

Hud Updating Guide

By Hypnotize (° ͜ °)

Before we start I recommend using notepad++ for this process and for hud editing in general!

This guide should cover all the major updates at least from 2012 (maybe 2011) to current day!

This is what I usually do to update any hud!

Mandatory Updates:

- | | |
|---|--------------------------------|
| 1. Add the Info.vdf file to the hud | (MYM/Jungle Update 2016 -2017) |
| 2. Update the HudAnimations_tf | (relase – current day) |
| 3. Update the HudLayout.res | (relase – current day) |
| 4. Update the Clientscheme.res | (relase – current day) |
| 5. Remove all the outdated files that might cause crashes | (Jungle Inferno Update 2017) |
| 6. Fix the Main Menu | (MYM/Jungle Update 2016 -2017) |
| 7. Add the support for the new war paints menu | (Jungle Inferno Update 2017) |
| 8. Fix the new item pickup panel (only if needed) | (Jungle Inferno Update 2017) |
| 9. Fix the Casual/Competitive UI and Timer | (MYM Update 2016) |
| 10.Fix the Tournament Hud | (MYM Update 2016) |
| 11.Fix the WinPanel.res | (MYM Update 2016) |
| 12.Fix the Scoreboard | (MYM Update 2016) |
| 13. Fix The Vaccinator Hud | (Mecha Update 2012) |

Non Mandatory Updates:

1. Fix the CharInfoPanel.res and StorePanel.res (only if needed)
2. Add the missing status icons to the HudPlayerHealth.res
3. Fix the Spectator.res/SpectatorTournament.res Item Panels
4. Fix the Floating Health cut-off problems
5. Add a killstreak icon if missing
6. Add a 3D Player Model if missing
7. Add the new weapons Meters and Counters

Non Mandatory And Not Covered Because They Are Too Specific On The Hud Style:

Just for anyone that has the experience to edit the files alone and wants the hud to be **p e r f e c t**

1. Add the Taunt and Eureka Effect menu to match the rest of the hud
2. Adapt the new Casual/Competitive/MvM/Training/Ping Menus to match the rest of the hud
3. Add the Halloween updates stuff to match the rest of the hud

1. ADDING THE INFO.VDF

Let's open the "Update Files" folder, here you should find a file called info.vdf, now copy it and paste it to the hud you want update.

Next and last thing we need to do is open the info.vdf file through notepad++ and change the name of it to match the name of the hud folder.

For example, if I'm updating Broesel Hud it will be:

```
"TYPE THE NAME OF YOUR HUD FOLDER INSTEAD OF THIS TEXT"  
{  
  "ui_version"    "3"  
}
```



```
"Broesel Hud"  
{  
  "ui_version"    "3"  
}
```

2. UPDATING HUDANIMATIONS

The first thing we need to do is add to our outdated hud the HudAnimations_Manifest file!

- Go to the “Update Files” folder
- Copy The HudAnimations_Manifest file
- Paste it inside the scripts folder of the hud we are updating

So now inside the scripts folder we should have 2 animations files

- HudAnimations_Manifest
- HudAnimations_tf

The next thing we need to do is rename the HudAnimation_tf to > HudAnimations_Custom just so it matches what’s inside the HudAnimations_Manifest we just added.

The last thing we need to do is removing all the unneeded codes inside the HudAnimations_Custom.

We can pretty much delete everything BUT these sections:

- Health Bonus Pulse
- Health Dying Pulse
- HudLowAmmoPulse
- FlagOutlineHide *(this might be unneeded but some huds edit this)*
- Spy Disguise *(this might be unneeded but some huds edit this)*
- HudMedicCharged
- And the HudTournamentSetupPanelOpen/Close if it’s different from the default hud

If you have doubts on what you can delete and what not you can compare your hud animations with the default hud ones, you want to delete every code that looks exactly the same as the default one.

(You can also find an example of what it should look like after updated inside the “Update Files” folder)

We are done with the animations, now this file should update completely alone for the future!

3. UPDATING THE HUDLAYOUT

All we need to do with this file is

- Launch our game
- Open the ingame console
- Look for “hud element” “name” “doesn’t have an entry” errors

