

Créer son propre skin sur *Celeste*

Cette documentation concerne la création et comment organiser votre skin sur Celeste. Si vous ne savez pas quoi faire ou par où commencer, alors nous essayerons de répondre à votre/vos question(s) ici.

SOMMAIRE

Composition du dossier du skin -----	1
Everest.yaml-----	3
SkinModHelperConfig.yaml-----	3
Dossier « Dialog » -----	4
Dossier « Graphics » -----	5
Dossier « Atlases » -----	9
Dossier « Portraits » -----	9
Dossier « Gameplay » -----	11
Dossier « characters » -----	12
Configuration skin -----	14
Colorgrading -----	16
Position Frames Animations -----	19
Paramètres de Skin Mod Helper Plus -----	20
« Advanced Options » -----	20
Celeste NET -----	21
Recommandations Logiciels -----	21

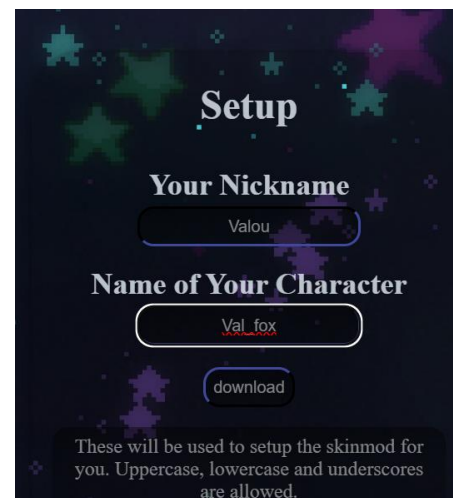
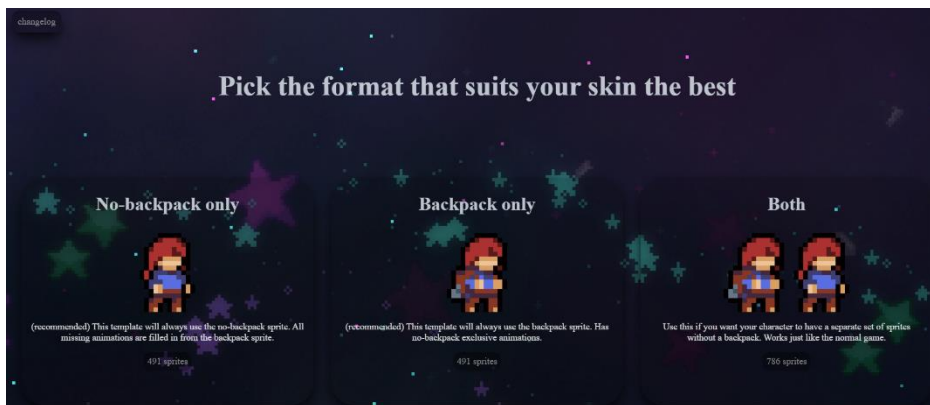
COMPOSITION DU DOSSIER DU SKIN :

Dialog	27/08/2025 01:15	Dossier de fichiers	
Graphics	27/08/2025 01:23	Dossier de fichiers	
everest.yaml	05/08/2025 23:05	Fichier source Yaml	1 Ko
SkinModHelperConfig.yaml	27/08/2025 01:15	Fichier source Yaml	1 Ko

Votre skin soit être un fichier zip contenant ces dossiers et fichiers :

Afin d'obtenir ce fichier ZIP bien organisé, je vous conseille d'aller voir ce [GitHub](https://kuksattu.github.io/celeste/skinmod-template) (créé par Kuksa) qui vous préparera un fichier ZIP pour vous avec tout ce dont vous aurez besoin pour votre skin (certaines animations peuvent manquer si vous faites un skin avec ou sans sac-à-dos, faites donc attention d'avoir tout ce qu'il vous faut !) :

<https://kuksattu.github.io/celeste/skinmod-template>



Sur ce [GitHub](https://kuksattu.github.io/celeste/skinmod-template) se trouve un choix parmi 3 types de skins :

- « No-backpack only » : qui vous fera un ZIP avec le sprite de Madeline sans backpack.
- « Backpack only » : qui vous fera un ZIP des sprites avec backpack.
- « Both » : qui vous fera un ZIP des sprites avec et sans backpack (qui seront séparés).

Votre « Nickname » et le nom de votre personnage feront partie de vos dossiers pour votre skin, vous les verrez souvent entre certains dossiers (dans le cas ci-dessous Valou/Val_fox/). Pour mieux vous expliquer « Valou/Val_fox / » se traduit par « YourName/YourCharacter/ » et sert à éviter les conflits avec d'autres sprites du même nom mais également cela correspond au système de nommage standard utilisé par la communauté Celeste pour l'architecture des mods.

Val_fox_Skinmod_Valou > Graphics > Valou > Val_fox

Tout d'abord, nous allons regarder les fichiers « *everest.yaml* » et « *SkinModHelperConfig.yaml* » :

EVEREST.YAML :

```
- Name: Val_Skinmod_Valou
  Version: 1.0.0
  Dependencies:
    - Name: Everest
      Version: 1.4465.0
    - Name: SkinModHelperPlus
      Version: 0.10.4
```

Juste un fichier contenant les dépendances de votre skin.

SKINMODHELPERCONFIG.YAML :

```
- SkinName: "Valou_Val_fox"
  Player_List: true
  Silhouette_List: false
  Character_ID: "Valou_Val_fox"
  OtherSprite_Path: "Valou/Val_fox"
  Mod: "SkinModHelper_Grouping_Val_fox"

- SkinName: "Valou_Val_fox_playback"
  Player_List: false
  Silhouette_List: true
  Character_ID: "Valou_Val_fox_playback"
  OtherSprite_Path: "Valou/Val_fox_playback"
```

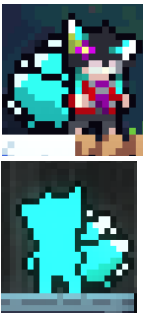
« SkinName » = le nom de votre skin

« Player_List » = activer si le skin est dans la liste du skin joueur/euse

« Silhouette_List » = activer si le skin est dans la liste du skin silhouette, aussi appelé « playback » (à activer **uniquement** si vous comptez en faire un)

« Character_ID » = nom utilisé dans vos fichiers .xml

« OtherSprite_Path » = chemin des dossiers menant au second fichier « Sprites.xml » pour les sprites supplémentaires.



