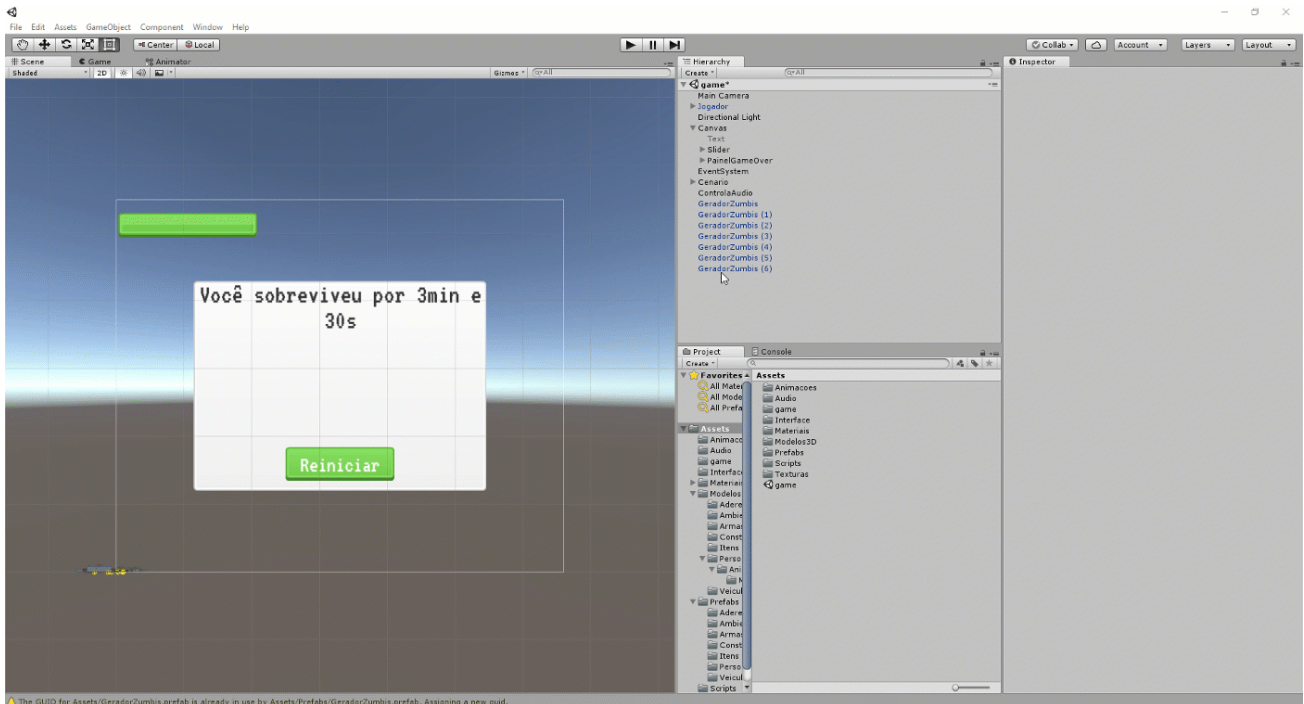


## Salvando Melhor Tempo

Seria legal se pudéssemos salvar o maior tempo que sobrevivemos para comparar a nossa performance no nosso jogo, não seria?

Vamos fazer isto então? Para isso vamos duplicar o texto que guarda quanto tempo sobrevivemos no nosso painel de Game Over, gerando um novo texto. Só que vamos ancorá-lo no meio do painel.



Agora vamos preencher o texto via programação? Para isto precisamos de duas variáveis: uma para referenciar nosso texto, e outra para guardar nossa melhor pontuação.

```
public Text TextoPontuacaoMaxima;  
private float tempoPontuacaoSalvo;
```

Vamos criar um método que ajusta este novo texto?

