

Remove the content of this function and replace it with a call to your own UID function.

Extra Configurations

Translations

In order to help translation, every text used by the system (including the ones in GUI) are stored in sept_tss_translation.NSS. The package came with some translation (sept_tss_translation_french , _german and spanish.NSS). Just replace the content of the base file by the content of the one you want. Nothing more to do. No XML to change. No other script to read. Easy isn't it ?

Configurations

In case you want to change some behavior of TSS, several configurable functions have been grouped in the "sept_tss_configuration" file.
This allows you to adapt the system to some of your needs without difficulty. Some basic code behaviors and examples are already provided in it.

Let me present some of these functions:
        int Sept_TSS_AllowedToViewVisualType(string sVisualType, object oPC, int iSlot);

This function enables you to hide certain categories from the PC. For instance, if you want to restrict some categories to be used only by the DM, or reserve specific appearances for talented tailors, this function can be utilized.

        TSS_IsBracerOk()
To change Bracers as if they were gloves, you need to modify the baseitem.2DA bracer line and set the modelType to 3. If you don't want to allow Bracer changes, simply return FALSE using this function.

        TSS_IsAppearanceAllowRemoveVariation
Some vault weapons place the 3 parts on the weaponPart 1, resulting in poor superimposed views when used. You can allow your players to set some weapon parts to "invisible" by configuring this function. It can also allow to hide the belt.

All hips in one button

By default, in order to avoid change in generic files, TSS use two button for hips (one for left hip, the other for right hip). We recommend to use the alternative, the all-in-one button. That will allow front and back hip too.

For that you will need to edit your sept_tailorsmoothsystem.xml file.

From that:

```xml
<!--   To use the generic "HIPS" :   Start Comment here. -->
    <UIButton name="ARMOR_RIGHT_HIP" x="60" y="160" width="30" height="30" update="true"
        OnLeftClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","100","RHip")'
        OnRightDoubleClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","101","RHip")'
        OnToolTip='UIObject_Tooltip_DisplayTooltipString(local:117,OBJECT_X,OBJECT_Y,SCREEN_TOOLTIP_2,ALIGN_NONE,ALIGN_NONE,0,0,ALIGN_LEFT)'>
        <UIFrame state="base" fill="it_am_breastplate00.tga" />
        <UIFrame state=up       fill="b_empty.tga" />
        <UIFrame state=down     fill="b_overlay.tga" />
        <UIFrame state=focused  fill="b_empty.tga" />
        <UIFrame state=hilited  fill="b_overlay.tga" />
        <UIFrame state=hifocus  fill="b_overlay.tga" />
        <UIFrame state=disabled fill="b_empty.tga" />
    </UIButton>

    <UIButton name="ARMOR_LEFT_HIP" x="100" y="160" width="30" height="30" update="true"
        OnLeftClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","100","LHip")'
        OnRightDoubleClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","101","LHip")'
        OnToolTip='UIObject_Tooltip_DisplayTooltipString(local:118,OBJECT_X,OBJECT_Y,SCREEN_TOOLTIP_2,ALIGN_NONE,ALIGN_NONE,0,0,ALIGN_LEFT)'>
        <UIFrame state="base" fill="it_am_breastplate01.tga" />
        <UIFrame state=up       fill="b_empty.tga" />
        <UIFrame state=down     fill="b_overlay.tga" />
        <UIFrame state=focused  fill="b_empty.tga" />
        <UIFrame state=hilited  fill="b_overlay.tga" />
        <UIFrame state=hifocus  fill="b_overlay.tga" />
        <UIFrame state=disabled fill="b_empty.tga" />
    </UIButton>
<!--   To use the generic "HIPS" :   End comment here  -->

<!-- To use generic "HIPS" remove comment here --
    <UIButton name="ARMOR_RIGHT_HIP" x="65" y="160" width="60" height="30" update="true"
        OnRightDoubleClick='UIObject_OnRadial_DisplayCustomRadial("root-remhiparmorpart")'
        OnLeftClick='UIObject_OnRadial_DisplayCustomRadial("root-hiparmorpart")'
        OnToolTip='UIObject_Tooltip_DisplayTooltipString(local:116,OBJECT_X,OBJECT_Y,SCREEN_TOOLTIP_2,ALIGN_NONE,ALIGN_NONE,0,0,ALIGN_LEFT)'>
        <UIFrame state="base" fill="b_hips.tga" />
        <UIFrame state=up       fill="b_empty.tga" />
        <UIFrame state=down     fill="b_overlay.tga" />
        <UIFrame state=focused  fill="b_empty.tga" />
        <UIFrame state=hilited  fill="b_overlay.tga" />
        <UIFrame state=hifocus  fill="b_overlay.tga" />
        <UIFrame state=disabled fill="b_empty.tga" />
    </UIButton>
    To use generic "HIPS", remove comment here THEN change the contextmenu.xml according to the doc -->
```