DOSSIER « DIALOG » :

inmod_Valou > Dialog			Rechercher...
Nom	Modifié le	Type	
English.txt	27/08/2025 01:15	Document texte	

English.txt - Bloc-notes

Fichier Edition Format Affichage Aide

```
SkinModHelper_Player__Valou_Val_fox= Val fox (Valou)
SkinModHelper_Player__Valou_Val_fox_playback= Silhouette Val fox (Valou)
SkinModHelper_Player__Valou_Val_fox__Description= Creator: Valou
SkinModHelper_Grouping__Val_fox= Val
```

Dans ce fichier .txt, vous pourrez nommer les noms de vos types de skins (normal ou playback) :

- « ***SkinModHelper_Player__Valou_Val_fox*** » = nom du skin joueur/euse
- « ***SkinModHelper_Player__Valou_Val_fox_playback*** » = nom du skin silhouette
- « ***SkinModHelper_Player__Valou_Val_fox__Description*** » = Description du skin (nom du créateur/rice)
- « ***SkinModHelper_Grouping__Valou_Val_fox*** » = nom du mod de votre skin

DOSSIER « GRAPHICS » :

Fichiers .xml :

Nom	Modifié le	Type
Atlases	23/08/2025 02:00	Dossier de fichiers
Valou	27/08/2025 02:14	Dossier de fichiers
Portraits.xml	27/08/2025 20:30	Microsoft Edge HT
Sprites.xml	27/08/2025 01:14	Microsoft Edge HT

Nom	Modifié le	Type
Sprites.xml	27/08/2025 03:37	Microsoft Edge H

Vous avez deux types de fichiers .xml, « Portraits » (contenant les informations sur les visages et boîtes de texte des différents personnages) et « Sprites » (contenant les informations sur chaque sprite de votre skin), nous verrons ces informations un peu plus en détail plus bas dans le document.

Vous avez certainement remarqué qu'il y a deux fichiers « Sprites », le premier se situant dans le dossier « Graphics » contient les informations des animations du jeu de base et des mods ainsi que sur le sprite « playback ».

Le second regroupe les informations sur les sprites utilisant des objets, par exemple, quand Madeline utilise la paire de jumelles (animation « Lookout »).

Attention : si vous mettez des animations de sprites utilisant un objet dans le premier fichier Sprites.xml, ces derniers vont rester actifs même si vous changez de skin, donc gardez bien à l'esprit qu'il faut séparer les sprites avec et sans objet.

L'intérieur d'un fichier .xml ressemble à ceci :

```
<Valou_Val_fox path="characters/Valou/Val_fox/" start="idle">
  <Origin x="16" y="32" />

  <Anim id="roll" path="roll" frames="0-12" delay="0.05"/>
  <Anim id="rollGetUp" path="roll" frames="12-16" delay=".1" goto="downed"/>
  <Loop id="downed" path="rollSit" delay="0.8" frames="0"/>

  <Loop id="idle_carry" path="idle_carry" delay="0.1" />
  <Anim id="runSlow_carry" path="run_carry" delay="0.07" />
  <Loop id="jumpSlow_carry" path="jump_carry" delay="0.1" frames="0,1" />
  <Anim id="fallSlow_carry" path="jump_carry" delay="0.1" frames="2,3" />
  <Anim id="pickUp" path="pickup" delay="0.06" />
  <Anim id="throw" path="throw" delay="0.06" goto="idle" />

  <Anim id="idle" path="idle" delay="0.1" goto="idle"/>

  <Anim id="idleA" path="idleA" delay="0.12" goto="idle"/>
  <Anim id="idleB" path="idleB" delay="0.16" goto="idle"/>
  <Anim id="idleC" path="idleC" delay="0.11" goto="idle"/>

  <Metadata>
    <Frames path="idle" hair="x"/>
    <Frames path="idleA" hair="x"/>
    <Frames path="idleB" hair="x"/>
    <Frames path="idleC" hair="x"/>

    <Frames path="lookUp" hair="x"/>
    <Frames path="edge" hair="x"/>
    <Frames path="edge_back" hair="x"/>
    <Frames path="walk" hair="x"/>
    <Frames path="push" hair="x"/>
    <Frames path="runSlow" hair="x"/>
    <Frames path="runFast" hair="x"/>
    <Frames path="run_wind" hair="x"/>
    <Frames path="runStumble" hair="x"/>
    <Frames path="dash" hair="x"/>
    <Frames path="dreamDash" hair="x"/>
    <Frames path="jumpSlow" hair="x"/>
    <Frames path="jumpFast" hair="x"/>
    <Frames path="tired" hair="x"/>
    <Frames path="climb" hair="x"/>
    <Frames path="duck" hair="x"/>
    <Frames path="fallPose" hair="x"/>
    <Frames path="hug" hair="x"/>
    <Frames path="sleep" hair="x"/>
    <Frames path="flip" hair="x"/>
    <Frames path="dangling" hair="x"/>
    <Frames path="shaking" hair="x"/>
  </Metadata>

  <Valou_Val_fox path="characters/Valou/Val_fox/" start="idle">
```

Vous avez le chemin menant à vos sprites commençant directement depuis le dossier « Gameplay » (n'oubliez pas le « / » à la fin de votre chemin) et start avec sur quelle animation votre skin va commencer.

```
<Anim id="roll" path="roll" frames="0-12" delay="0.05"/>
```

Chaque ligne comme celle-ci possède une ID qui est un nom d'une animation référencée aussi dans le code même du jeu, un chemin (path) qui montre le chemin de dossiers menant jusqu'à l'animation, les frames qui sont le nombre de sprites pour une animation (exemple pour l'animation « roll » ci-dessus, les frames vont du sprite « roll0 » au sprite « roll12 »).

Mais ces frames peuvent être changées pour vous donner d'autres possibilités pour vos animations, pas besoin de faire défiler votre animation en allant juste de 0 à 12, vous pouvez faire ce que vous voulez :

```
<Anim id="bigFallRecover" path="bigFall" frames="5*5,6*4,7*3,8,8,9,9,10,10,10" delay="0.08" goto="swimIdle"/>
```

Par exemple, dans cette animation avec des frames comme :

« 5*5, 6*4, 7*3, 8, 8, 9, 9, 10, 10, 10 »

Nous pouvons traduire cela par :

« 5, 5, 5, 5, 5, 6, 6, 6, 6, 7, 7, 7, 8, 8, 9, 9, 10, 10, 10 »

Voulant dire que sur le total des 19 frames :

- Les 5 premières sont le sprite « 5 »
- Les 4 suivantes le sprite « 6 »
- Les 3 suivantes le sprite « 7 »
- Les 2 suivantes le sprite « 8 »
- Les 2 autres suivantes le sprite « 9 »
- Les 3 dernières sont le sprite « 10 »

Le « delay » est le temps (en seconde) entre chaque sprite de l'animation (à noter que 0.1 delay = 10 FPS) :

Exemple avec l'animation « idle » ayant un « delay » de 0.1 :

```
<Anim id="idle" path="idle" delay="0.1" goto="idle"/>
```

0 delay



0,1 delay



0,2 delay



0,3 delay



0,4 delay



0,5 delay



0,6 delay



```
<Frames path="idle" hair="-2,4|-2,4|-2,4|-2,4|-2,4|-2,4|-2,4|-2,4|-2,4"/>
```

