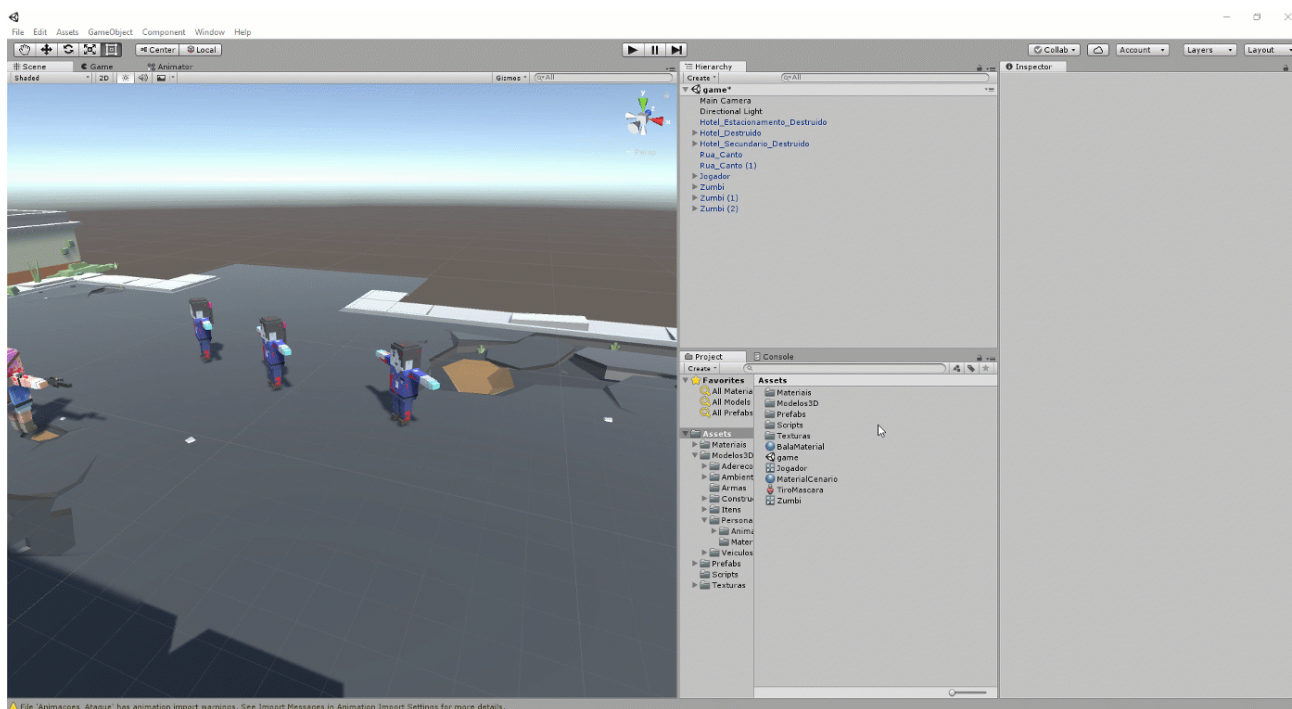


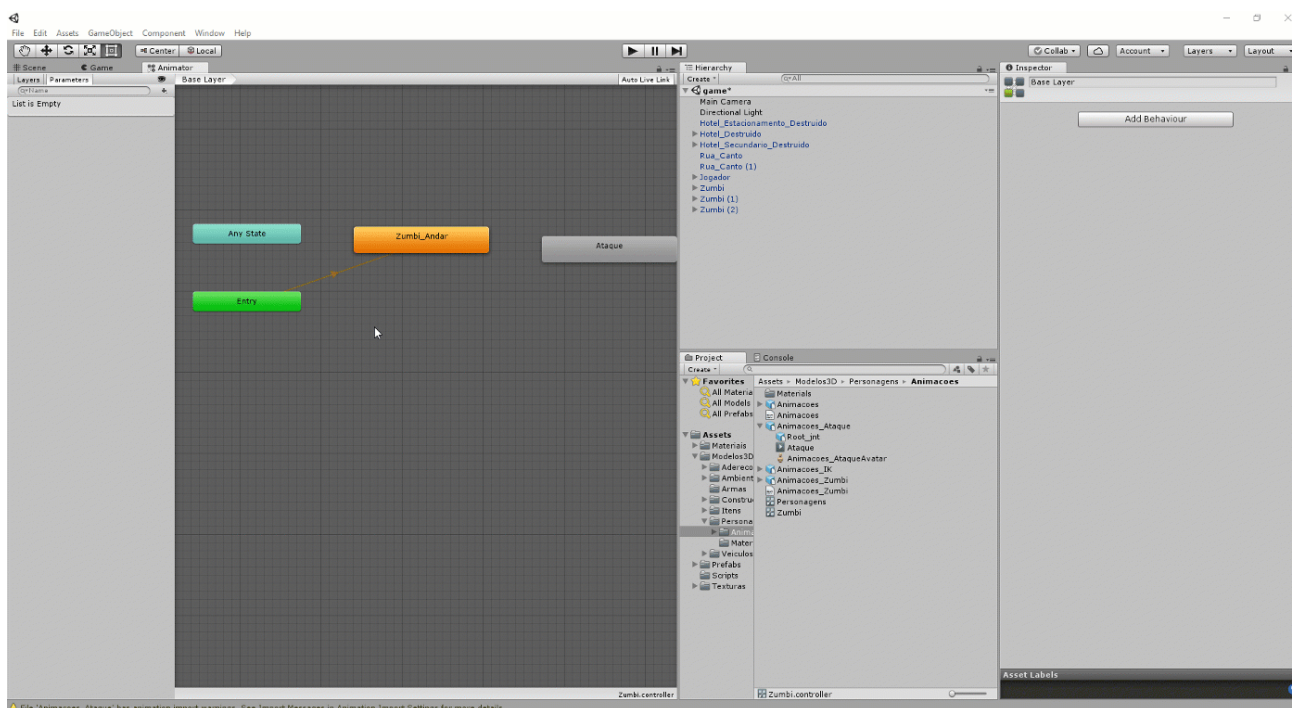
Ataque do Inimigo

Estamos matando nossos Zumbis agora falta ele bater no jogador vamos fazer isto?

Para isto vamos colocar uma nova animação no Inimigo então vamos até a pasta na janela de *Projeto Modelos3D > Personagens > Animacoes* e vamos selecionar o arquivo **Animacoes_Ataque** que já tem a animação cortada temos somente que jogá-la para o Zumbi para isso vamos abrir o arquivo na seta ao lado do nome e selecionar a animação **Ataque** e arrastar para o Animator do Zumbi.



Agora que temos a animação no Animator vamos criar um parâmetro do tipo **Bool** chamado **Atacando** e criar a transição entre as duas animações por este parâmetro, lembre-se de desmarcar a opção **Has Exit Time**.



Agora no `FixedUpdate` do *Script* `ControlaInimigo` vamos incluir a linha que irá fazer ele atacar e parar de atacar utilizando o parâmetro que seria o `GetComponent<Animator>().SetBool()` como já estamos testando se o personagem está perto ou não do Inimigo o que temos que fazer é quando ele está perto mandar atacar e quando está longe parar de atacar incluindo duas linhas no nosso código uma utilizando o `else` deste para saber se ele está perto e outra no final do `if` para parar de atacar.

```
void FixedUpdate()
{
    float distancia = Vector3.Distance(transform.position, Jogador.transform.position);

    Vector3 direcao = Jogador.transform.position - transform.position;

    if (distancia > 2.5)
    {
        GetComponent<Rigidbody>().MovePosition(
            GetComponent<Rigidbody>().position + direcao.normalized * Velocidade * Time.deltaTime);

        Quaternion novaRotacao = Quaternion.LookRotation(direcao);
        GetComponent<Rigidbody>().MoveRotation(novaRotacao);

        GetComponent<Animator>().SetBool("Atacando", false);
    }
    else
    {
        GetComponent<Animator>().SetBool("Atacando", true);
    }
}
```