To that :

```xml
<!--   To use the generic "HIPS" :   Start Comment here. |
    <UIButton name="ARMOR_RIGHT_HIP" x="60" y="160" width="30" height="30" update="true"
        OnLeftClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","100","RHip")'
        OnRightDoubleClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","101","RHip")'
        OnToolTip='UIObject_Tooltip_DisplayTooltipString(local:117,OBJECT_X,OBJECT_Y,SCREEN_TOOLTIP_2,ALIGN_NONE,ALIGN_NONE,0,0,ALIGN_LEFT)'>
        <UIFrame state="base" fill="it_am_breastplate00.tga" />
        <UIFrame state=up       fill="b_empty.tga" />
        <UIFrame state=down     fill="b_overlay.tga" />
        <UIFrame state=focused  fill="b_empty.tga" />
        <UIFrame state=hilited  fill="b_overlay.tga" />
        <UIFrame state=hifocus  fill="b_overlay.tga" />
        <UIFrame state=disabled fill="b_empty.tga" />
    </UIButton>

    <UIButton name="ARMOR_LEFT_HIP" x="100" y="160" width="30" height="30" update="true"
        OnLeftClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","100","LHip")'
        OnRightDoubleClick='UIObject_Misc_ExecuteServerScript("gui_sept_craft_esthe","101","LHip")'
        OnToolTip='UIObject_Tooltip_DisplayTooltipString(local:118,OBJECT_X,OBJECT_Y,SCREEN_TOOLTIP_2,ALIGN_NONE,ALIGN_NONE,0,0,ALIGN_LEFT)'>
        <UIFrame state="base" fill="it_am_breastplate01.tga" />
        <UIFrame state=up       fill="b_empty.tga" />
        <UIFrame state=down     fill="b_overlay.tga" />
        <UIFrame state=focused  fill="b_empty.tga" />
        <UIFrame state=hilited  fill="b_overlay.tga" />
        <UIFrame state=hifocus  fill="b_overlay.tga" />
        <UIFrame state=disabled fill="b_empty.tga" />
    </UIButton>
    To use the generic "HIPS" :   End comment here  -->

<!-- To use generic "HIPS" remove comment here -->
    <UIButton name="ARMOR_RIGHT_HIP" x="65" y="160" width="60" height="30" update="true"
        OnRightDoubleClick='UIObject_OnRadial_DisplayCustomRadial("root-remhiparmorpart")'
        OnLeftClick='UIObject_OnRadial_DisplayCustomRadial("root-hiparmorpart")'
        OnToolTip='UIObject_Tooltip_DisplayTooltipString(local:116,OBJECT_X,OBJECT_Y,SCREEN_TOOLTIP_2,ALIGN_NONE,ALIGN_NONE,0,0,ALIGN_LEFT)'>
        <UIFrame state="base" fill="b_hips.tga" />
        <UIFrame state=up       fill="b_empty.tga" />
        <UIFrame state=down     fill="b_overlay.tga" />
        <UIFrame state=focused  fill="b_empty.tga" />
        <UIFrame state=hilited  fill="b_overlay.tga" />
        <UIFrame state=hifocus  fill="b_overlay.tga" />
        <UIFrame state=disabled fill="b_empty.tga" />
    </UIButton>
```

Save your file, then add the content of "all-hip-in-one-contextmenu.txt" at the end of your contextmenu.xml.
/!\ In this file (all-hip-in-one-contextmenu.txt), and this file only you will need to translate without the nss file

Manage your models :

This can be see and use only by DM

Choose the gender, race and modelPart (helm, boots, gloves, cloak, belt or armor) filter.

List of VisualCategory matching the filters above. Select the one you want to work on. The two button next to it will set/unset the restricted flag for it (eye/cross eye icon) or remove this whole VisualCategory if possible (red X).

With those, you can copy the whole "from VisualCategory" (upper arrow) the selected variation (lower arrow) to the Destination.

Destination VisualCategory for the arrows select an existing one on top, or just "create" a new one by using the lower part.

List of Variations for the selected "from VisualCategory" Will ungrey and fill only if you don't have any "All" in the filters part.

List of VisualCategory containing the selected "From Variation" click on the red X will remove the "From Variation" from the selected one.

The floppy disk is to make the change persistent (else everything will go back to "normal" after reboot).

The reload, will reset your change, using the last saved configuration.

As you can see, you can create the Category that suit you. Mixing and/or splitting different basic categories.

For example: here I have copied the whole Hide VisualType to the new VisualType "MyCat" and then add to it the SC_01 from Scale VisualType

Restricted VisualType :

The restricted VisualType is a quick way to put a full VisualType name not visible by PC (or other configuration wanted by edit the Sept_TSS_AllowedToViewVisualType function)

User Guide

How to use the Tailor Smooth System.

Basic usage

When you talk to the Mannequin, it will copy your specie, gender, and equipment.

When you have clicked on the icon of the item you want to change, it will open this panel.

You can select appearance and change the colors (with the bars or directly hex code). This will be reflected on the Mannequin so you can preview it.

When you are happy, you can click the validation button and it will apply on you. If not, just click the cancel button and the mannequin will go back to your current outfit.

This panel is for the armor part. If you uncheck the checkbox, this will remove it.

The weapon panel (it can have only one part for some weapons).

Change the Mannequin Specie and or Gender.

Just choose what you want on the list or select button, the Mannequin will be change.
The species listed are the one with specific models. If you don't find the one you search, it will probably be Human.