They should look like this

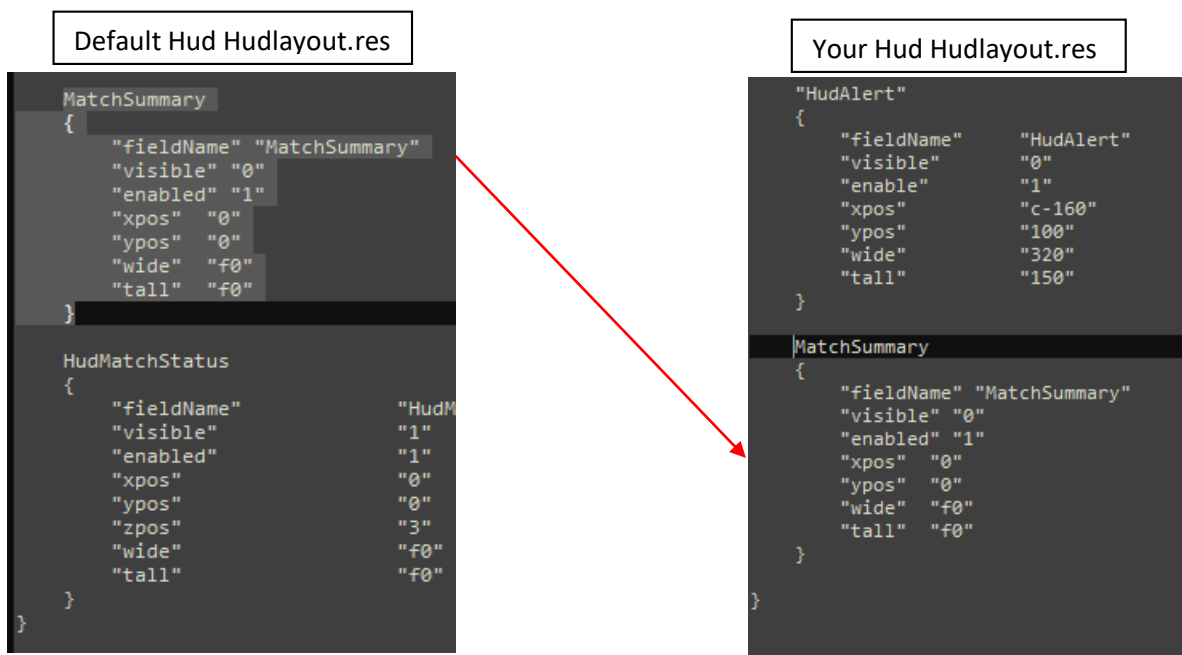
```
Hud element 'CTFMatchSummary' doesn't have an entry 'MatchSummary' in scripts/HudLayout.res
Hud element 'CQuestNotificationPanel' doesn't have an entry 'QuestNotificationPanel' in scripts/HudLayout.res
Hud element 'CTFHudSpectatorExtras' doesn't have an entry 'HudSpectatorExtras' in scripts/HudLayout.res
Hud element 'CHudMiniGame' doesn't have an entry 'HudMiniGame' in scripts/HudLayout.res
Hud element 'CHudMenuTauntSelection' doesn't have an entry 'HudMenuTauntSelection' in scripts/HudLayout.res
Hud element 'CTFHudMatchStatus' doesn't have an entry 'HudMatchStatus' in scripts/HudLayout.res
Hud element 'CHudItemAttributeTracker' doesn't have an entry 'ItemAttributeTracker' in scripts/HudLayout.res
Hud element 'CTFFlagCalloutPanel' doesn't have an entry 'CTFFlagCalloutPanel' in scripts/HudLayout.res
Hud element 'CTFStreakNotice' doesn't have an entry 'CTFStreakNotice' in scripts/HudLayout.res
```

So let’s take as example the first error line.

It says that “MatchSummary” doesn’t have an entry in our HudLayout.

What we need to do is:

1. Go to the “Update Files” folder and open the HudLayout
2. Search for “MatchSummary”
3. Copy the whole code and paste it to our HudLayout
4. Pasting it the end of the file is fine



Then all you need to do is repeat this for all the errors you get!

If you do it correctly when you’ll launch the game again the console shouldn’t display any of these errors!

4. CLIENTSCHEME

In this file we need to add the new:

- Colors
- Fonts
- Borders

So I got I pretty accurate list of every color – font – border that has been added in the updates (should cover everything from 2011/12 to 2018) the Clientscheme.res we need to update might already have most of them depending on how old it is!

Let's start with opening the "Colors-Fonts-Borders you need to add" file which you can find inside the "Update Files" folder as always.

What we need to do is:

- Select all the colors, from "Purple" to "QualityColorPaintkitWeapon_GreyedOut"
- Add them to our ClientScheme.res
- We can add them at the end of the colors section, that means right after "UpgradeDisabledBg"

Now the same thing for the Fonts and Borders

- Select all the fonts
- Add them at the end of the fonts section, right after "EconFontMedium" or "ControllerHintText" depending on how old the hud is.

For the borders

- Select all the Border
- Add them at the end of the borders section, right after
ReplayBrowser.ScrollBar.SliderButton.Border

Now there is a last small thing we need to do which is replacing the backpack and loadout outdated borders:

- Let's open the "Borders you need to replace" file inside the "Update Files" Folder
- Select everything in there and copy it (from "EconItemBorder" to "BackpackItemGreyedOutSelectedBorder_RarityAncient" just to be clear)
- Then we move to our ClientScheme.res and search for "EconItemBorder"
- Once again select everything from "EconItemBorder" and stop before you hit "StoreItemBorder"
- Now simply replace everything you have selected with the borders you have copied before.

5. REMOVING USELESS OUTDATED FILES

Let's now remove all the files that became useless with the new update and might cause crashes!

The files we need to remove are:

- Any file that starts with "matchmaking..."
- LobbyContainerFrame.res and LobbyContainerFrame_Casual / Comp / MvM
- LobbyPanel.res and LobbyPanel_Casual / Comp / MvM
- MainMenuPlayListEntry.res (if present and outdated makes the game crash 100% of the times)

6. MAINMENU

IF YOU DON'T REALLY CARE ABOUT HAVING A CUSTOM MAIN MENU THE EASIEST WAY IS TO REMOVE GAMEMENU.RES AND MAINMENU.RES AND SKIP THIS SECTION!

This is for sure the hardest part of this update as it includes a lot of editing based on the hud we want to update. Every menu is different and needs different customizations so I'll explain this in the most generic way possible.

First thing we need to do is removing the Casual, Comp, MvM buttons that became useless after the jungle inferno update

To do that we need to edit both MainMenuOverride.res and GameMenu.res and remove all the codes related to the buttons.

I really suggest to start from the GameMenu.res and identify the three buttons in question, in this case:

```
"CasualButton"
{
    "label"                "CASUAL"
    "command"              "play_casual"
    "OnlyAtMenu"          "1"
}
"MvMButton"
{
    "label"                "MVM"
    "command"              "play_mvm"
    "OnlyAtMenu"          "1"
}
"CompetitiveButton"
{
    "label"                "COMPETITIVE"
    "command"              "play_competitive"
    "OnlyAtMenu"          "1"
}
```

A lot of times they might have different names, but usually we need to get rid of any button called:

- QuickPlayButton / QuickPlayChangeButton / CasualButton
- PlayPVEButton / MvMButton
- CompetitiveButton

After we know what buttons needs to be removed we can delete the code from both GameMenu.res and MainMenuOverride.res, THEY ALWAYS HAVE THE SAME NAME IN BOTH FILES!