Também queremos que o nosso personagem rotacione no ataque assim como ele olha para o personagem quando está andando, como queremos que isso aconteça o tempo todo vamos retirar as duas linhas de rotação do `if` para acontecer o tempo todo.

```
void FixedUpdate()
{
    float distancia = Vector3.Distance(transform.position, Jogador.transform.position);

    Vector3 direcao = Jogador.transform.position - transform.position;

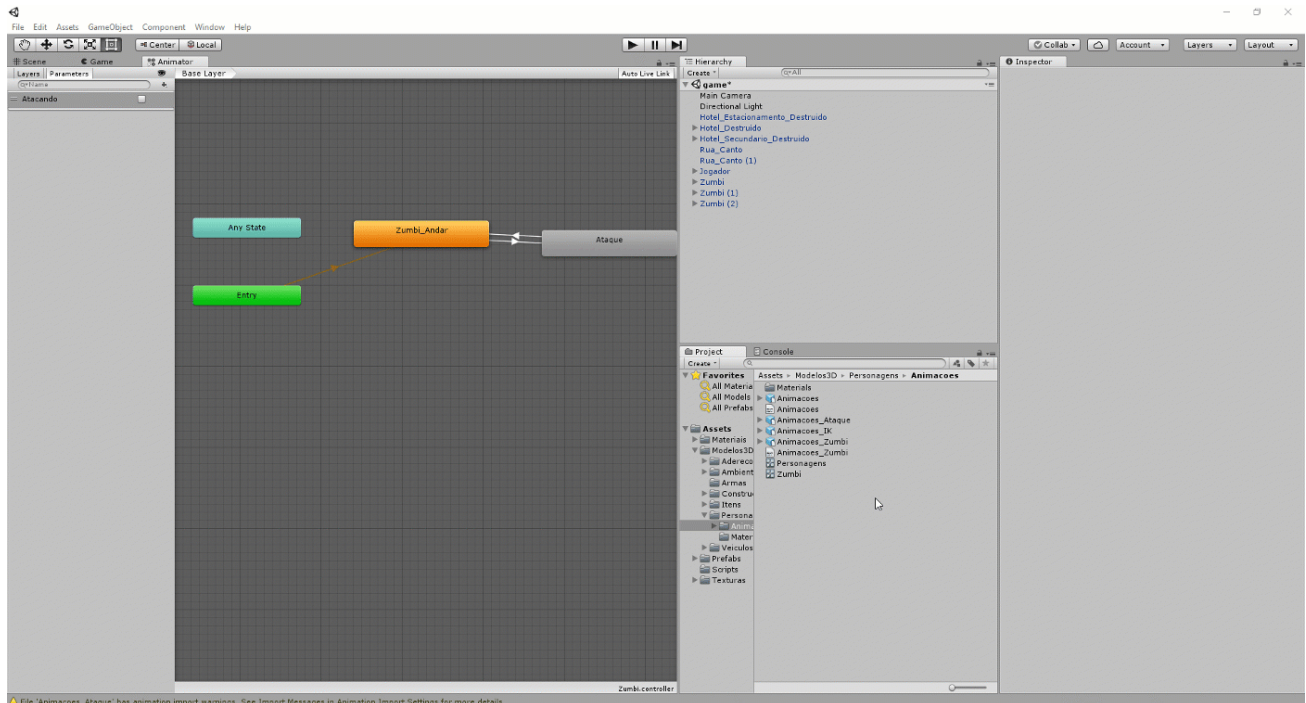
    Quaternion novaRotacao = Quaternion.LookRotation(direcao);
    GetComponent<Rigidbody>().MoveRotation(novaRotacao);

    if (distancia > 2.5)
    {
        GetComponent<Rigidbody>().MovePosition(
            GetComponent<Rigidbody>().position + direcao.normalized * Velocidade * Time.deltaTime);

        GetComponent<Animator>().SetBool("Atacando", false);
    }
    else
    {
        GetComponent<Animator>().SetBool("Atacando", true);
    }
}
```

Legal nosso Inimigo agora está atacando mas como podemos pedir para o **Unity** que algo aconteça exatamente quando ele bate no personagem?

Para isso vamos utilizar os **Eventos de Animação** então selecione o arquivo **Animacoes_Ataque** novamente e com a animação **Ataque** seleciona vamos em **Events** e vamos criar um evento quando o Inimigo está tocando o Personagem e vamos dar o nome de **AtacaJogador**.



Agora para esse evento funcionar temos que criar um método com o mesmo nome do Evento assim ele irá chamar o método automaticamente então no *Script* `ControlaInimigo` vamos criar o seguinte método:

```
void AtacaJogador ()  
{  
  
}
```