Dans la partie « metadata » se trouve également un chemin de dossiers mais également une variable « hair » dans laquelle se trouve la position x et y des coordonnées des pixels des cheveux sur chaque frame de l'animation. Ces coordonnées sont placées par rapport aux coordonnées d'origine du sprite (appelées « Origin » dans le fichier .xml), ces coordonnées sont séparées par un « | », ci-dessus vous voyez :

```
« -2,4|-2,4|-2,4|-2,4|-2,4|-2,4|-2,4|-2,4|-2,4 »
```

Vous pouvez le visualiser ainsi :

```
« x,y|x,y|x,y|x,y|x,y|x,y|x,y|x,y »
```

Les premières coordonnées sont pour la frame 0, dans ce cas précis le sprite « idle0 » et les prochaines seront pour « idle1 », « idle2 » ... jusqu'à « idle8 ». Comme ceci :

« x,y | x,y | x,y | x,y | x,y | x,y | x,y | x,y | x,y »

« *frame 0* | *frame 1* | *frame 2* | *frame 3* | *frame 4* | *frame 5* | *frame 6* | *frame 7* | *frame 8* »

<Frames path="idle" hair="x"/>

Si vous mettez un « x » à la place des coordonnées, les pixels seront retirés.

goto="runFast"

Certaines lignes possèdent « goto » qui représente l'animation qui va se jouer à la suite de l'animation en question.

DOSSIER « ATLASES » :

Skinmod_Valou > Graphics > Atlases >			
Nom	Modifié le	Type	
Gameplay	27/08/2025 02:58	Dossier de fichiers	
Portraits	23/08/2025 02:00	Dossier de fichiers	

Dans le dossier « Atlases » se trouve deux dossiers, le dossier « Gameplay » où vont se trouver tous les sprites de votre skin, et le dossier « Portraits » où vont se trouver les images des portraits de votre skin (vous n'êtes pas obligé(e) d'avoir des portraits pour votre skin).
Passons rapidement sur le dossier « Portraits » :

DOSSIER « PORTRAITS » :

Skinmod_Valou > Graphics > Atlases > Portraits >			
Nom	Modifié le	Type	
textbox	23/08/2025 02:00	Dossier de fichiers	
Valou	27/08/2025 20:28	Dossier de fichiers	

Il contient un dossier « textbox » où vous aurez la boîte de dialogue de votre personnage.

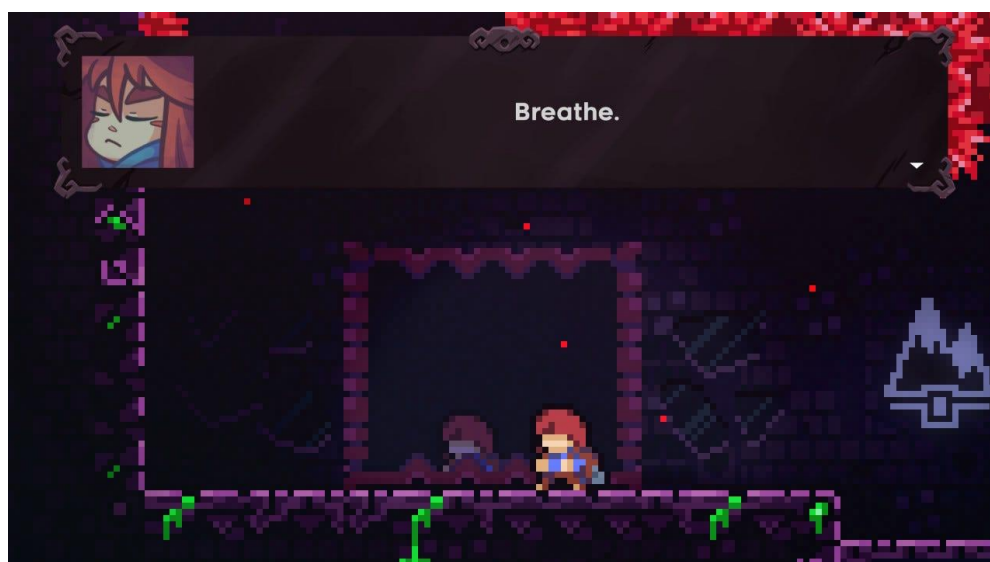
Et le second dossier correspond à celui contenant les expressions du visage de votre personnage :

Skinmod_Valou > Graphics > Atlases > Portraits > Valou > Val_fox >			
Nom	Modifié le	Type	
madeline	23/08/2025 02:00	Dossier de fichiers	
mirror	23/08/2025 02:00	Dossier de fichiers	
phone	23/08/2025 02:00	Dossier de fichiers	

Le dossier « madeline » pour les expressions courantes de votre skin :



Le dossier « mirror » pour les expressions de votre skin au cours du chapitre 5 :

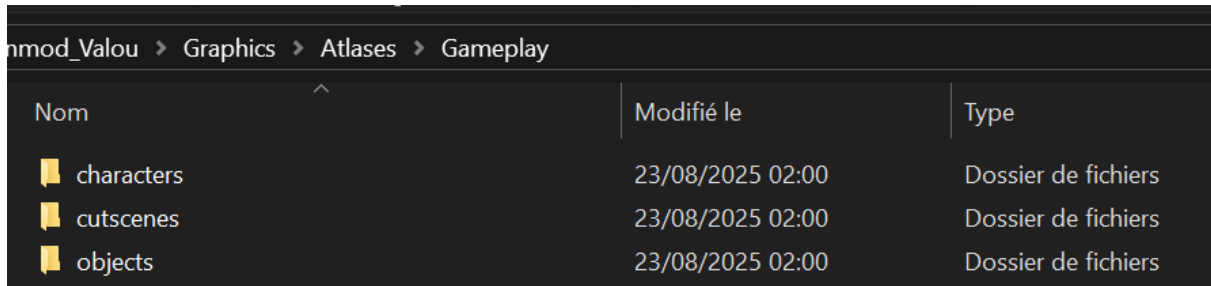


Le dossier « phone » pour les expressions de votre skin à la fin du chapitre 2 :



DOSSIER « GAMEPLAY » :

Maintenant passons au dossier « Gameplay »