Fixing possible crashes:

YOU CAN IGNORE THIS IF YOU DON'T GET ANY CRASHES BASICALLY

Now if you get a game crash while joining a server is probably because inside the GameMenu.res there is a button that isn't defined inside the MainMenuOverride.res

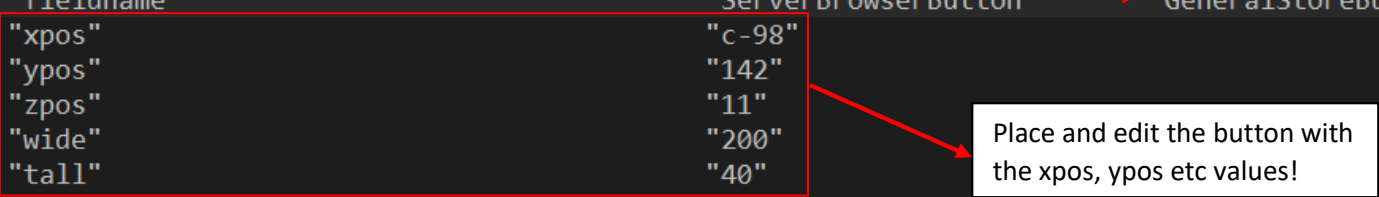
So we can either

- Delete it if we don't need it (like explained before)
- Define it inside the MainMenuOverride.res if it's a button that we actually want to be part of our main menu

Let's take as example the Store Button, there might be a case where it is not defined inside the MainMenuOverride.res and makes the game crash, for me the easiest way to fix that is to create a new button that is literally the same as any other menu button, for example the "ServerBrowserButton", copy and paste it to create an identical button and then change his name to "GeneralStoreButton" and then work with the xpos, ypos and all the other values to bring it in position.

```
"ServerBrowserButton" → "GeneralStoreButton"
{
  "ControlName"          "EditablePanel"
  "fieldname"            "ServerBrowserButton" → "GeneralStoreButton"
  "xpos"                  "c-98"
  "ypos"                  "142"
  "zpos"                  "11"
  "wide"                  "200"
  "tall"                  "40"
  "visible"              "1"

  "SubButton"
  {
```



Place and edit the button with the xpos, ypos etc values!

Now let's add the **new stuff** to the menu!

What we need to do is:

- Go to the "Update Files" folder and open Additions to MainMenuOverride
- Copy and paste everything to our MainMenuOverride.res
- You can paste the codes wherever you want, after "SaxxySettings" is fine!

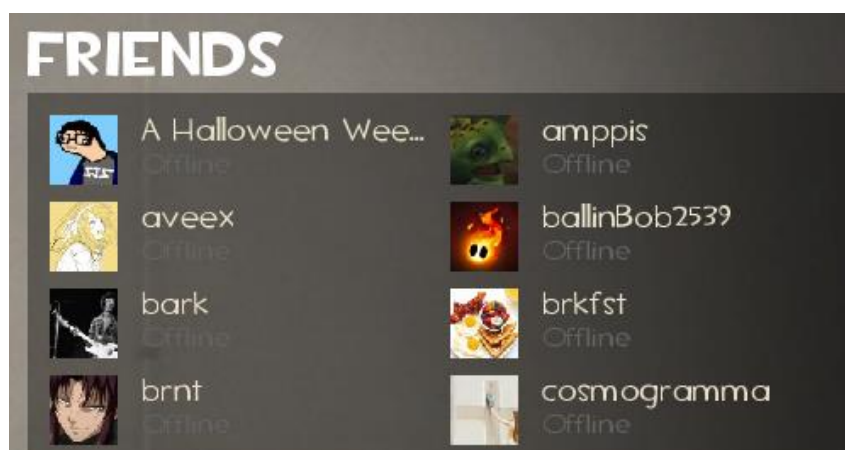
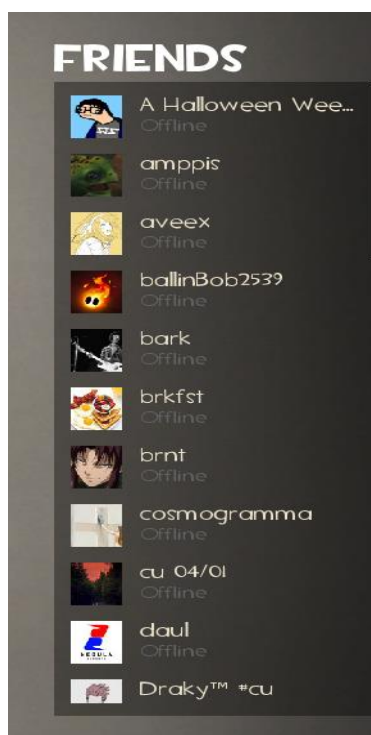


By importing these codes, we added:

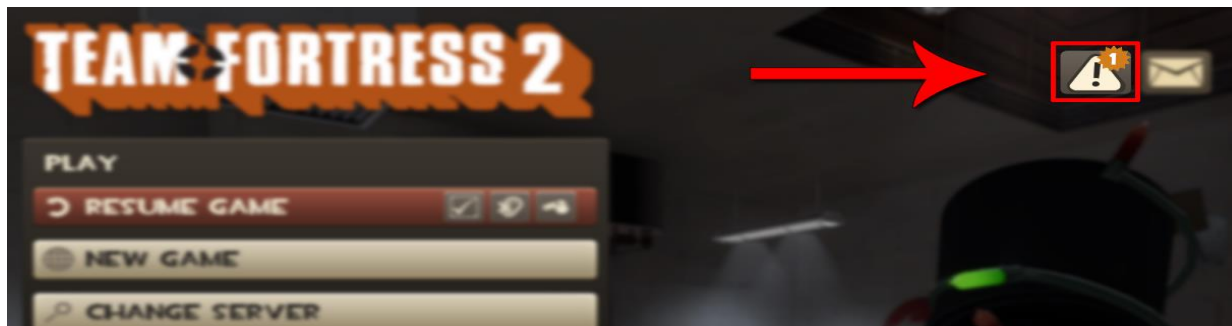
- Twitch streams button ("WatchStreamButton")
- Contracker Button ("QuestLogButton")
- Steam FriendList ("FriendsContainer")
- Name Rank and Level ("RankPanel" and can be edited by using PvPRankPanel.res)
- Rank Badge ("RankModelPanel")
- Casual/Comp toggle button ("CycleRankTypeButton")

You can edit the xpos, ypos and the other values for each of these codes in order to get them wherever you want in the menu and change their look!

