

Ajustando animação jogadora

Transcrição

[00:00] Continuando a nossa série de refatoração e ajustes do nosso jogo, a gente precisa ajustar mais um detalhe de animação dentro do nosso jogo. Dessa vez não é no zumbi, é na jogadora, por que? Você repara que ela está parada aqui no meio da tela, eu só iniciei o jogo e ela está andando, a animação está andando. Quando eu mexo o mouse, ela vai para animação de parado, só que não é isso que a gente quer.

[00:23] A animação dela não pode só ser atualizada quando o mouse move. Porque é isso que está acontecendo. Se eu estou na animação de parado e o meu mouse está movendo, ok, se eu deixo o meu mouse parado e vou andando, aperto o meu teclado, ela fica deslizando pela tela, e justamente não é isso que eu quero. A mesma coisa se eu estou andando, deixo meu mouse parado e paro de andar né, solto todas as teclas, ela continua andando.

[00:46] A animação da nossa jogadora, depende muito do meu mouse, ela só atualiza quando eu mexo o meu mouse, a gente precisa arrumar isso. Então, como que a gente vai fazer isso? A gente vai precisar alterar o script dela. Onde que a gente está alterando a animação da jogadora? A gente tem que aqui no evento de mousemove, chamar, a gente está chamando o mudar direção da animação, que é esse evento que ele faz atualização. Mas o que esse evento faz aqui, o que ele precisa fazer?

[01:15] Ele chama o calcular a direção do mouse, passando o evento para lá, então ele precisa da direção do mouse, ele pega o estado atual da jogadora, então ele tem esse script para saber o estado atual, e ele chama o controle de animação, passando a direção do mouse e o estado atual da jogadora. Só que ele só faz isso quando o meu mouse se move, já que ele é um evento de mousemove.

[01:38] O que eu quero fazer, é fazer essa atualização da jogadora, da animação dela, todo frame. Porque todo frame eu posso ou mover o mouse, ou me mexer, ou ficar parado normal enquanto o meu mouse está se movendo, eu posso fazer várias coisas que alteram o estado e a direção da animação da jogadora todo frame. Então, o que a gente vai precisar fazer?

[01:58] No meu método do update. A gente tem esse monte de coisa aqui, o que eu quero fazer, é no final dele chamar um `this.atualiza.atualizaanimacao`, e esse `atualiza animação` vai ser o nosso método, que a gente vai criar. Então, `atualiza animação` vai ser uma função e o que ela vai fazer, ela vai juntamente pegar o controle de animação e mudar a animação da jogadora, passando a direção do mouse e o estado da jogadora daqui. Então, vamos tirar isso daqui e vamos colar no `atualizar a animação`.

[02:37] Só que a direção do mouse, eu só calculava ela, essa direção que vai ser a direção do mouse, eu só calculava ela quando a minha animação era chamada. Quando eu mudar a direção da animação era chamada. O que eu quero fazer é poder usar a posição do mouse todo frame. Para isso eu vou precisar salvar ela numa propriedade na classe, porque aí eu vou calcular a direção do mouse quando o mouse se mover, e enquanto ele não se mover a direção dele é aquela, então fica salvo para mim.

[03:06] Então, na verdade essa direção do mouse do meu `atualizar animação`, vamos dar um espaço aqui para a gente achar melhor o nosso código, essa direção do mouse vai ser uma propriedade, então eu posso chamar o `this.direçãodomouse`, que a gente pode criar ele aqui em cima. É uma propriedade que só interessa a minha jogadora, então eu vou transformá-la em uma propriedade privada, e ela vai ser um vetor, então `cc.vec2`, é um vetor.

[03:34] Eu vou inicializar ela, porque a gente sempre precisa inicializar as nossas propriedades, é uma boa prática a gente fazer isso, porque é sempre um problema você ter uma propriedade não inicializada, porque às vezes você quer

usar ela e ela não está inicializada, então ela vai te retornar um nulo e você vai ter um erro no seu jogo, no seu código, simplesmente porque você não veio e inicializou a sua propriedade. Então, `this.direçãoomouse=cc.vec2.up`, essa constante né, que é um vetor com $y = 1$ ou $x = 0$.