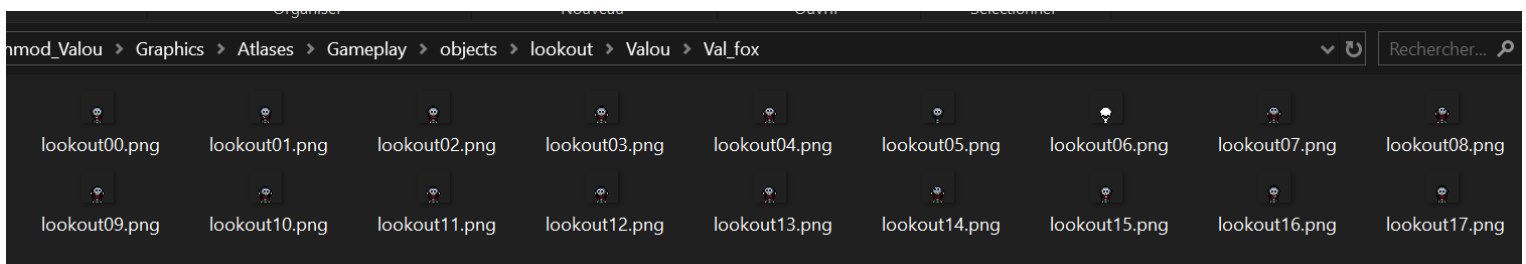


hmod_Valou > Graphics > Atlases > Gameplay

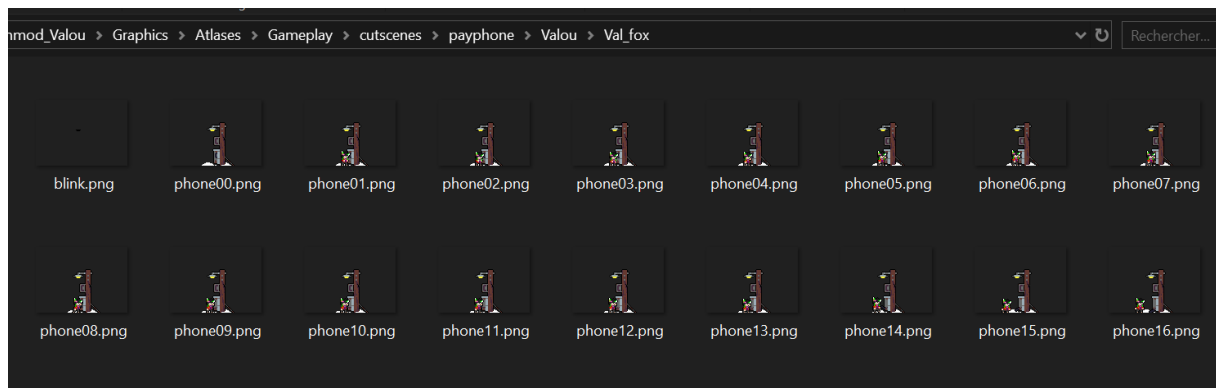
Nom	Modifié le	Type
characters	23/08/2025 02:00	Dossier de fichiers
cutscenes	23/08/2025 02:00	Dossier de fichiers
objects	23/08/2025 02:00	Dossier de fichiers

Il contient un total de 3 dossiers :

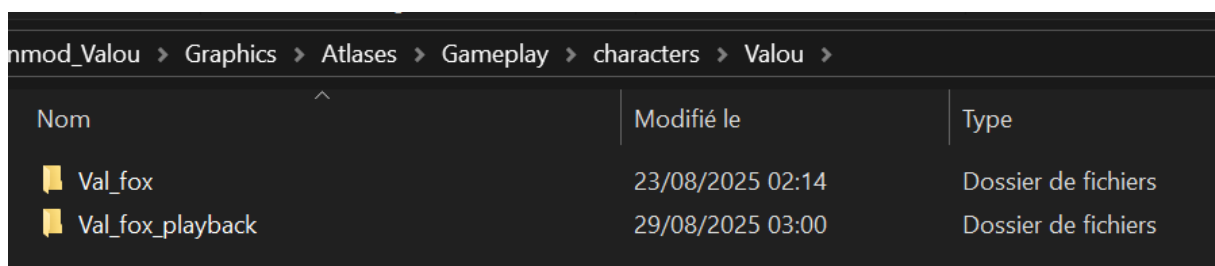
- Un dossier « objects » contenant l'animation des jumelles



- Un dossier « cutscenes » avec l'animation du téléphone du chapitre 2A



- Un dossier « characters » avec toutes les animations du skin et du playback (du jeu de base et des mods)

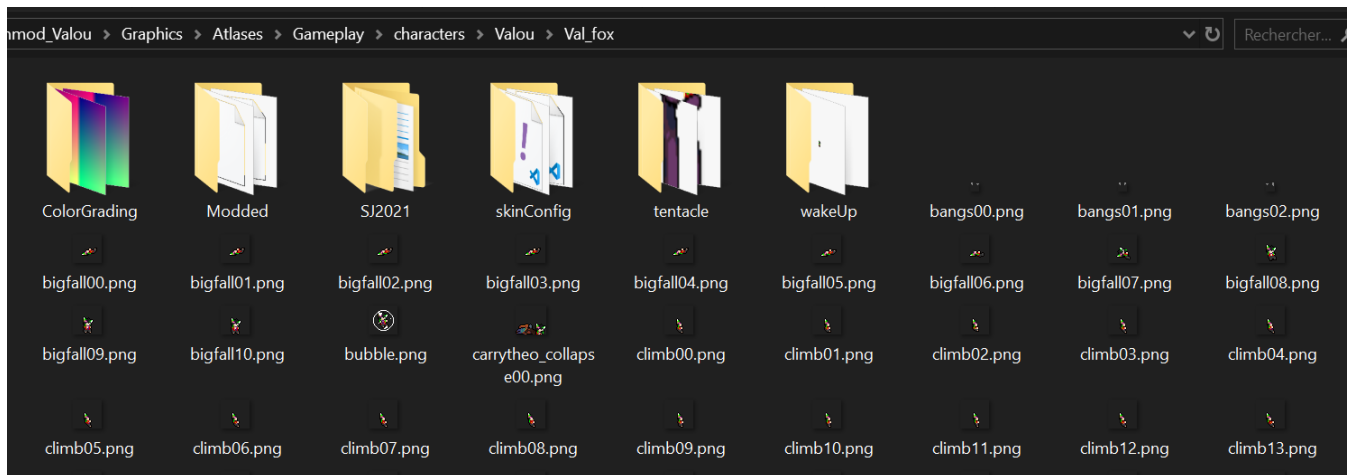


hmod_Valou > Graphics > Atlases > Gameplay > characters > Valou >

Nom	Modifié le	Type
Val_fox	23/08/2025 02:14	Dossier de fichiers
Val_fox_playback	29/08/2025 03:00	Dossier de fichiers