Sadly, each menu is different so I can't really help on that, but I'll leave these 2 presets for the Steam Friendlist in the Update Files/Friend List Presets folder, both work by default with any hud!



Next thing is **fixing the Notification and MOTD buttons.**



Let's open our MainMenuOverride.res and what we need to do is add a new line to the MOTD and Notification codes that are already there.

So what we need to add is **"actionsignallevel" "2"**

Now inside our MainMenuOverride.res let's search for Notifications_ShowButtonPanel_SB and just after "default" "1" we add the new "actionsignallevel" "2" Line.

It should then look something like this:

```
"Notifications_ShowButtonPanel_SB"
{
    "ControlName"    "CEImageButton"
    "fieldName"      "Notifications_ShowButtonPanel_SB"
    "xpos"           "0"
    "ypos"           "0"
    "zpos"           "1"
    "wide"           "32"
    "tall"           "32"
    "autoResize"     "0"
    "pinCorner"       "3"
    "visible"         "1"
    "enabled"         "1"
    "tabPosition"    "0"
    "labelText"       ""
    "font"           "HudFontSmallestBold"
    "textAlignment"  "center"
    "dulltext"        "0"
    "brighttext"     "0"
    "default"         "1"
    "actionsignallevel" "2"
```

Now we need to repeat this for these other codes:

- Notifications_ShowButtonPanel_SB
- MOTD_ShowButtonPanel_SB
- MOTD_CloseButton
- MOTD_URLButton
- MOTD_PrevButton
- MOTD_NextButton
- Notifications_CloseButton

If you did that correctly, you should now be able to click on the Notification and MOTD buttons successfully.

7. WAR PAINTS SUPPORT

First thing to do is check if our hud has these two files:

- CharInfoLoadoutSubPanel.res
- ui/econ/InspectionPanel.res

If these two files are not in the hud you are updating, then you can skip this section!

While if they are preset we need to delete or rename InspectionPanel.res and update CharInfoLoadoutSubPanel.res

To fix the CharInfoLoadoutSubPanel.res we need to do 2 things. First replace the backpack, craft, trade and catalog buttons plus add the new war paint button, to do that we need to:

- Open the Additions to CharInfoLoadoutSubPanel File (we can find it inside the Update Files Folder!)
- Select and copy from ShowBackpackButton to ShowPaintkitsLabel (included)
- Open our huds CharInfoLoadoutSubPanel.res
- Select from ShowBackpackButton to ShowTradeLabel and replace everything with the text with selected before

Second and last thing, we need to add the “InspectionPanel” section to our CharInfoLoadoutSubPanel.res as well or the war paints menu is not going to show up.

We can find the “InspectionPanel” section at the end of the file, all we need to do is take it from the Additions to CharInfoLoadoutSubPanel file and add it at the end of our huds CharInfoLoadoutSubPanel.res (after armory_panel)

8. FIXING THE ITEM PICKUP PANEL

This fix is only needed if we have a custom item pickup panel, the file in question can be found inside: `Resource/ui/econ/ItemPickupPanel.res` and `ItemDiscardPanel.res`

If these two files are not present, then you can skip this!

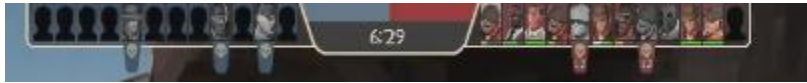
This is pretty easy to fix; all we need to do is:

- Open `ItemPickupPanel.res`
- Change the “zpos” from “100000” to “10000”
- Scroll down until we find "ItemsFoundLabel"
- Change the “ypos” from “20” to "30"

While for the `ItemDiscardPanel.res` we need to

- Add the “zpos” “10000” to "item_discard" (right after the ypos is fine)
- Scroll down "modelpanel" and change the “ypos” from “20” to “30”

9. CASUAL-COMPETITIVE UI AND TIMER



First of all, we need a new file “HudMatchStatus.res” so let’s go to the “Update Files” folder, copy it and paste it to our hud.

Now on the hud we are updating there should be a file called HudObjectiveStatus.res (if you don’t have it the timer is either the default one or in already updated)

Let’s open it and copy the whole code in it (first screen), then open the HudMatchStatus.res we just added and look for “ObjectiveStatusPanel”.

Now all we need to do is select that “ObjectiveStatusTimePanel” and replace it with the one we copied before. As you can see from the screenshots.