[04:16] Então, a minha direção do mouse já está inicializada, e quando que eu vou calcular essa direção do mouse? Bom, eu vou calcular a direção do mouse no mouse move, e qual que é a função do meu método, dessa classe da jogadora que faz isso? O meu método é o calcular direção do mouse, é esse método que faz isso. Então, no meu evento de mousemove, eu vou chamar o calcular a direção do mouse. O calcular a direção do mouse vai alterar essa direção do mouse, essa propriedade que a gente criou.

[04:43] Então, eu vou lá no meu calcular a direção do mouse, em vez de retornar a direção, eu vou falar que o `this.direçãoomouse` mexe vai ser igual a minha direção, então eu não vou retornar uma direção, eu vou atribuir o cálculo que esse método fez à minha direção do mouse. Quem que usava essa calcular a direção do mouse? Bom, o meu atirar usava. Então, minha direção, em vez de ser o calcular a direção do mouse, vai ser agora o direção do mouse a nossa propriedade. Aí eu não preciso mais chamar esse método através do método atirar.

[05:17] Mais alguém usava esse método? Vamos ver. Eu tenho o mudar animação, esse mudar animação a gente não está usando mais. O que mais? Então, ninguém. Então, agora o meu mouse já está sendo calculado, ele já está sendo guardado em uma propriedade na classe. A segunda coisa que a gente precisava arrumar aqui é o estado, e como a gente ajustava, a gente viu qual era o estado do jogador? Dentro do próprio mudar animação, mudar a direção da animação.

[05:42] Então, se a gente em vez de usar esse método, esse mudar a direção da animação, para realmente mudar a animação, para atualizar ela, a gente pode chamar ele de estado atual. Sempre que a gente chama esse estado atual, eu vou retornar, então eu vou dar um return no estado, que está sendo calculado aqui dentro. Então, a gente não tem mais a direção do mouse, a gente não precisa mais calcular aquilo, a gente já fez isso com uma propriedade e a gente tem agora o método estado atual, que vai ser chamado aqui, então `this.estadoatual` e esse é um método então eu preciso por parênteses.

[06:18] E aí todo frame eu vou chamar o atualiza animação, mudando a animação para direção do mouse que eu tenho, toda vez que ele se mexe a gente vai atualizar isso, e para o estado da jogadora. Então, quando apertar alguma tecla do teclado, quando ela começar a andar, o meu estado vai ser andando, quando ela parar o meu estado vai ser parado, e isso vai ser atualizado todo frame.

[06:44] Então, agora a animação da nossa jogadora, não depende mais só do mouse, ela vai ser visualizado todo frame. Aí vem uma tela preta, então a gente deve ter algum erro. Eu vou clicar no pause aqui em cima, onde ele para o update da cocus, para a gente conseguir ver melhor o console, e ele está falando que ele não consegue achar o X, no controle de animação 16. Esse erro está vindo do mudar a animação, do controle de animação, mudar animação, no jogador atualiza animação, então ele tem toda pilha de chamadas para a gente conseguir ver.

[07:23] Como a gente mexeu no jogador, eu chuto que o nosso erro está aqui dentro do jogador e não no controle de animação. Então, no jogador linha 60, a gente está tendo erro. Então, vamo lá, jogador linha 60, onde eu estou chamando controle de animação, muda a animação com direção no mouse, o direção no mouse é uma propriedade privada. Então, eu preciso do underline, a gente colocou ele como underline aqui em cima, vamos voltar lá e vamos dar um play de novo não, vamos fechar a aba anterior.

[07:52] E agora sim, a nossa jogadora ela começou já parada e meu o mouse está parado, mas eu começo a andar e ela começa a andar, eu paro, ela para. Então, a nossa jogadora, a animação dela está muito mais condizente com o estado do jogo, que é muito melhor, dá uma sensação melhor para o nosso jogador.

