

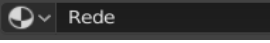
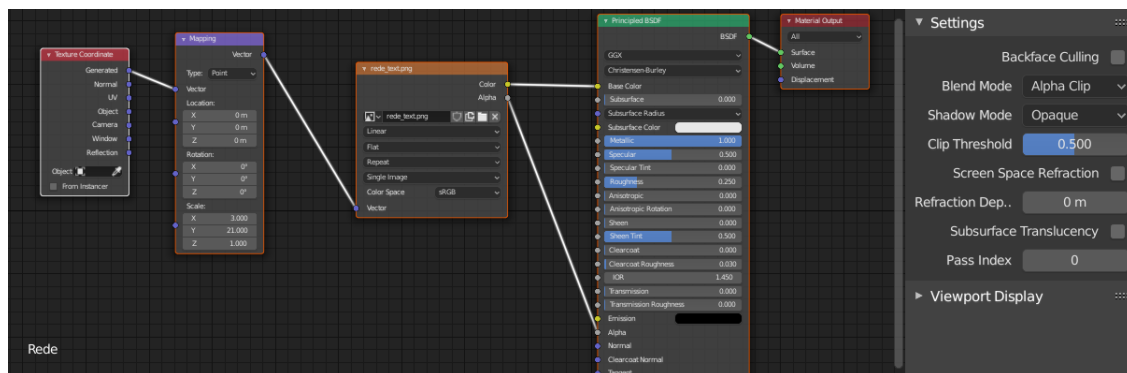



# Shading - Parte III

## 1. Texturas com transparências (canal alfa)


- Abrir o projeto *FCG\_05\_Shading\_C.blend*;
- Confirmar que o *workspace* selecionado é o **Shading** e que o **Viewport Shading** está em modo *Material Preview* (  );
- Selecionar o modelo *Parede*, localizado na *Collection*;
- Em **Properties > Material Properties** (  ), criar um novo material (+ **New**) e nomear o material criado como “Rede” (  );
- No **Shader Editor** adicionar um *node Image Texture* (**Shift+A > Texture > Image Texture**);
- No *node Image Texture* clicar em *Open* e carregar o ficheiro de imagem *Rede.png*. Este é um ficheiro .png com um canal alfa (*alpha channel*), indicando que certas partes da imagem são transparentes;
- Ligar o *output Color* ao *input Base Color* do *Principled BSDF*;
- Para mapear corretamente o gradeamento, criar um *node Texture Coordinate* (**Shift+A > Input > Texture Coordinate**) e colocá-lo à esquerda no **Shader Editor**;
- De seguida, criar um *node Mapping* (**Shift+A > Vector > Mapping**) e em *Scale* alterar o valor no eixo **X** para **3.0** e no eixo **Y** para **21.0**;
- Ligar o *output Generated* do *node Texture Coordinate* ao *input Vector* do *node Mapping*;
- Ligar o *output Vector* do *node Mapping* ao *input Vector* do *node Image Texture*;
- De seguida, ligar o *output Alpha* do *node Image Texture* ao *input Alpha* do *Principled BSDF*;
- Abrir o painel **Settings (N)** e alterar o campo *Blend Mode* para *Alpha Clip*;
- No *Principled BSDF*, alterar o valor dos campos *Metallic* para **1.0** e *Roughness* para **0.25**.



## 2. Texturas aplicadas a formas não planares

- Selecionar o modelo *Globo*, localizado na *Collection3*;
- Em **Properties > Material Properties** () , criar um novo material (+ **New**) e dar-lhe o nome de “Globo terrestre”;
- No **Shader Editor** adicionar um *node* do tipo *Image Texture* (**Shift+A > Texture > Image Texture**) e, clicando em *Open*, carregar o ficheiro de imagem *terra.png*;
- Ligar o *output Color* ao *input Base Color* do *Principled BSDF*;
- Pode observar-se que a textura da imagem (visualizar ficheiro *terra.png* no *Image Editor*) não surge aplicada no modelo. Para corrigir esta situação, criar um *node Texture Coordinate* (**Shift+A > Input > Texture Coordinate**) e um *node Mapping* (**Shift+A > Vector > Mapping**);
- Ligar o *output Generated* do *node Texture Coordinate* ao *input Vector* do *node Mapping*, e o *output Vector* do *node Mapping* ao *input Vector* do *node Image Texture*;
- Verificar que o mapeamento da imagem ainda não está correto;
- No *node Image Texture*, substituir a opção *Flat* por *Sphere* (alterando o método de projeção da imagem 2D no objeto);
- No *node Principled BSDF*, alterar o valor do *Roughness* para **0.3**, ficando o Globo com o aspeto da figura abaixo;



- Selecionar agora o modelo *Caixote*, localizado na *Collection3*;
- Em **Properties > Material Properties** () , criar um novo material (+ **New**) e dar-lhe o nome de “Madeira caixote”;
- No **Shader Editor** adicionar um *node* do tipo *Image Texture* e abrir o ficheiro de imagem *caixote.png*;
- Ligar o *output Color* ao *input Base Color* do *Principled BSDF* e observar que a textura da imagem (visualizar ficheiro *caixote.png* no *Image Editor*) não se encontra bem aplicada;
- Criar um *node Texture Coordinate* (**Shift+A > Input > Texture Coordinate**) e um *node Mapping* (**Shift+A > Vector > Mapping**);
- Ligar o *output Generated* do *node Texture Coordinate* ao *input Vector* do *node Mapping*, e o *output Vector* do *node Mapping* ao *input Vector* do *node Image Texture*;
- Para que o mapeamento da imagem fique correto, no *node Image Texture*, substituir a opção *Flat* por *Box*;
- No *node Principled BSDF*, alterar o valor do *Roughness* para **1.0**, ficando o Caixote com o aspeto da figura abaixo.