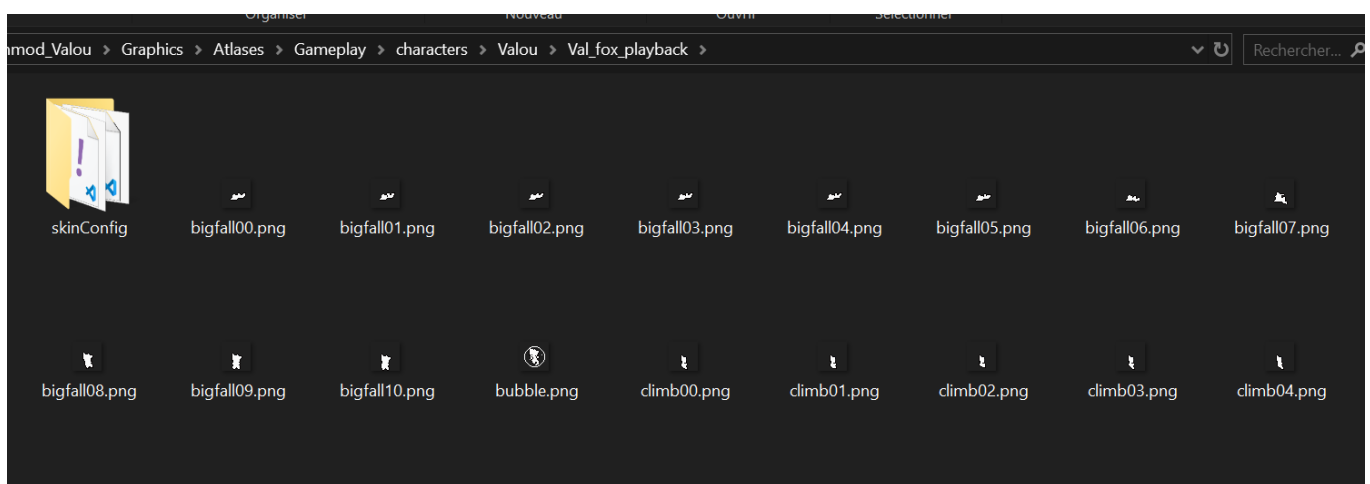
DOSSIER « CHARACTERS » :

Skin classique :

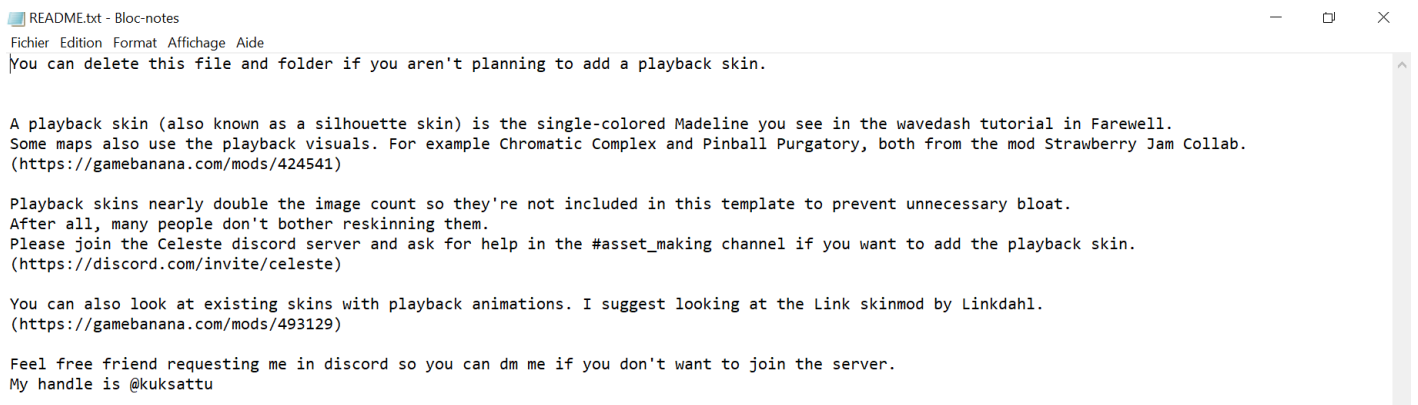


Playback :

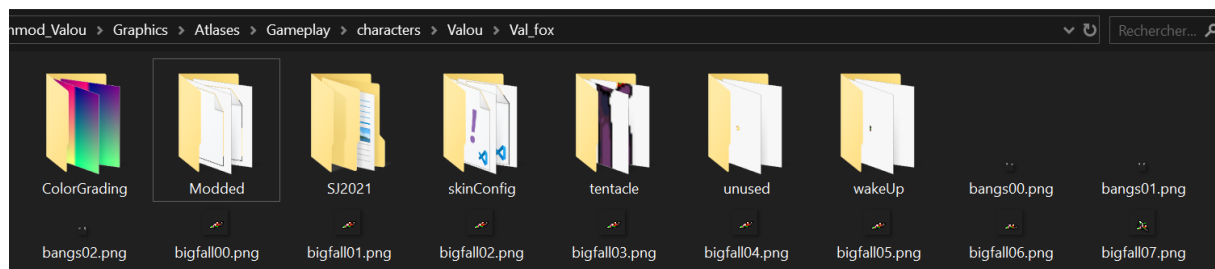
Le screen du dossier que je vais vous montrer est celui que j'ai fait moi-même, il ne sera pas déjà prêt pour vous quand vous aurez votre fichier ZIP entre vos mains.



ATTENTION : Quand vous téléchargerez votre fichier ZIP, vous obtiendrez un dossier avec un fichier « READ.me » vous disant ceci :



Regardons le dossier du skin classique :



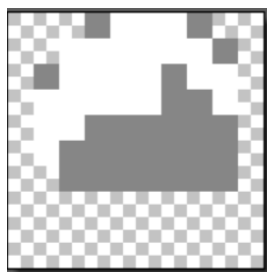
Vous devez avoir les dossiers :

- « **ColorGrading** » (qui va gérer le changement de couleur de votre dash sur votre skin)
- « **Modded** » (celui n'est pas obligatoire mais il contient les sprites de différents helpers/mods)
- « **SJ2021** » (celui n'est pas obligatoire et a été ajouté manuellement par mes soins, il contient des animations du niveau Paint de la Strawberry Jam)
- « **skinConfig** » (qui contient deux fichiers .cs qui gère les paramètres graphiques de votre skin, nous y reviendrons plus tard)
- « **unused** » (qui contient des frames non utilisées)
- Et les dossiers « **tentacle** » et « **wakeUp** » contiennent des sprites du jeu de base pour votre skin (chapitre 6 avec les tentacules de Badeline et le réveil de Madeline du chapitre 2)

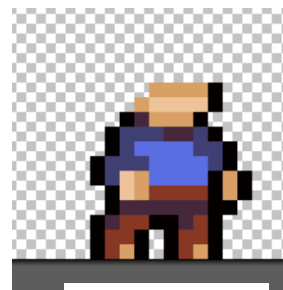
Attention : Il y a des dossiers que vous devrez créer vous-même et qui ne sont pas déjà prêts dans le fichier ZIP, comme par exemple le dossier « ColorGrading ».

Le reste des fichiers sont les sprites du jeu de base pour votre skin, à vous de les modifier également comme bon vous semble.