HudObjectiveStatus.res

HudMatchStatus.res

HudMatchStatus.res

```
"Resource/UI/HudObjectiveStatus.res"
{
  "ObjectiveStatusTimePanel"
  {
    "ControlName" "EditablePanel"
    "fieldName" "ObjectiveStatusTimePanel"
    "xpos" "c-200"
    "ypos" "0"
    "zpos" "1"
    "wide" "400"
    "tall" "350"
    "visible" "0"
    "enabled" "1"
    "delta_item_x" "100"
    "delta_item_start_y" "-3"
    "delta_item_end_y" "3"
    "PositiveColor" "0 255 0 255"
    "NegativeColor" "255 0 0 255"
    "delta_lifetime" "3"
    "delta_item_font" "surface28"

    "TimePanelValue"
    {
      "ControlName" "CExLabel"
      "fieldName" "TimePanelValue"
      "font" "surface36shadow"
      "fgcolor" "255 255 255 255"
      "xpos" "0"
      "ypos" "-4"
      "zpos" "3"
      "wide" "400"
      "tall" "36"
      "visible" "1"
      "enabled" "1"
      "textAlignment" "center"
      "labelText" "0:00"
    }
  }
}
```

```
if_match
{
  "visible" "1"
}

"ObjectiveStatusTimePanel"
{
  "ControlName" "EditablePanel"
  "fieldName" "ObjectiveStatusTimePanel"
  "xpos" "c-55"
  "xpos_hidef" "c-150"
  "xpos_lodef" "c-150"
  "ypos" "0" [SWIN32]
  "ypos_minmode" "-14" [SWIN32]
  "ypos" "24" [X360]
  "zpos" "2"
  "wide" "110"
  "wide_hidef" "300"
  "wide_lodef" "300"
  "tall" "150"
  "visible" "0"
  "enabled" "1"
  "delta_item_x" "22" [SWIN32]
  "delta_item_x" "115" [X360]
  "delta_item_start_y" "50"
  "delta_item_end_y" "70"
  "PositiveColor" "0 255 0 255"
  "NegativeColor" "255 0 0 255"
  "delta_lifetime" "1.5"
  "delta_item_font" "HudFontMedium"

  "TimePanelValue"
  {
    "ControlName" "CExLabel"
    "fieldName" "TimePanelValue"
    "font" "HudFontMediumSmall"
    "font_minmode" "HudFontSmall"
    "font_lodef" "HudFontMedium"
    "fgcolor" "Tanlight"
    "xpos" "23"
    "xpos_minmode" "39"
    "xpos_hidef" "114"
    "xpos_lodef" "114"
    "ypos" "11"
    "ypos_minmode" "6"
    "ypos_hidef" "15"
    "ypos_lodef" "18"
    "zpos" "3"
    "wide" "45"
    "wide_minmode" "30"
    "wide_lodef" "50"
    "tall" "31"
    "visible" "1"
    "enabled" "1"
    "textAlignment" "center"
    "labelText" "0:00"
  }
}

"TeamStatus"
{
  "ControlName" "CTFTeamStatus"
  "fieldName" "TeamStatus"
  "xpos" "0"
```

```
if_match
{
  "visible" "1"
}

"ObjectiveStatusTimePanel"
{
  "ControlName" "EditablePanel"
  "fieldName" "ObjectiveStatusTimePanel"
  "xpos" "c-200"
  "ypos" "0"
  "zpos" "1"
  "wide" "400"
  "tall" "350"
  "visible" "0"
  "enabled" "1"
  "delta_item_x" "100"
  "delta_item_start_y" "-3"
  "delta_item_end_y" "3"
  "PositiveColor" "0 255 0 255"
  "NegativeColor" "255 0 0 255"
  "delta_lifetime" "3"
  "delta_item_font" "surface28"

  "TimePanelValue"
  {
    "ControlName" "CExLabel"
    "fieldName" "TimePanelValue"
    "font" "surface36shadow"
    "fgcolor" "255 255 255 255"
    "xpos" "0"
    "ypos" "-4"
    "zpos" "3"
    "wide" "400"
    "tall" "36"
    "visible" "1"
    "enabled" "1"
    "textAlignment" "center"
    "labelText" "0:00"
  }
}

"TeamStatus"
{
  "ControlName" "CTFTeamStatus"
  "fieldName" "TeamStatus"
  "xpos" "0"
```

Next thing we are going to do is getting the timer in position:



To do that we need to add we need to add the “if_match” definitions to our timer codes

```
"ObjectiveStatusTimePanel"
{
  "ControlName"      "EditablePanel"
  "fieldName"        "ObjectiveStatusTimePanel"
  "xpos"              "c-200"
  "ypos"              "0"
  "zpos"              "1"
  "wide"              "400"
  "tall"              "350"
  "visible"           "0"
  "enabled"           "1"
  "delta_item_x"      "100"
  "delta_item_start_y" "-3"
  "delta_item_end_y"  "3"
  "PositiveColor"      "0 255 0 255"
  "NegativeColor"      "255 0 0 255"
  "delta_lifetime"     "3"
  "delta_item_font"    "surface28"

  if_match
  {
    "zpos"              "5"
  }

  "TimePanelValue"
  {
    "ControlName"      "CExLabel"
    "fieldName"        "TimePanelValue"
    "font"              "surface36shadow"
    "fgcolor"           "255 255 255 255"
    "xpos"              "0"
    "ypos"              "-4"
    "zpos"              "3"
    "wide"              "400"
    "tall"              "36"
    "visible"           "1"
    "enabled"           "1"
    "textAlignment"     "center"
    "labelText"         "0:00"

    if_match
    {
      "font"            "HudFontSmall"
      "ypos"            "-1"
    }
  }
}
```

The “if_match” on the first section is fundamental to make the timer be above the UI and not under it. So always add it no matter what.

The second one is where we actually edit the timer placement and the font size.

If the font is too big/too small and you are unsure to what font you can use I recommend using “HudFontSmall” as it is the default font and always work, as shown in the screen.

If the timer is off centered, you need to add to the “if_match” section the “xpos” and “ypos” and work with them.

As always if the timer is a bit too much to the left or right you need to work with the “xpos”, if it’s too high or low you will work with the “ypos”.

Now you can just join a casual server and fix it using the hud_reloadscheme command every time you make a change!