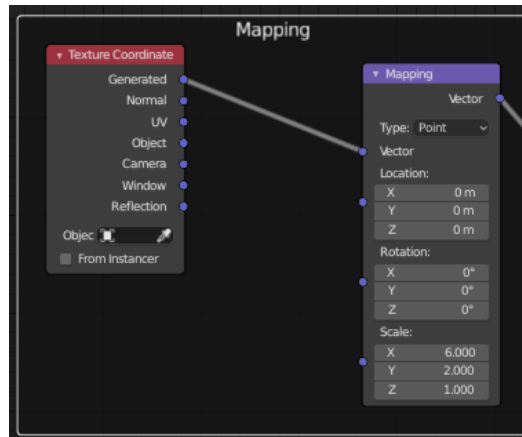


### 3. Texturas procedimentais (*procedural textures*)

Em Computação Gráfica podem ser criadas texturas através de algoritmos e expressões matemáticas (texturas procedimentais). Este tipo de texturas tem a vantagem de não ter limites de resolução (como acontece com uma textura criada a partir de uma imagem *raster*).

- Assegurar que o *Object Mode* está selecionado;
- Selecionar o modelo *Chão*, localizado na *Collection*;
- Para criar o material base:

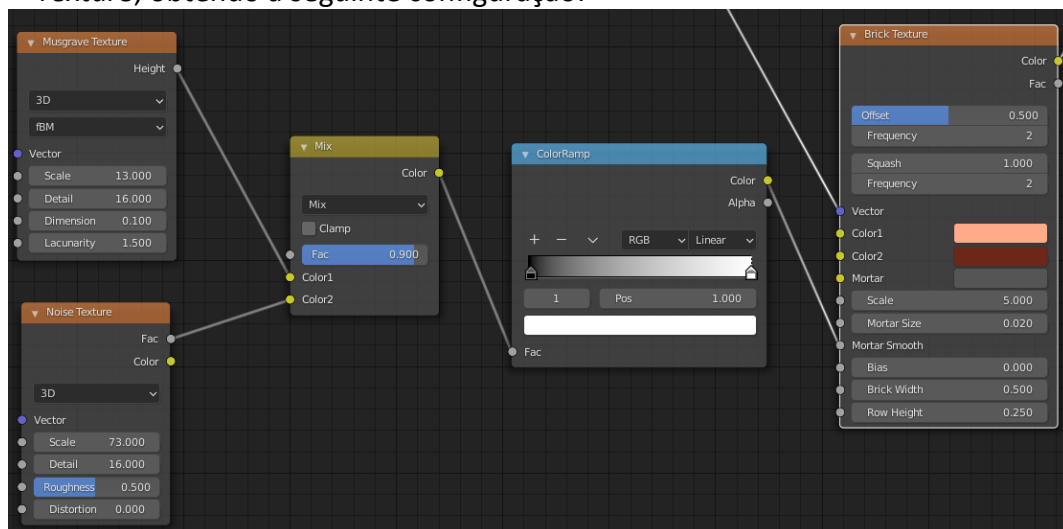
- Em **Properties > Material Properties** (🔴), criar um novo material (+ **New**) e nomear o material criado como “Tijolos” (📁 Tijolos);
- No **Shader Editor** adicionar um *node* do tipo **Brick Texture** (**Shift+A > Texture > Brick Texture**);
- Ligar o *output Color* ao *input Base Color* do **Principled BSDF**;
- Pode observar-se que a textura dos tijolos não se encontra mapeada corretamente (dimensões exageradas dos tijolos). Para corrigir esta situação devemos criar um *node Texture Coordinate* (**Shift+A > Input > Texture Coordinate**) e colocá-lo à esquerda no **Shader Editor**;
- De seguida, criar um *node Mapping* (**Shift+A > Vector > Mapping**) e em *Scale* alterar o valor no eixo **X** para **6.0** e no eixo **Y** para **2.0**;
- Ligar o *output Generated* do *node Texture Coordinate* ao *input Vector* do *node Mapping*;
- Ligar o *output Vector* do *node Mapping* ao *input Vector* do *node Brick Texture*;
- Selecionar os *nodes Texture Coordinate* e *Mapping* e pressionar **Ctrl+J** para os agrupar numa moldura (*Frame*). Esta ferramenta permite organizar os *nodes* de forma a que seja mais rápido compreender a função de cada um. Clicar com o **RMB** na moldura e selecionar *Rename*. Nomear esta moldura como *Mapping*:



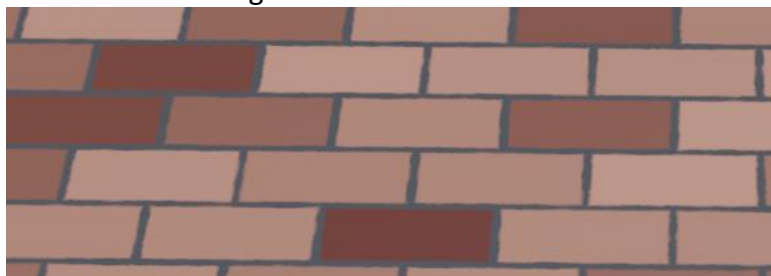
- No *node Brick Texture*, alterar a cor dos tijolos (*Color 1* e *Color 2*) e do cimento (*Mortar*) para os seguintes valores RGB:
  - **Color 1**: 1; 0.4; 0.25;
  - **Color 2**: 0.15; 0.02; 0.01;
  - **Mortar**: 0.09; 0.09; 0.09;