Avant de se plonger dans les dossiers « skinConfig » et « Colorgrading », j'aimerais vous parler de ce fichier s'appelant « bang ».





Bang



Sprite (idle)

Sur Celeste, les cheveux de Madeline sont séparés de son corps, si vous voulez modifier la base des cheveux, alors modifiez le fichier « bang », ou alors vous pouvez directement mettre les cheveux sur le corps de base du skin (certains skins ont les cheveux directement sur le sprite, laissez-les). **Mais attention, si vous mettez les cheveux directement sur le sprite, ils ne réagiront pas au vent et aux mouvements directionnels.**

CONFIGURATION SKIN :

inmod_Valou > Graphics > Atlases > Gameplay > characters > Valou > Val_fox > skinConfig			
Nom	Modifié le	Type	Taille
 CharacterConfig.yaml	31/07/2025 22:09	Fichier source Yaml	1 Ko
 HairConfig.yaml	22/08/2025 02:53	Fichier source Yaml	1 Ko

Maintenant, le dossier « SkinConfig »

Il contient deux fichiers :

- CharacterConfig

```
SilhouetteMode: false      # If enabled, the player will be tinted to the hair color.
LowStaminaFlashColor: "FF0000" # Specifies the color that the player will flash when their stamina is below 20.
LowStaminaFlashHair: false  # If enabled, low stamina flash will color the hair in addition to the player.
# DeathParticleColor: "FFFFFF" # Uncomment to take effect. Changes the color of the 8 particles that flash when you die. Normally
```

« **SilhouetteMode** » : Le sprite sera teinté de la couleur des cheveux si en « true » (c'est pour cela que les sprites du skin playback sont blancs).

« **LowStaminaFlashColor** » : Spécifie la couleur qui clignotera lorsque l'endurance du joueur/euse sera faible/épuisée.

« **LowStaminaFlashHair** » : Indique une endurance faible qui colorera les cheveux en plus du sprite.

« **DeathParticleColor** » : Spécifie la couleur des particules du skin quand ce dernier meurt.

```
# These are the idle animations when a map has no core mode or the core mod is set to cold.
IdleColdOptions:
- A, 3      # First the identifier, then the weight of the probability
- B, 5      # This corresponds to <Anim id="idleB" ... />
- C, 1      # This corresponds to <Anim id="idleC" ... />
```

« IdleColdOptions » : Représente la probabilité que l'animation « idle » A, B ou C se joue quand il y a le mode « froid » de Core sur la map (chapitre 8) ou quand il n'y a pas de mode :

- 3 chances 9 pour avoir l'animation « idleA »
- 5 chances 9 pour avoir l'animation « idleB »
- 1 chances 9 pour avoir l'animation « idleC »

Mode « froid » du chapitre 8 :




```
# Uncomment these two lines to enable the custom idle animations defined inside the Sprites.xml.
# - D, 1      # You can add more animations like this. This corresponds to <Anim id="idleD" ... />
# - Sit, 1    # The identifier doesn't have to be a single letter. This corresponds to <Anim id="idleSit" .
```

Ces deux lignes sont là si vous souhaitez activer des animations customisées que vous pouvez supprimer (informations sur le Sprites.xml dans le dossier « Graphics ») :

```
<!-- These two animations are custom idle animations. They don't currently do anything. -->
<!-- If you want to make them appear ingame, go to the CharacterConfig.yaml and follow the instructions there. -->
<Anim id="idleD" path="lookUp" delay="0.1" frames="0-4,5*10,5,4,3,2,1" goto="idle"/>
<Anim id="idleSit" path="sitDown" delay="0.1" frames="8-14,15*9999" goto="idle"/>
```

```
# These are the idle animations when the core mode is set to hot.
IdleWarmOptions:
- A, 3
- B, 5
- C, 1
```

« IdleWarmOptions » : Représente la probabilité que l'animation « idle » A, B ou C se joue quand il y a le mode « chaud » de Core sur la map (chapitre 8) :

- 3 chances 9 pour avoir l'animation « idleA »
- 5 chances 9 pour avoir l'animation « idleB »
- 1 chances 9 pour avoir l'animation « idleC »

Mode « chaud » du chapitre 8 :



```
# The chance for an idle animation to play every time the idle animation loop finishes.
# The default value is 0.2=20%. This template makes it a lower 5% because vanilla 20% plays idle animations basically constantly.
IdleAnimationChance: 0.05
```

« IdleAnimationChance » : Représente la probabilité que les animations « idleA », « idleB », « idleC » soient jouées à la fin de l'animation « idle ».

- HairConfig

```
# Please use an online yaml validator if you encounter issues.
HairFlash: true # Enables the white flash that happens when your dash count changes.
HairFloatingDashCount: 2 # The dash count at which the hair starts floating like when you have two dashes. -1 to make the hair never
OutlineColor: "040000" # This specifies the hair outline color, not the sprite outline of the player.

# If your custom bangs or hair texture is larger than 10x10 pixels, it will get offset ingame.
# Use these offsets to move the hair back into the right position.
BangsOffset: 0, 0
HairOffset: 0, 0
```

```
# Here you change the hair color/scale/length
HairAttrWithDashes:
- Dashes: -1 # -1 Means the feather tail.
  Scale: 1, 0.357 # The scale of the hair from start to end
  Length: 7

- Dashes: 0
  Color: "46C3FF"
  Scale: 1, 0.4375
  Length: 4

- Dashes: 1
  Color: "AC3232"
  Scale: 1, 0.4375
  Length: 4

- Dashes: 2
  Color: "FF7BFF"
  Scale: 1, 0.4375
  Length: 5
```

```
- Dashes: 3
  Color: "FF7BFF"
  Scale: 1, 0.4375
  Length: 5
  SegmentAttrs:
  - Segment: 0
    Color: "FF7BFF"
  - Segment: 1
    Color: "B359C4"
  - Segment: 2
    Color: "733688"
  - Segment: 3
    Color: "B359C4"
  - Segment: 4
    Color: "FF7BFF"
```

« **HairFlash** » : Active le flash blanc qui apparaît lorsque votre nombre de dashes change.

« **HairFloatingDashCount** » : Spécifie le nombre de dashes auxquels les cheveux du skin flottent (comme quand Madeline possède deux 2 dashes).

« **OutlineColor** » : Spécifie la couleur du contour des cheveux, et non celle du sprite.

« **BangsOffset** » : Permet de changer la position du sprite « bang » skin (à faire si ce dernier dépasse la taille de 10x10).

« **HairOffset** » : Permet de changer la position des cheveux du skin (à faire si ce dernier dépasse la taille de 10x10).