Como já calculamos nosso valor de minutos e segundos no script de Game Over, não faz sentido recalcularmos esse valor novamente. Vamos então criar um método que recebe estes valores já calculados como parâmetros.

```
void AjustarPontuacaoMaxima (int min, int seg)  
{  
}
```

Agora, temos que passar valor para as nossas variáveis. Na `tempoPontuacaoSalvo`, vamos colocar o valor do nosso tempo de jogo, mas somente se o valor feito for maior do que o valor que já temos salvo, pois nesse caso teremos uma nova pontuação mais alta.

```
void AjustarPontuacaoMaxima (int min, int seg)
{
    if(Time.timeSinceLevelLoad > tempoPontuacaoSalvo)
    {
        tempoPontuacaoSalvo = Time.timeSinceLevelLoad;
    }
}
```

Se isso acontecer, também temos que preencher a nossa `TextoPontuacaoMaxima`. Mas vamos ver uma nova forma de fazer isso. Vamos utilizar o `string.Format`: nele você passa suas variáveis como valores `{0}`, `{1}`, `{2}` e após o método colocamos a ordem desses valores após a vírgula.

```
void AjustarPontuacaoMaxima (int min, int seg)
{
    if(Time.timeSinceLevelLoad > tempoPontuacaoSalvo)
    {
        tempoPontuacaoSalvo = Time.timeSinceLevelLoad;
        TextoPontuacaoMaxima.text = string.Format("Seu melhor tempo é {0}min e {seg}s", min, seg);
    }
}
```

Depois disso, como essa nova pontuação é a maior pontuação que já temos salva, temos que guardá-la. Para guardar uma pontuação mesmo que o jogo seja fechado ou as variáveis serem reiniciadas ao rodar uma nova Cena, vamos utilizar os [PlayerPrefs](https://docs.unity3d.com/ScriptReference/PlayerPrefs.html) (<https://docs.unity3d.com/ScriptReference/PlayerPrefs.html>) da Unity, que são uma forma simples de guardar chave e valor.

No `PlayerPrefs` temos que passar uma `string` de como vai se chamar do valor que iremos guardar.

```
void AjustarPontuacaoMaxima (int min, int seg)
{
    if(Time.timeSinceLevelLoad > tempoPontuacaoSalvo)
    {
        tempoPontuacaoSalvo = Time.timeSinceLevelLoad;
        TextoPontuacaoMaxima.text = string.Format("Seu melhor tempo é {0}min e {seg}s", min, seg);
        PlayerPrefs.SetFloat("PontuacaoMaxima", tempoPontuacaoSalvo);
    }
}
```

Agora podemos chamar este método no nosso Game Over:

```
AjustarPontuacaoMaxima(minutos, segundos);
```

E lembre-se: como agora estamos salvando uma pontuação no início do nosso jogo no método `Start`, temos que buscar a maior pontuação que já fizemos naquele computador.

```
tempoPontuacaoSalvo = PlayerPrefs.GetFloat("PontuacaoMaxima");
```

Agora, quando fazemos uma pontuação máxima nós podemos ver o nosso texto. Mas não seria legal vermos sempre a pontuação máxima quando perdermos?

Então vamos fazer um `if` que verifica se o texto está preenchido. Se o texto não estiver, vamos preenchê-lo. Só que, para isto, temos calcular o valor de segundos e minutos da nossa pontuação máxima para preencher o nosso texto, ao invés do nosso tempo atual.

```
void AjustarPontuacaoMaxima (int min, int seg)
{

    if(Time.timeSinceLevelLoad > tempoPontuacaoSalvo)
    {
        tempoPontuacaoSalvo = Time.timeSinceLevelLoad;
        TextoPontuacaoMaxima.text = string.Format("Seu melhor tempo é {0}min e {seg}s", min, seg);
        PlayerPrefs.SetFloat("PontuacaoMaxima", tempoPontuacaoSalvo);
    }
    if(TextoPontuacaoMaxima.text == "")
    {
        min= (int)tempoPontuacaoSalvo / 60;
        seg = (int)tempoPontuacaoSalvo % 60;
        TextoPontuacaoMaxima.text = string.Format("Seu melhor tempo é {0}min e {1}s", min, seg).
    }
}
```