- Para que o cimento seja mais irregular e tenha um ar menos artificial:

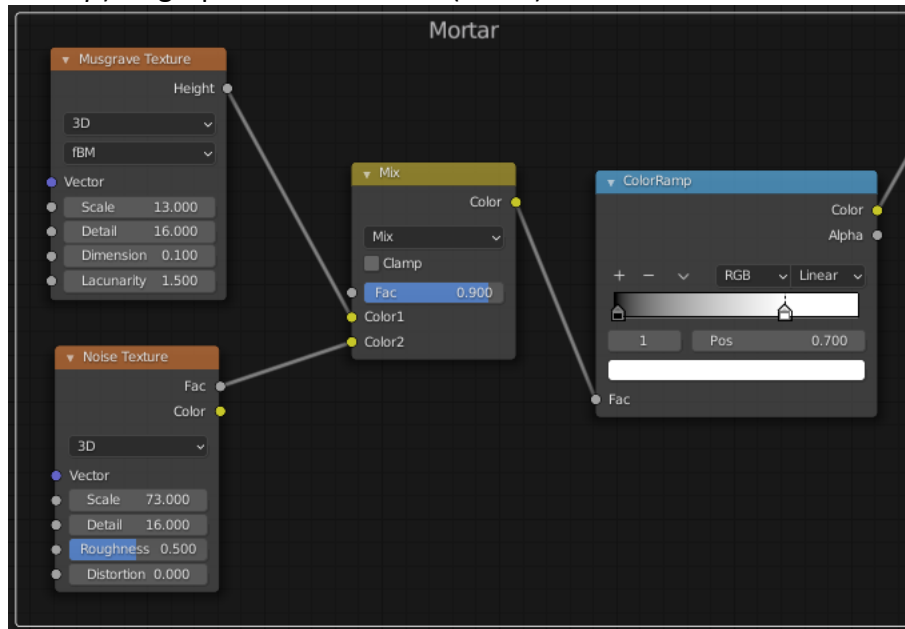
- Começar por adicionar um *node Musgrave Texture* (**Shift+A > Texture > Musgrave Texture**) com os seguintes parâmetros:
  - **Scale: 13.0;**
  - **Detail: 16.0;**
  - **Dimension: 0.1;**
  - **Lacunarity: 1.5;**
- De seguida, adicionar um *node Noise Texture* (**Shift+A > Texture > Noise Texture**) com os seguintes parâmetros:
  - **Scale: 73.0;**
  - **Detail: 16.0;**
- Adicionar um *node MixRGB* (**Shift+A > Color > MixRGB**) e definir o valor de *Fac* como **0.9;**
- Ligar o *output Height* da *Musgrave Texture* e o *output Fac* da *Noise Texture* aos *inputs Color* do *MixRGB*;
- Adicionar um *node ColorRamp* (**Shift+A > Converter > ColorRamp**) e ligar o *output Color* do *node MixRGB* ao *input Fac* do *node ColorRamp*;
- Ligar o *output Color* do *node ColorRamp* ao *input Mortar Smooth* do *node Brick Texture*, obtendo a seguinte configuração:



- Observar que os limites do cimento podem ser alterados utilizando os *sliders* no gradiente do *node ColorRamp*. Mover o *slider* branco para 0.7, obtendo um resultado semelhante ao seguinte:

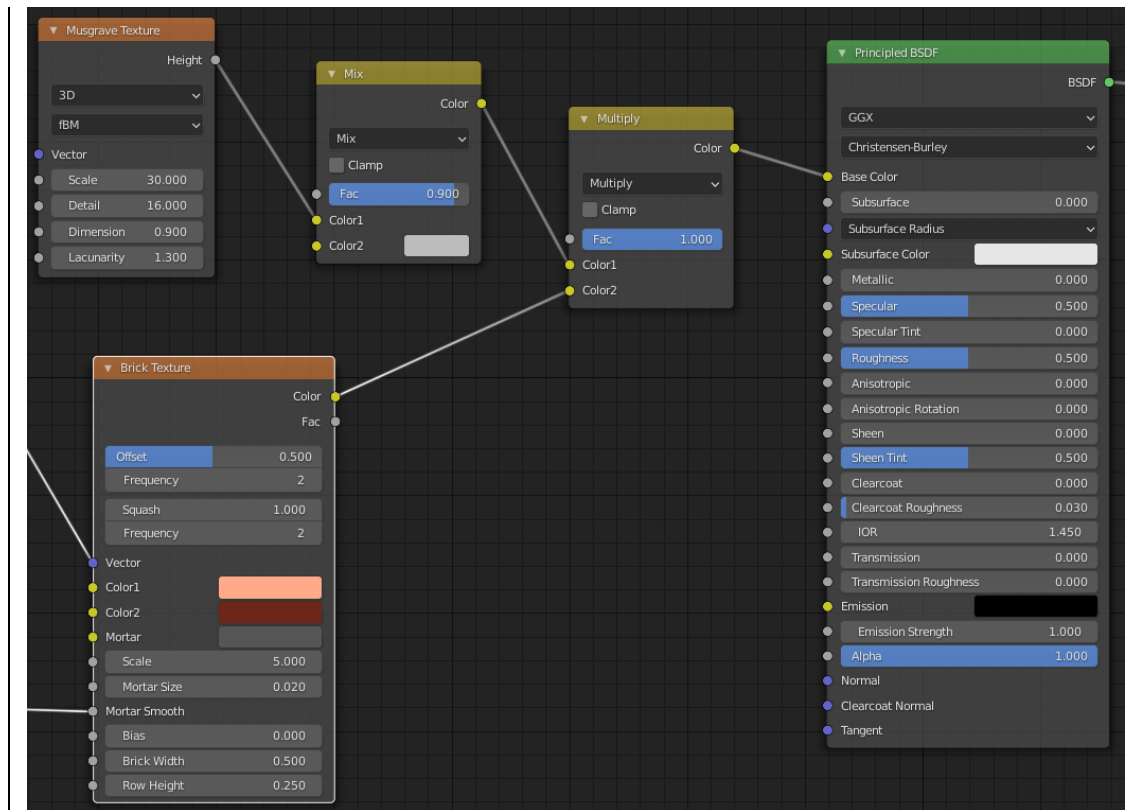


- Selecionar os quatro *nodes* criados (*Musgrave Texture*, *Noise Texture*, *MixRGB* e *ColorRamp*) e agrupar numa moldura (**Ctrl+J**) com o nome *Mortar*:

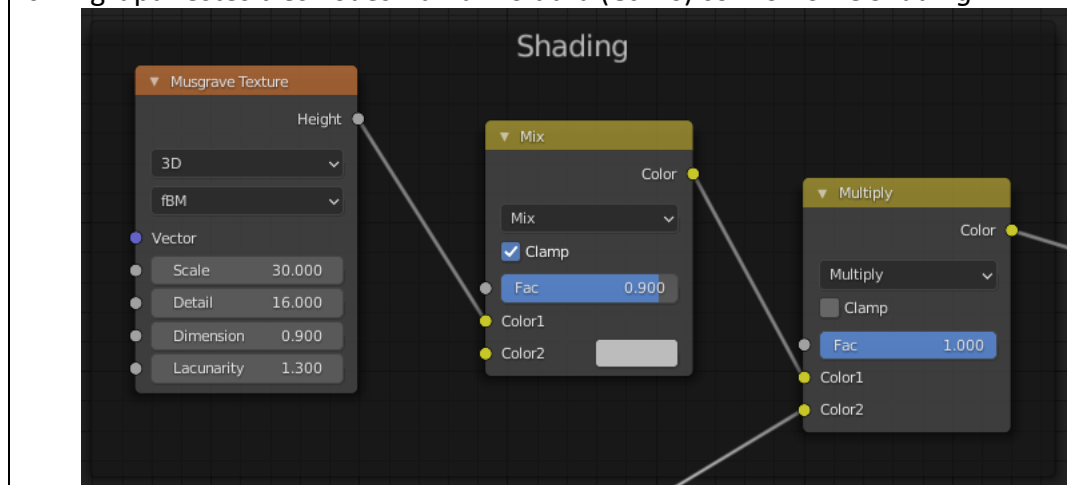


- Para tornar a superfície dos tijolos menos monótona:




- Começar por adicionar um novo *node Musgrave Texture* (**Shift+A > Texture > Musgrave Texture**) com os seguintes parâmetros:
  - **Scale: 30.0;**
  - **Detail: 16.0;**
  - **Dimension: 0.9;**
  - **Lacunarity: 1.3;**
- Adicionar um *node MixRGB* (**Shift+A > Color > MixRGB**) e definir o valor **0.9** para o campo *Fac*;
- Ligar o *output Height* de *Musgrave Texture* ao *input Color 1* do *MixRGB*;
- Criar mais um *node MixRGB* (**Shift+A > Color > MixRGB**) e alterar, neste *node*, o campo *Mix* para *Multiply*. De seguida, mudar o valor do campo *Fac* para **1.0**;
- Remover a ligação do *output Color* do *Brick Texture* ao *Base Color* do *Principled BSDF*;
- Ligar o *output Color* do *node MixRGB* ao *input Color 1* do *node MixRGB* (que tem o nome de *Multiply*) e o *output Color* do *node Brick Texture* ao *input Color 2* do *node MixRGB* (que tem o nome de *Multiply*);
- De seguida, ligar o *output Color* do *node MixRGB* (que tem o nome de *Multiply*) ao *input Base Color* do *Principled BSDF*;



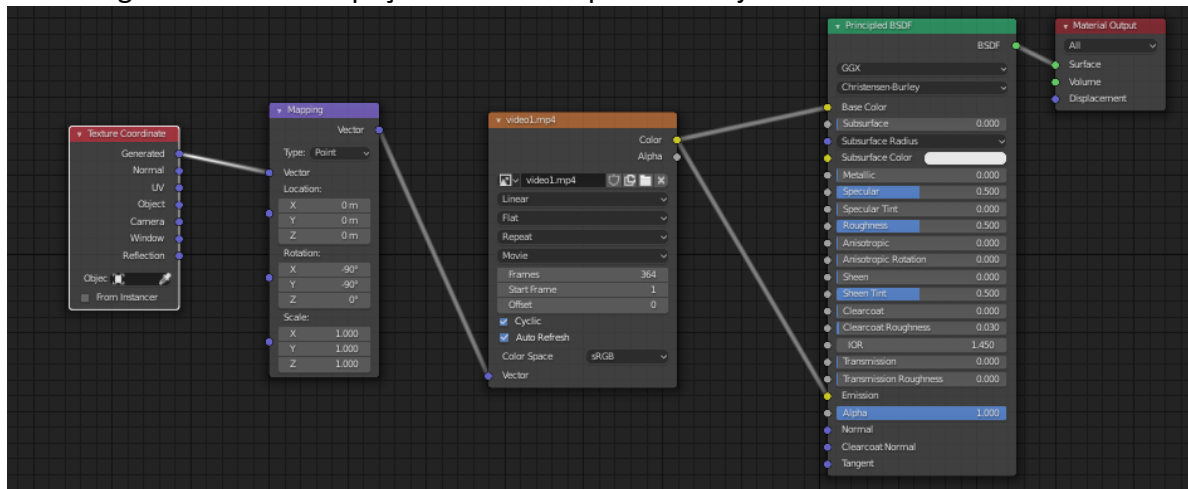
- Agrupar estes três *nodes* numa moldura (**Ctrl+J**) com o nome *Shading*:



#### 4. Utilização de vídeo para obtenção de texturas animadas

- Criar um novo projeto *.blend*, apagar o cubo e salvar o ficheiro na mesma pasta onde se encontra o ficheiro *video1.mp4*;
- Mudar o *workspace* para *Shading*;
- Ir a **Edit > Preferences > Add-ons** e no campo de procura escrever “*import images as planes*”;
- Adicionar o *add-on Import-Export: Import Images as Planes*, se ainda não tiver sido ativado, e gravar as alterações (*Save Preferences*);
- No **3D Viewport**, carregar em **Shift+A > Image > Images as Planes** e seleccionar o ficheiro *video1.mp4*;
- No **Shader Editor** pode observar-se que foi adicionado um *node Image Texture* com o ficheiro de vídeo, indicando o número de *frames* correspondentes à duração do mesmo (neste caso, 364), o qual se encontra ligado ao *Principled BSDF*;
- Clicar na Barra de Espaços para confirmar que o vídeo está a correr no plano;
- Esta é a forma mais rápida de importar vídeo como textura. No entanto, este fica automaticamente aplicado a um plano;
- Para exemplificar a aplicação de uma textura vídeo a um modelo, eliminar o vídeo anterior e importar o ficheiro *Ecrã.fbx* (**File > Import > FBX**);
- Entrar em **Edit Mode** e, em modo de seleção de arestas com toda a malha seleccionada, clicar em **RMB > Clear Sharp**;
- Em **Properties > Material Properties** () , alterar o nome do material para *Monitor*;
- No **Shader Editor**, eliminar o *node NormalMap* (se existir);
- No *Principled BSDF*, alterar a *Base Color* para preto e definir o valor de *Roughness* como **0.1**;
- Em **Properties > Material Properties** () , adicionar um segundo *material slot* () e carregar em **+ New**;
- Alterar o nome do novo material para *Ecrã*;
- Em modo de seleção de faces, seleccionar a face do modelo correspondente ao ecrã e (tendo o material *Ecrã* seleccionado) carregar em *Assign*;
- No **Shader Editor**, adicionar um *node Image Texture* (**Shift+A > Texture > Image Texture**), clicar em *Open* e carregar o ficheiro *video1.mp4*;
- No campo *Frames* inserir o valor **364** e seleccionar *Cyclic* e *Auto Refresh*;
- Ligar o *output Color* do *node Image Texture* aos *inputs Base Color* e *Emission* do *Principled BSDF*;
- Colocar o valor do campo *Roughness* do *node Principled BSDF* a **0**;
- Passar para **Object Mode**, carregar na Barra de Espaços e observar que a textura vídeo não se encontra com a orientação correcta. Para a corrigir, criar um *node Texture Coordinate* (**Shift+A > Input > Texture Coordinate**) e um *node Mapping* (**Shift+A > Vector > Mapping**);
- No campo *Rotation* do *node Mapping*, alterar o valor do eixo **X** para **-90°** e do eixo **Y** para **-90°**;
- Ligar o *output Generated* do *node Texture Coordinate* ao *input Vector* do *node Mapping*;
- Ligar o *output Vector* do *node Mapping* ao *input Vector* do *node Image Texture*;

- Carregar na Barra de Espaços e observar que o vídeo já está corretamente orientado.



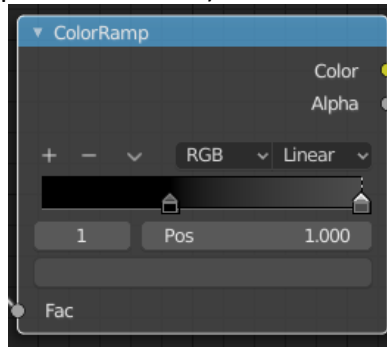
## 5. Exercício proposto

Complete o exercício 3, acrescentando mais realismo ao material do objeto *Chão (Tijolos)*, através da criação de desgaste e fendas.

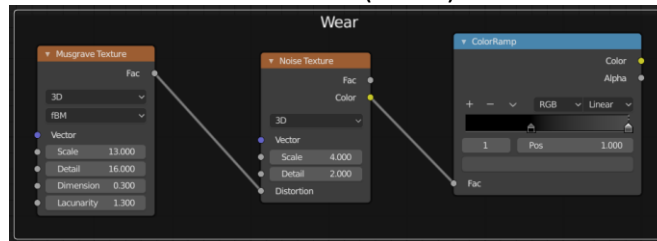
- Para adicionar desgaste à superfície:
  - Começar por adicionar um *node Musgrave Texture* (**Shift+A > Texture > Musgrave Texture**) com os seguintes valores:
    - **Scale: 13.0;**
    - **Detail: 16.0;**
    - **Dimension: 0.3;**
    - **Lacunarity: 1.3;**
  - Criar um *node Noise Texture* (**Shift+A > Texture > Noise Texture**) com o valor da escala a **4.0**;
  - Ligar o *output Fac* do *node Musgrave Texture* ao *input Distortion* do *node Noise Texture*;
  - Adicionar um novo *node ColorRamp* (**Shift+A > Converter > ColorRamp**);
  - Ligar o *output Color* do *node Noise Texture* ao *input Fac* do *node ColorRamp*;