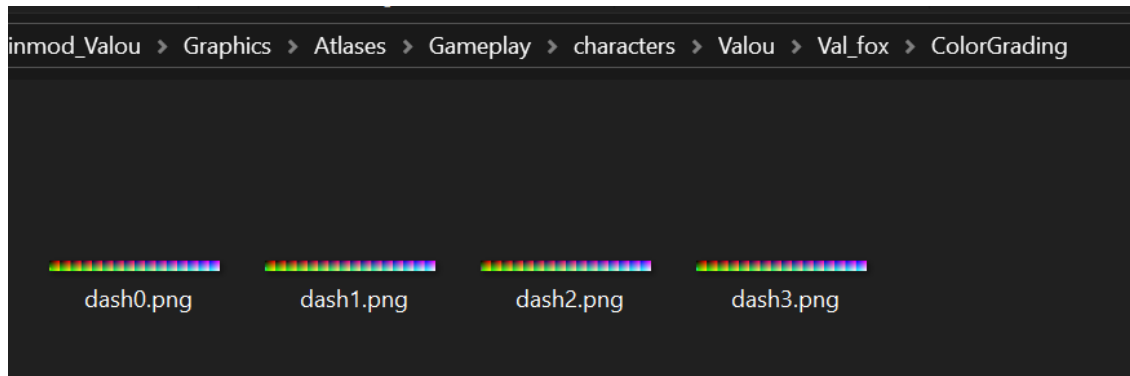
« **Dashes** » : Représente le nombre de dashes du skin (« -1 » pour la plume, « 0 » pour aucun dash etc...)

« **Color** » : Modifie la couleur **des cheveux et du dash (traînée visuelle)** de votre skin en agissant comme un filtre (en code HEX) en fonction de votre nombre de dashes.

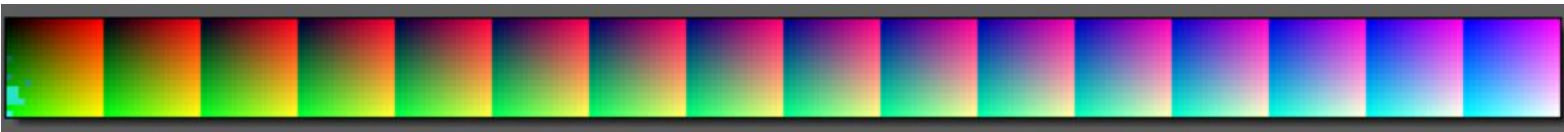
« **Scale** » : Modifie la taille des cheveux de votre skin (du début à la fin) en fonction de votre nombre de dashes.

« **Length** » : Modifie la longueur des cheveux de votre skin en fonction de votre nombre de dashes.

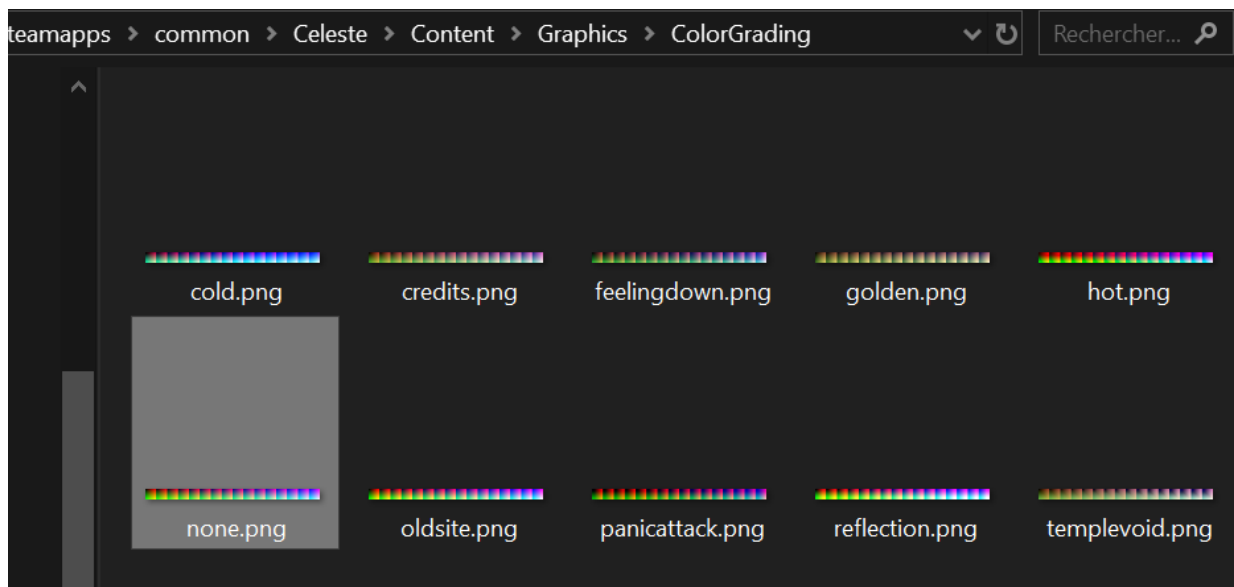
COLORGRADING :



Le colorgrading est une étape très importante dans la confection de votre skin.



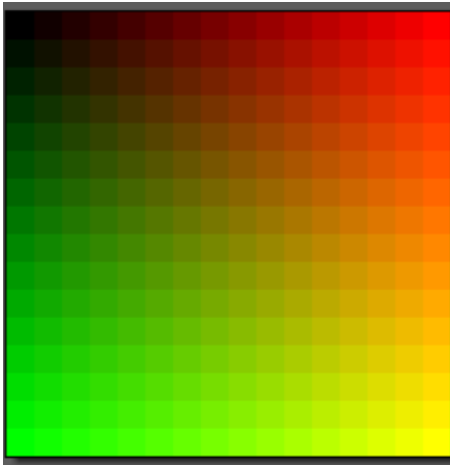
Le colorgrading est une palette de couleurs qui va remplacer les couleurs qui seront sur votre skin par les couleurs modifiées (vous pourrez trouver les images ci-dessus dans les dossier du jeu Celeste) :



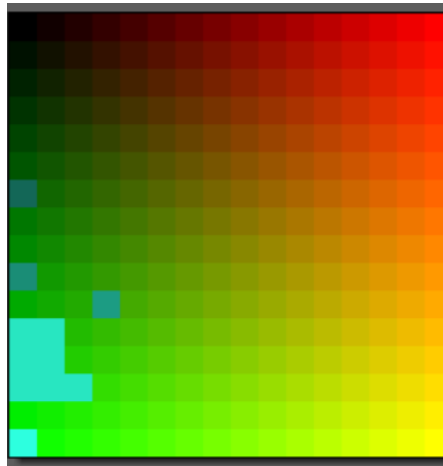
Vous aurez juste à reprendre l'image s'appelant « none », représentant le colorgrading vide, sans modifications et à l'utiliser pour votre skin.