We need to do the same thing for the King of the hill timers

But this time we need to work with HudObjectiveKothTimePanel.res

As before add the “if_match” sections and work with the fonts-positioning values to get the timers in positions as shown there:

```
"BlueTimer"
{
    "ControlName"        "CTFHudTimeStatus"
    "fieldName"          "BlueTimer"
    "xpos"                "0"
    "ypos"                "0"
    "zpos"                "2"
    "wide"                "100"
    "tall"                "50"
    "visible"             "1"
    "enabled"             "1"

    "TimePanelValue"
    {
        "ControlName"    "CExLabel"
        "fieldName"      "TimePanelValue"
        "font"            "surface14"
        "fgcolor"         "255 255 255 255"
        "xpos"            "0"
        "ypos"            "0"
        "zpos"            "3"
        "wide"            "100"
        "tall"            "13"
        "visible"         "1"
        "enabled"         "1"
        "textAlignment"   "center"
        "labelText"       "0:00"
    }
}
```

```
"HudKothTimeStatus"
{
    if_match
    {
        "zpos"          "5"
    }
}

"BlueTimer"
{
    "ControlName"        "CTFHudTimeStatus"
    "fieldName"          "BlueTimer"
    "xpos"                "0"
    "ypos"                "0"
    "zpos"                "2"
    "wide"                "100"
    "tall"                "50"
    "visible"             "1"
    "enabled"             "1"

    if_match
    {
        "xpos"           "38"
    }

    "TimePanelValue"
    {
        "ControlName"    "CExLabel"
        "fieldName"      "TimePanelValue"
        "font"            "surface14"
        "fgcolor"         "255 255 255 255"
        "xpos"            "0"
        "ypos"            "0"
        "zpos"            "3"
        "wide"            "100"
        "tall"            "13"
        "visible"         "1"
        "enabled"         "1"
        "textAlignment"   "center"
        "labelText"       "0:00"

        if_match
        {
            "font"        "surface12"
            "ypos"        "10"
        }
    }
}
```

```
"TimePanelBG"
{
    "ControlName"        "ScalableImagePanel"
    "fieldName"          "TimePanelBG"
    "xpos"                "31"
    "ypos"                "0"
    "zpos"                "2"
    "wide"                "38"
    "tall"                "13"
    "visible"             "1"
    "enabled"             "1"
    "image"               "../hud/objectives_t"
    "scaleImage"          "1"
    "src_corner_height"   "3"
    "src_corner_width"    "3"
    "draw_corner_width"   "0"
    "draw_corner_height"  "0"

    if_match
    {
        "xpos"            "9999"
    }
}
```

Last thing we need to do is opening HudObjectiveTimePanel.res and after every code we are going to add a “if_match” section with “xpos” “9999” in it, that way we can remove every background and overtime/waiting for players etc. labels that otherwise would get in the way.

Do that for every code you find inside HudObjectiveTimePanel.res BUT "ServerTimeLimitLabel".

10. Tournament Hud

Is located on resource>ui>HudTournament.res

I have my own way of doing this which might not be the best or easiest but is the best way I found that works with every hud out there.

First of all, let's open the file and edit the HudTournament positions and wide if needed.

This is what you might have

```
HudTournament
{
  "ControlName"      "EditablePanel"
  "fieldName"        "HudTournament"

  "xpos"             "c-125"
  "ypos"             "0"
  "wide"             "250"
  "tall"             "480"
```

This is what we need to get

```
HudTournament
{
  "ControlName"      "EditablePanel"
  "fieldName"        "HudTournament"

  "xpos"             "0"
  "ypos"             "0"
  "wide"             "f0"
  "tall"             "480"
```

So edit the values to match the right screen and then let's scroll down until we find "HudTournamentBG".

(YOU DON'T NEED TO DO THAT IF THE VALUES WERE ALREADY RIGHT!)

Now starting from "HudTournamentBG" we need to adjust the "xpos" of every code until "CountdownBG" to make our panel centered again.

Before we changed the values from "c-125" to "0" now we need to do the opposite, if "HudTournamentBG" "xpos" is "0" we need to change it to "c-125", if it is 50 it will be c-75 for example.

Here are some screens of examples:

```
"HudTournamentBG"
{
  "ControlName"      "ScalableImagePanel"
  "fieldName"        "HudTournamentBG"
  "xpos"             "0"
  "ypos"             "0"
```

```
"HudTournamentBG"
{
  "ControlName"      "ScalableImagePanel"
  "fieldName"        "HudTournamentBG"
  "xpos"             "c-125"
  "ypos"             "0"
```

```
"HudTournamentBG"
{
  "ControlName"      "ScalableImagePanel"
  "fieldName"        "HudTournamentBG"
  "xpos"             "56"
  "ypos"             "0"
```

```
"HudTournamentBG"
{
  "ControlName"      "ScalableImagePanel"
  "fieldName"        "HudTournamentBG"
  "xpos"             "c-69"
  "ypos"             "0"
```

```
"HudTournamentBG"
{
  "ControlName"      "ScalableImagePanel"
  "fieldName"        "HudTournamentBG"
  "xpos"             "150"
  "ypos"             "0"
```

```
"HudTournamentBG"
{
  "ControlName"      "ScalableImagePanel"
  "fieldName"        "HudTournamentBG"
  "xpos"             "c25"
  "ypos"             "0"
```


Now that the harder part is done, what we need to do now is add the players panel “base_y” plus the new “teams_player_delta_x_comp” “45”

First of all, let’s add the “teams_player_delta_x_comp” as is shown in the screen. In the screen the value is “45” but you should set to be the same as “team2_player_delta_x”

```
"team2_player_base_y"      "40"
"team2_player_delta_x"     "45"
"team2_player_delta_y"     "18"
"teams_player_delta_x_comp" "45"

if_competitive
{
    "team1_player_base_y"   "86"
}
if_readymode
{
    "team1_player_base_y"   "86"
}

"playerpanels_kv"
{
    "visible"               "0"
```

Then add the “if_competitive” and “if_readymode” section the same way as the screen, you can change it to whatever value you like, 0 will be right at the top of the screen while 70 for example is going to be right after the killfeed.

Last thing before we are done is adding the “if_competitive” and “if_readymode” to the codes we edited before.

So once again starting from "HudTournamentBG" at the end of every code we are going to add these sections with inside “visible” “0” so all those panels are not going to show when we join a casual or competitive server, they will only be there in server with the tournament mode enabled.