- Colocar o *slider* preto na posição 0.4 e alterar a cor do *slider* branco para um cinzento escuro (HSV: 0; 0; 0.3 - para alterar a cor, seleccionar o *slider* e carregar na barra de cor branca por cima de *Fac*):

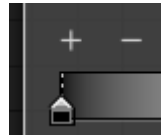


- Agrupar estes três *nodes* numa moldura (**Ctrl+J**) com o nome *Wear*:

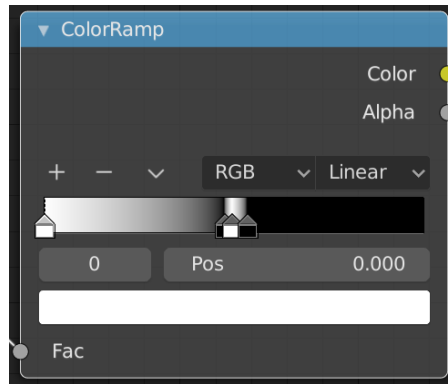


- De seguida, para adicionar fendas na superfície:

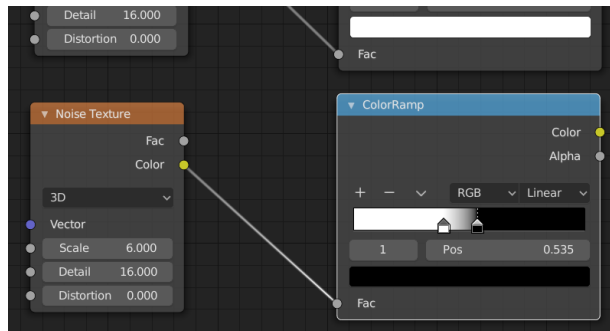
- Começar por criar um *node Noise Texture* (**Shift+A > Texture > Noise Texture**) com o valor da escala (*Scale*) a **6.0** e o valor de detalhe (*Detail*) a **16.0**;
- Adicionar mais um *node ColorRamp* (**Shift+A > Converter > ColorRamp**) e ligar o *output Color* do *node Noise Texture* ao *input Fac* do *node ColorRamp*;
- Mover o *slider* preto para a posição **0.475**;
- No *node ColorRamp* podemos utilizar o botão **+** para adicionar mais *sliders* ao gradiente:



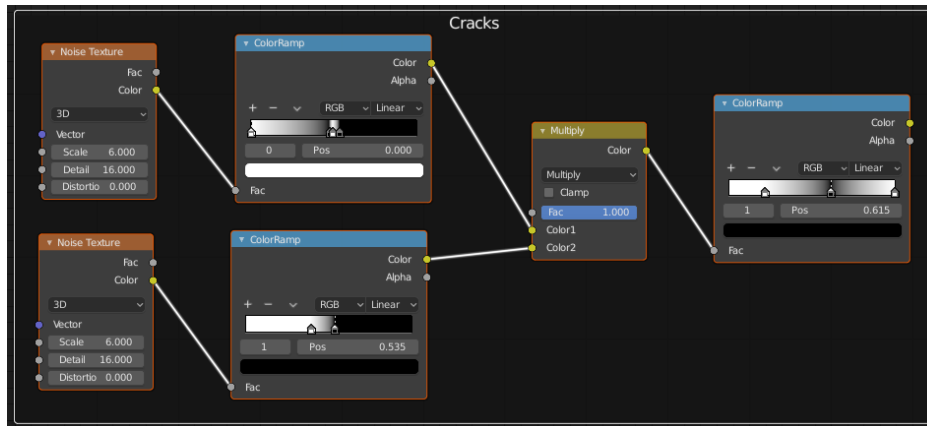
- Adicionar os seguintes *sliders*:
  - Branco: **0.0**;
  - Branco: **0.495**;
- Por fim, seleccionar o *slider* branco na posição 1.0, alterar a cor para preto e modificar a sua posição para **0.535**, obtendo o seguinte resultado:



- Selecionar os *nodes* *Noise Texture* e *ColorRamp* e duplicar (**Shift+D**);
- Neste novo *node ColorRamp*, utilizar o botão - (—) para retirar dois *sliders*, ficando apenas com dois. Estes devem ter os seguintes valores:  
 - Branco: **0.39**;  
 - Preto: **0.535**;