Voici un exemple :

Colorgrading vide, sans modification



Colorgrading avec modification



Sprite dans votre fichier ZIP



Sprite dans le jeu



Le colorgrading n'est qu'une palette sur laquelle vous posez les couleurs que vous voudrez voir à la place d'autres couleurs sur votre skin (et cela pour chaque dash).

Les couleurs de la palette de base (colorgrading vide, sans modification) que vous mettrez sur votre skin vont donc être remplacées en jeu par les nouvelles couleurs que vous aurez placées sur la palette (colorgrading avec modification).

N'oubliez pas d'avoir le bon nom pour chaque fichier de colorgrading ! (« dash1 » pour quand votre skin n'a que 1 seul dash etc... voir le premier screenshot du dossier ci-dessus un peu plus haut).

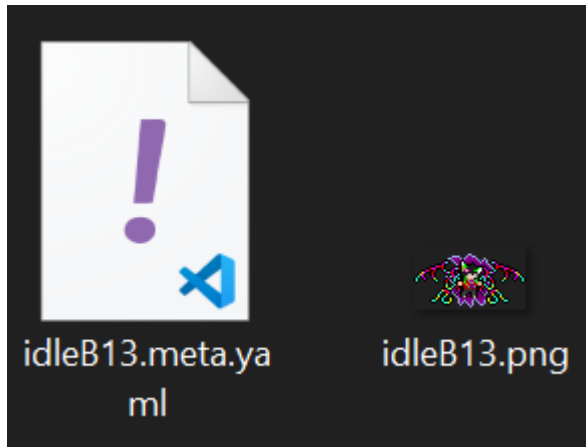
Attention : Certaines animations de helpers/mods et vanilla peuvent ne pas prendre en compte le colorgrading, vous devrez donc appliquer la couleur de votre dash directement sur le sprite.

POSITION FRAMES ANIMATION :

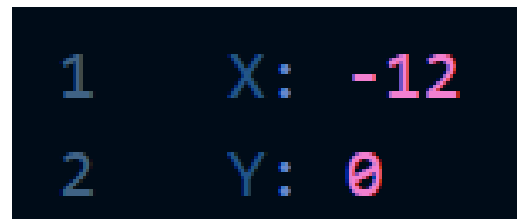
Ceci est un point assez spécifique mais je me devais de l'écrire ici, si vous cherchez à modifier la position d'un sprite précis dans une animation, sachez que vous pouvez !

Je m'explique, si vous voulez par exemple changer la position de la frame 13 de l'animation « idleB », vous devez tout simplement avoir un fichier « .meta.yaml » avec le nom de votre frame (ici « idleB13.meta.yaml ») et vous y mettez simplement de combien de pixels vous voulez décaler votre sprite :

Intérieur de votre dossier avec vos sprites



Intérieur du fichier « .meta.yaml »



Ici, le sprite « idleB13 » sera décalé uniquement de 12 pixels à gauche.

Attention : Votre fichier « .meta.yaml » doit être dans le même dossier que le sprite concerné.

PARAMÈTRES DE SKIN MOD HELPER PLUS :



Variant

Otherself variant

Silhouette variant



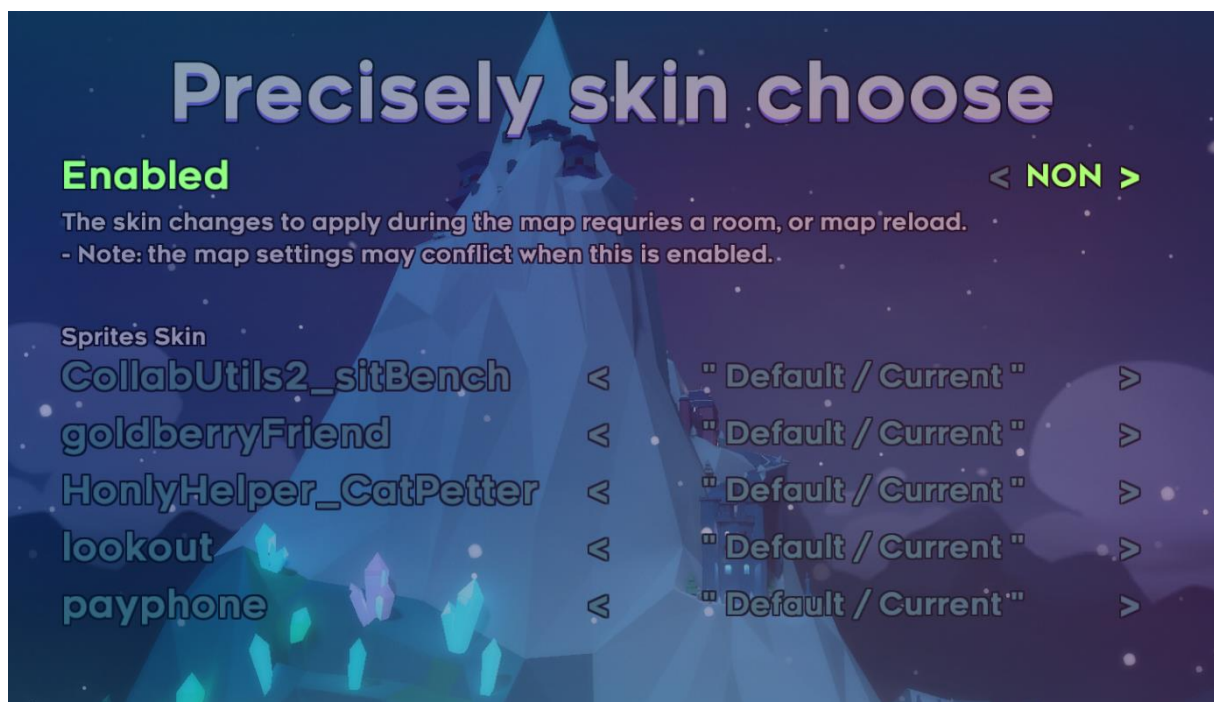
<https://gamebanana.com/mods/53734>

PS : Je n'en ai pas pour mon skin (je vous laisse le lien du mod (fait par Misirter E) pour voir par vous-même).



« ADVANCED OPTIONS » :

Ces paramètres ci-dessous sont pour savoir si vous souhaitez utiliser des animations spécifiques ou alors celles par défaut (celles dans les dossiers de votre skin).



CELESTE NET :

« Celeste NET » est un mod permettant de faire du multijoueur sur Celeste, vous pouvez également montrer votre skin à vos amis !

Il vous suffira que la personne qui joue avec vous possède votre fichier ZIP afin que Celeste NET puisse reconnaître le skin et l'affiche de son côté.

RECOMMANDATIONS LOGICIELS :

Petite section, si vous ne savez pas vraiment par où commencer afin de créer votre skin, voici quelques logiciels gratuits pour faire de la création de sprites et créer d'autres choses :

- Paint.Net
- Krita
- Piskel
- GraphicsGale