This is how it needs to be:

```
"enabled"      "1"
"image"        "../HUD/tournamentBG.png"
"proportionaltoparent" "1"

"src_corner_height" "23"
"src_corner_width"  "23"

"draw_corner_width" "8"
"draw_corner_height" "8"

if_mvmm
{
    "visible"      "0"
}
if_competitive
{
    "visible"      "0"
}
if_readymode
{
    "visible"      "0"
}

}
"TournamentLabel"
{
    "ControlName"      "Label"
    "fieldName"        "TournamentLabel"
    "font"              "HudFontSmall"
```

Now, do that until you reach “CountdownBG”, cause here we are still going to add the “if_competitive” and “if_readymode” but instead of using “visible” “0” we are going to use “wide” “0”

Next we have "CountdownLabel" and "CountdownLabelShadow"
Here instead of “visible” “0” we are going to use

“xpos” “c-16” for "CountdownLabel" and
“xpos” “c-15” for "CountdownLabelShadow"

If you still have doubts open the “Example of What an Updated HudTournament.res Files Should Look Like” inside the “Update Files” folder and take a look!

11. WINPANEL

```
"BlueScoreBG"
{
  "ControlName"    "EditablePanel"
  "fieldName"      "BlueScoreBG"
  "wide"           "0"
  "tall"           "0"
  "visible"        "0"
  "enabled"        "0"
}
"RedScoreBG"
{
  "ControlName"    "EditablePanel"
  "fieldName"      "RedScoreBG"
  "wide"           "0"
  "tall"           "0"
  "visible"        "0"
  "enabled"        "0"
}
"BlueScoreBG2"
{
  "ControlName"    "EditablePanel"
  "fieldName"      "BlueScoreBG2"
  "xpos"           "50"
  "ypos"           "15"
  "wide"           "135"
  "tall"           "40"
  "autoResize"     "0"
  "pinCorner"      "0"
  "visible"        "1"
  "enabled"        "1"
  "border"         "TFFatLineBorderBlueBGMoreOpaque"
}
"RedScoreBG2"
{
  "ControlName"    "EditablePanel"
  "fieldName"      "RedScoreBG2"
  "xpos"           "197"
  "ypos"           "15"
  "wide"           "135"
  "tall"           "40"
  "autoResize"     "0"
  "pinCorner"      "0"
  "visible"        "1"
  "enabled"        "1"
  "border"         "TFFatLineBorderRedBGMoreOpaque"
}
```

We can find it here: [resource/ui/Winpanel.res](#)

Right at the start of the file you might have two codes "BlueScoreBG" "RedScoreBG".

What we need to do first is rename both to "BlueScoreBG2" and "RedScoreBG2"

If you didn't have these 2 codes is fine, you can just do what it follows!

Then add two new codes called "BlueScoreBG" "RedScoreBG" and let's make them invisibles by setting the "wide" "tall" "visible" and "enabled" values to "0"

It should be exactly like the screen on the left.

Last thing we need to do is add the new border code.

Scroll down until you find the "WinningTeamLabel" and just above that we will add the new border.

It will be called "WinPanelBGBorder" and again with the "wide" "tall" "visible" "enabled" values set to "0".

```
}
"WinPanelBGBorder"
{
  "ControlName"    "EditablePanel"
  "fieldName"      "WinPanelBGBorder"
  "wide"           "0"
  "tall"           "0"
  "visible"        "0"
  "enabled"        "0"
}
"WinningTeamLabel"
{
```

If you have any doubts take a look at the "Example of What an Updated WinPanel.res Should Look Like" file inside the "Update Files" folder!

12. FIXING THE SCOREBOARD

```
"scores"
{
    "ControlName"      "CTFClientScore"
    "fieldName"        "scoreinfo"
    "xpos"              "c-300"
    "ypos"              "0"
    "wide"              "600"
    "tall"              "516"
    "zpos"              "20000"
    "autoResize"        "0"
    "pinCorner"         "0"
    "visible"           "1"
    "enabled"           "1"
    "tabPosition"       "0"
    "medal_width"       "12"
    "avatar_width"      "55"
    "spacer"            "2"
    "name_width"        "85"
    "nemesis_width"     "15"
    "class_width"       "15"
    "score_width"       "20"
    "ping_width"        "21"
    "killstreak_width"  "12"
    "killstreak_image_width" "12"
}
```

First thing we need to fix the score panel (the part of the scoreboard that displays the avatar – name – score – class – ping etc... of every player)

To do that we need to update the “scores” sections values!

Basically we can replace everything after “tabposition” “0” with the values shown in the screenshot.

These values are the most generic values I could find and they work perfectly with most of the huds.

But there might be cases where the ping or any other value is getting cut off, so all you need to do is increase that value until you can see it entirely!

```
"Kills" → "Kills2"
{
    "ControlName"      "CExLabel"
    "fieldName"        "Kills" → "Kills2"
    "font"              "surface32"
    "fgcolor"           "255 255 255 255"
    "labelText"         "%kills%"
    "textAlignment"     "east"
    "xpos"              "123"
    "ypos"              "9"
    "zpos"              "3"
    "wide"              "50"
    "tall"              "31"
    "autoResize"        "0"
    "pinCorner"         "0"
    "visible"           "1"
    "enabled"           "1"
}
```

Once we fixed the scores we can also make the Kills – Deaths – Assists etc... labels not turn green and stay white when active.

A lot of people like them as they are other don't so is up to you to do this step or not.

Anyway if we want these labels to not change color all we need to do is change their name.

For example, “Kills” will become “Kills2” as shown in the screen.

Do the same with all the other numbers like deaths assist etc...

Leave the “KillsLabel” “DeathsLabel” “AssistLabel” Intact, we only need to change the codes without “Label” in their names.