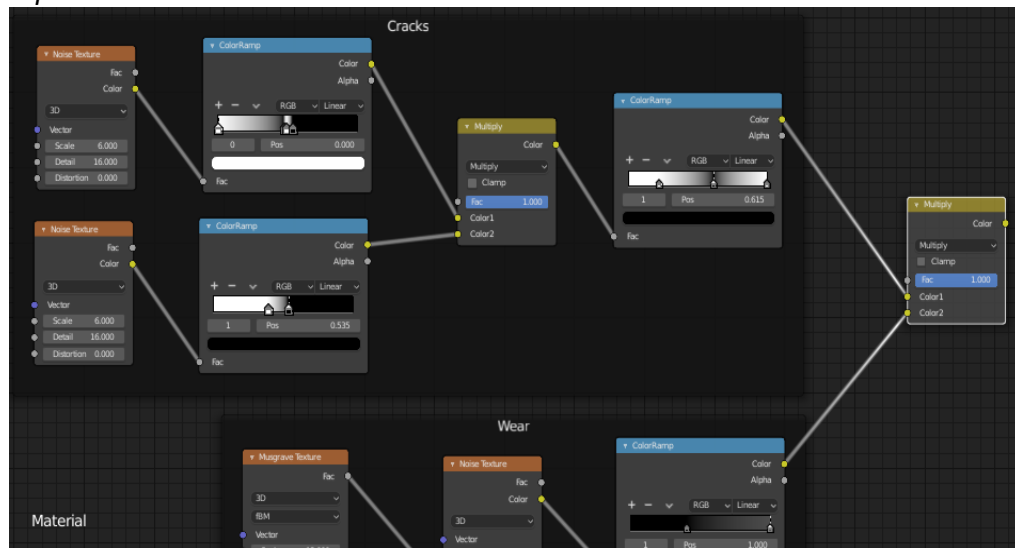


- Adicionar *node MixRGB* (**Shift+A > Color > MixRGB**) e alterar o campo *Mix* para *Multiply*. Alterar o campo *Fac* para **1.0**. Ligar os *outputs Color* das duas *ColorRamps* aos *inputs Color 1* e *2*;
- Adicionar um *node ColorRamp* para servir de controlador da configuração das fendas (**Shift+A > Converter > ColorRamp**). No gradiente, definir os seguintes *sliders*:  
 - Branco: **0.215**;  
 - Preto: **0.615**;  
 - Branco: **1.0**;
- Ligar o *output Color* do *node MixRGB* ao *input Fac* do *node ColorRamp*;
- Selecionar estes seis *nodes* e agrupá-los numa moldura (**Ctrl+J**) com o nome *Cracks*:

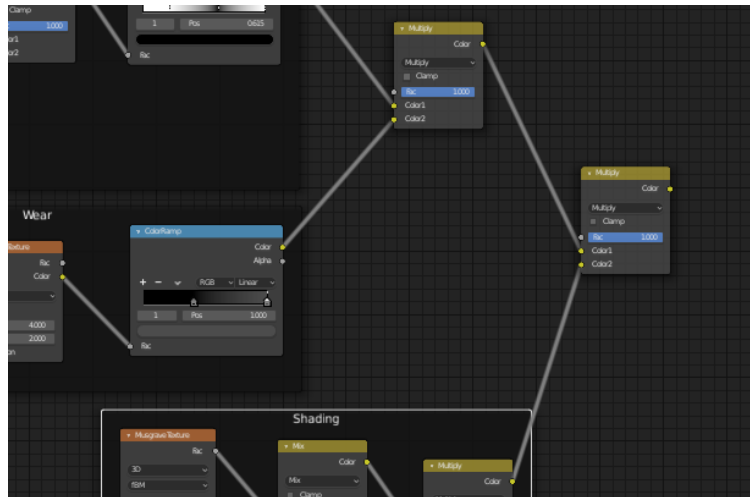


- Para combinar o desgaste com as falhas:

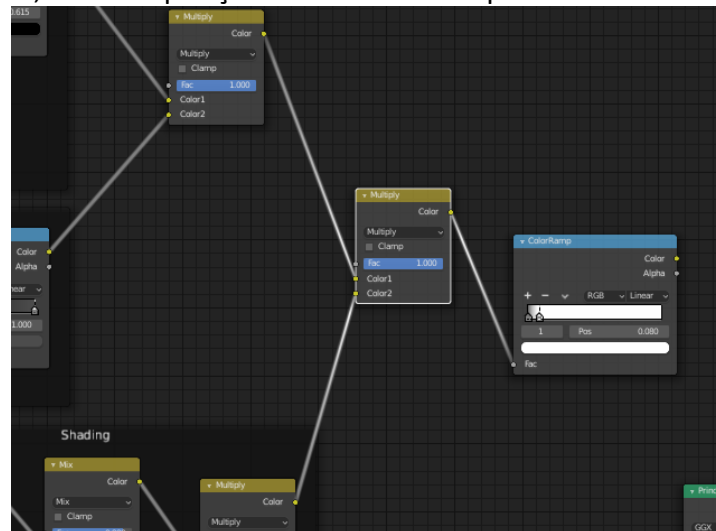
- Adicionar um *node MixRGB* (**Shift+A > Color > MixRGB**), alterar o campo *Mix* para *Multiply* e o campo *Fac* para **1.0**;
- Ligar os *outputs Color* dos *nodes ColorRamp* das molduras *Wear* e *Cracks* aos *inputs Color 1 e 2*:



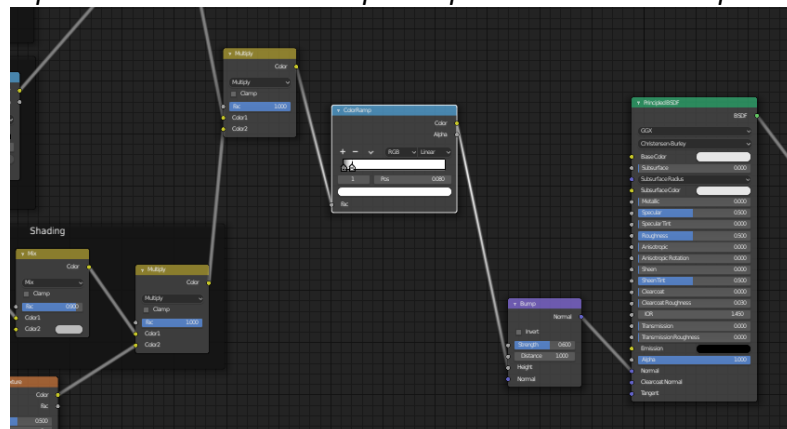
- Duplicar este último *node MixRGB* (**Shift+D**) e ligar o *output Color* do *MixRGB* ao *input Color1* do *MixRGB* criado;
- Ligar o *output Color* do *MixRGB* (que tem o nome de *Multiply*) da moldura *Shading* ao *input Color2*;
- Remover a ligação do *output Color* do *MixRGB* (que tem o nome de *Multiply*) da moldura *Shading* ao *Base Color* do *Principled BSDF*;