13. FIXING THE VACCINATOR HUD

Let's start from opening the "Additions to HudMedicCharge" file you can find inside the "Update Files" folder

This file contains everything you need to add in order to fix the Vaccinator Hud so:

- Select everything and copy it
- Then let's open the HudMedicCharge.res of the hud we are updating
- And now we can just paste everything we copied before at the end of the file!

First of all, we need to get the bar charges in position.

They might be out of the screen (invisibles) depending on what are the general positioning values of the HudMedicCharge.res file we are updating, so if that is the case we can set the "xpos" and "ypos" of "ChargeMeter1" both to 0 and that should make at least the first bar visible.

Now we need to slowly work with the "xpos" "ypos" "wide" and "tall" of "ChargeMeter1" "ChargeMeter2" "ChargeMeter3" "ChargeMeter4" to get them all in the desired position.

Last thing is the resistance icon,



to move it we are going to use the "xpos" and "ypos" of "ResistIconAnchor",

If we want to resize it, we need to change the "wide" and "tall" values of "ResistIcon".

Keep in mind they need to be the same or it will look really bad.

THOSE STEPS ARE NOT MANDATORY, BUT YOUR HUD WILL BE A HAPPY LITTLE THREE IF YOU DO THEM! 😊

1. CHARINFOPANEL.RES FIX

Once again this fix is only needed if the files are present in the hud.

All we need to do with this file is:

- Scroll down until we find "Sheet"
- Add this new line "yoffset" "14" (needs to be added right after "transition_time")
- Change the value to whatever is needed to show the tabs (Loadout / Stats in this case) in the right way

The exact same thing needs to be done with resource/ui/econ/store/v2/StorePanel.res

2. MISSING STATUS ICONS

To be clear we are talking about these icons:



All these icons can be found and edited from the HudPlayerHealth.res

What I like to do when I update these icons is create an anchor so you can move all the icons at once.

To do that once again open the Update Files Folder and the Additions to HudPlayerHealth file.

Now copy everything inside the file and replace the old icons in our HudPlayerHealth.res (the first one usually is "PlayerStatusBleedImage" so select and replace everything from there to the end)

Once we do that our hud should have all the icons recently added but we still need to place them.

To place them, we only need to work with the "xpos" and "ypos" of "PlayerStatusAnchor" until we get them in position!

Note if after the process the icons are still not visible check if the "xpos" and "ypos" of "PlayerStatusHealthImage" are offscreen, if they are then set both to 0 and do the same for the "wide" and "tall"

3. FIXING SPECTATOR ITEM PANELS

We are talking about this panels:



With the last couple of major updates those panels got broken in many huds so we are going to replace them with the default panels.

First of all, delete HudInspectPanel.res if present in the hud.

Now we can copy the new panels from the “Additions to Spectator and SpectatorTournament” file inside the Update Files folder.

All we need to do is copy the first ItemPanel (Spectator.res) and replace it in our Spectator.res.

Then do the same thing with the SpectatorTournament.res.

4. FIXING FLOATING HEALTH



Fixing the floating health is pretty easy, let's start with adding the default file.

It's called HealthIconPanel.res and we can find it in the Update Files Folder.

After we added the file to our hud we can join an offline server and spawn a bot to edit the floating number.

We got to work with both "wide" and "tall" of "HealthIcon" until the number is completely visible and centered with the body.

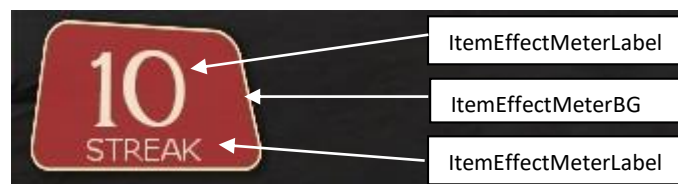
Remember the floating health is only visible if

`tf_hud_target_id_disable_floating_health` is set to 0.

5. ADD THE KILLSTREAK ICON

Once again let's start by adding the file that controls the killstreak to our hud.

It's called HudItemEffectMeter_KillStreak and we can find it in the Update Files Folder.



We can edit it as we want using the values of "ItemEffectMeterBG" "ItemEffectMeterLabel" and "ItemEffectMeterCount"

And move the whole thing by using the values of "HudItemEffectMeter" at the start of the file.

6. ADD THE 3D MODEL

Last thing, we can add a 3d model if needed. Open The Additions to HudPlayerClass file and copy & paste everything inside our HudPlayerClass.res (right after "classmodelpanelBG" its fine).

Now we have our 3d model, if we want to edit it we need to work with the various angles, origin, frame, fov values. Is possible to rotate it vertically, horizontally, make it bigger etc... just experiment with the values a bit!

7. ADD THE NEW WEAPONS METERS AND COUNTERS

With the last updates new weapons got added to the game and so new meters and counters, for example the organs counter for the vita saw and the Phlog, Thermal Thruster meters.

Again I have my own method to do this, basically I make the new meters work with the #base system so I have to do pretty much nothing.

But first let's add the files from the Update Files folder and let's grab:

- HudItemEffectMeter_Organs
- HudItemEffectMeter_Pyro
- HudRocketPack

Each one of this files is based on the original HudItemEffectMeter, which basically means that the meter already looks the same as the others and all we need to do is reposition it! And we can easily do that by using the xpos and ypos that we can find inside each file!

HudItemEffectMeter_Organs is just a counter and that's why it doesn't have the xpos and ypos cause usually doesn't need to get repositioned!

Depending on how old the hud is it might be missing more than the meters listed above, if so we can take the missing meters from the Default Hud Files/Resource/UI folder. After that I suggest coping and pasting the content of HudItemEffectMeter_Pyro inside the file we are adding, and then finally place it in the right position.

Same steps for any other meter.

You have done it NICEEEEEEEEEEEEE!!!!!!



Shoutouts to:

Rays

Omni

Doodle

Flame

Wiet

Huds.tf

TF.TV people

for the updating hud posts and hud editing guides!