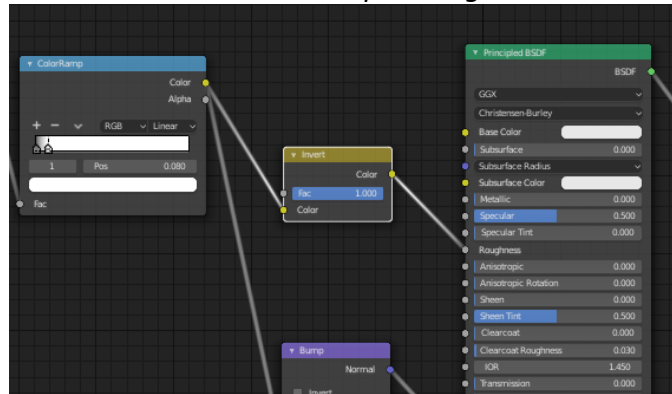
- Adicionar um **node ColorRamp** (**Shift+A > Converter > ColorRamp**) e ligar o **output Color** do último **node MixRGB** ao **input Fac** do **node ColorRamp**;
- No gradiente, definir a posição do **slider branco** para **0.08**:



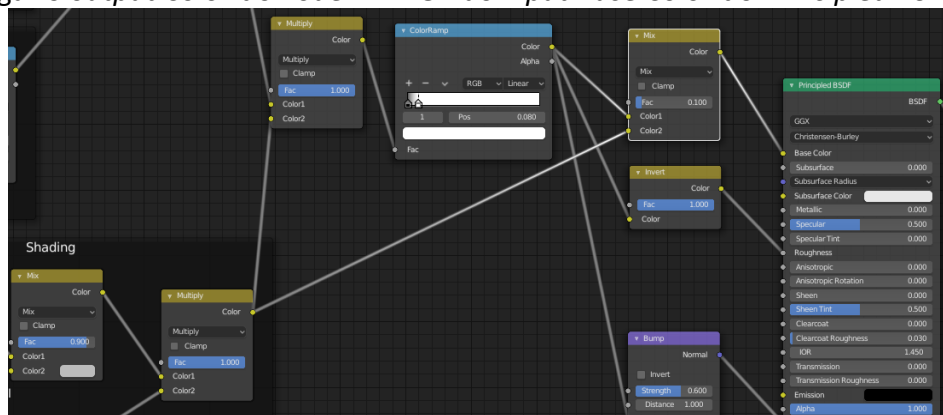
- Adicionar um **node Bump** (**Shift+A > Vector > Bump**) com o valor **Strength** a **0.6**;
- Ligar o **output Color** do **node ColorRamp** ao **input Height** do **node Bump**;
- Ligar o **output Normal** do **node Bump** ao **input Normal** do **Principled BSDF**:



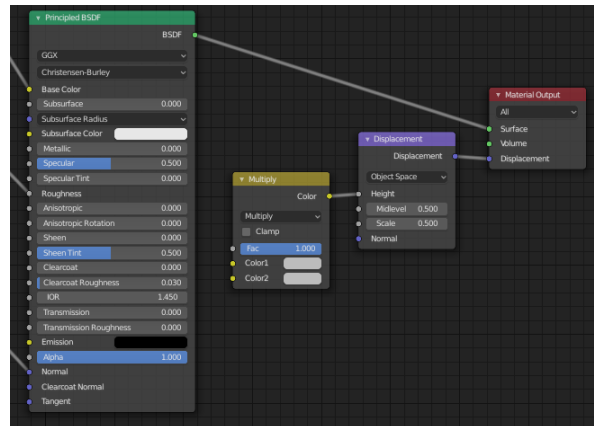
- Adicionar um node **Invert** (**Shift+A > Color > Invert**) e ligar o output **Color** do node **ColorRamp** ao input **Color** do node **Invert**;
- Ligar o output **Color** do node **Invert** ao input **Roughness** do **Principled BSDF**:



- Adicionar um node **MixRGB** (**Shift+A > Color > MixRGB**) com o valor **Fac** a **0.1**;
- Ligar o output **Color** do node **ColorRamp** ao input **Color1** do node **MixRGB** e o output **Color** do **MixRGB/Multiply** da moldura **Shading** ao input **Color2**;
- Ligar o output **Color** do node **MixRGB** ao input **Base Color** do **Principled BSDF**:



- Adicionar um node **Displacement** (**Shift+A > Vector > Displacement**) com **Scale** a **0.5**;
- Ligar o output **Displacement** ao input **Displacement** do node **Material Output**;
- Adicionar node **MixRGB** (**Shift+A > Color > MixRGB**) e alterar o campo **Mix** para **Multiply**.
- No node **MixRGB** criado, alterar o campo **Fac** para **1.0** e ligar o output **Color** ao input **Height** do node **Displacement**:



**Atenção:** A utilização do *Displacement* é mais exigente em termos de processamento e poderá provocar um *crash* em máquinas menos potentes. É aconselhável salvar o trabalho antes de avançar para este passo.

- Ligar o *output Color* da *ColorRamp* da moldura *Cracks* ao *input Color1* do último *MixRGB* criado;
- Ligar o *output Color* do *MixRGB* da moldura *Shading* ao *input Color2*:

Diagrama final do material *Tijolos*:

