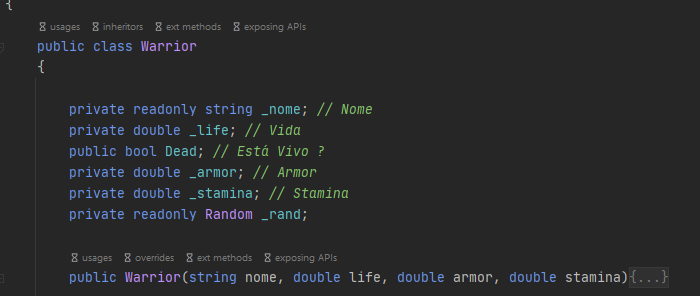
**Relatório POO**

Programação de Sistemas Informáticos - Professor Hugo Rosas  
Henrique Melo

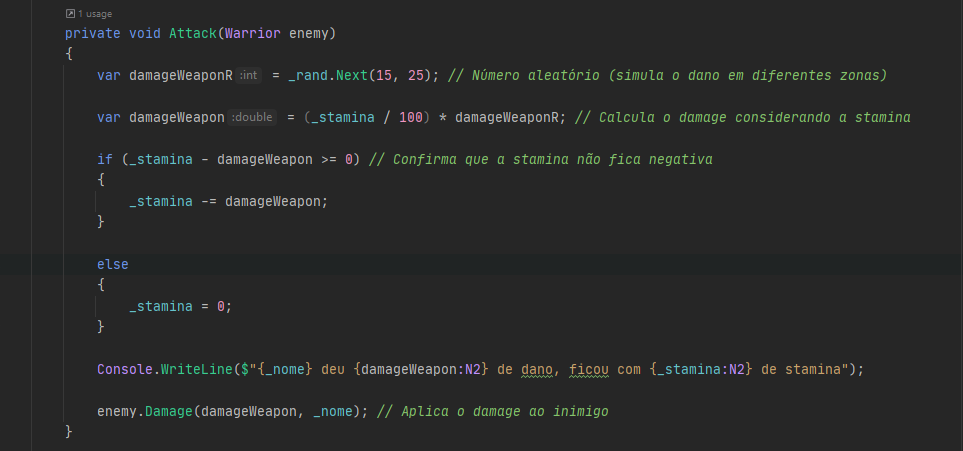
Neste projeto, foi proposto a criação de um jogo no 'cmd' em que dois jogadores combatessem entre si aleatóriamente e no final um dos jogadores ganhasse e o outro perdesse.

Para tornar o jogo mais interessante eu decidi que cada jogador teria para além da sua vida, stamina e armadura.

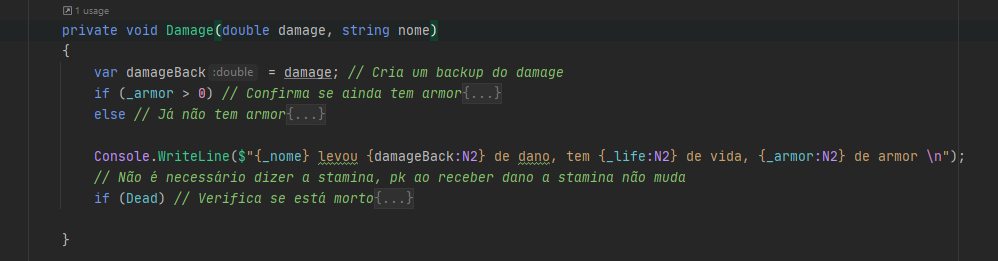


Na imagem acima temos a Class Warrior onde estão declaradas as variáveis '\_nome', que vai ser a variável que guarda o nome do jogador, '\_life', que vai guardar a vida do jogador, 'Dead', esta variável controla o fim de jogo, ou seja se esta variável for 'true' quer dizer que um dos jogadores morreu e o jogo acabou, 'armor', guarda o valor da armor do jogador e 'stamina' guarda o valor da stamina.

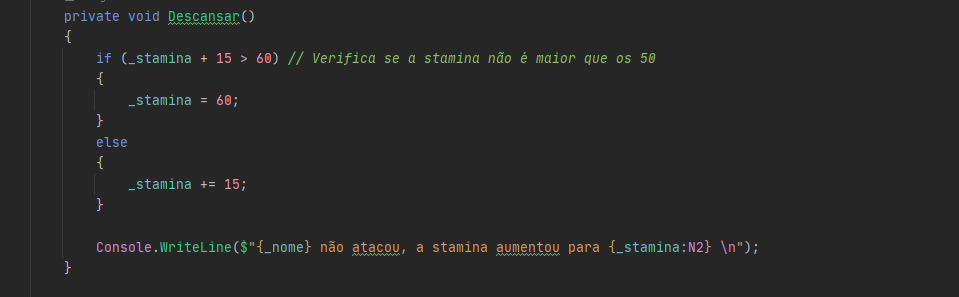
Depois das variáveis serem declaradas e serem atribuídos os seus valores iniciais, é necessário haver interação entre os dois jogadores, para isso a Class Warrior tem alguns métodos.



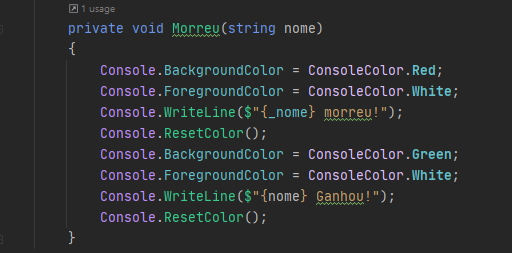
O método Attack, na minha opinião é bastante interessante, pois ele calcula um valor random entre 15 e 25, que vai simular as diferentes partes do corpo em que o jogador pode atacar, depois calcula o dano que vai dar ao inimigo, através de um cálculo que tem em conta a stamina do jogador, desta maneira quanto maior a stamina maior o dano.



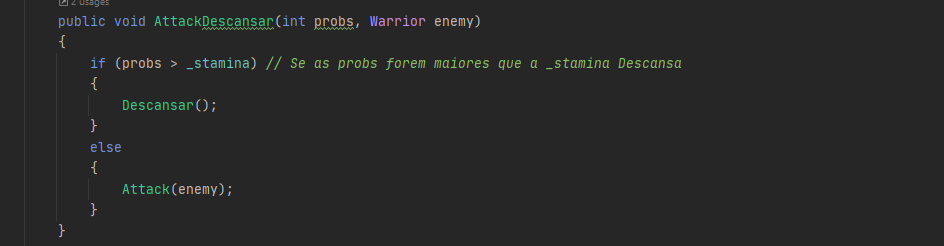
O método Attack acaba por invocar o método Damage, este é o método que se certifica de aplicar o dano ao inimigo, se o jogador ainda tiver armadura o dano é retirado à armadura e assim que a armadura acabar o dano é aplicado na vida do jogador.



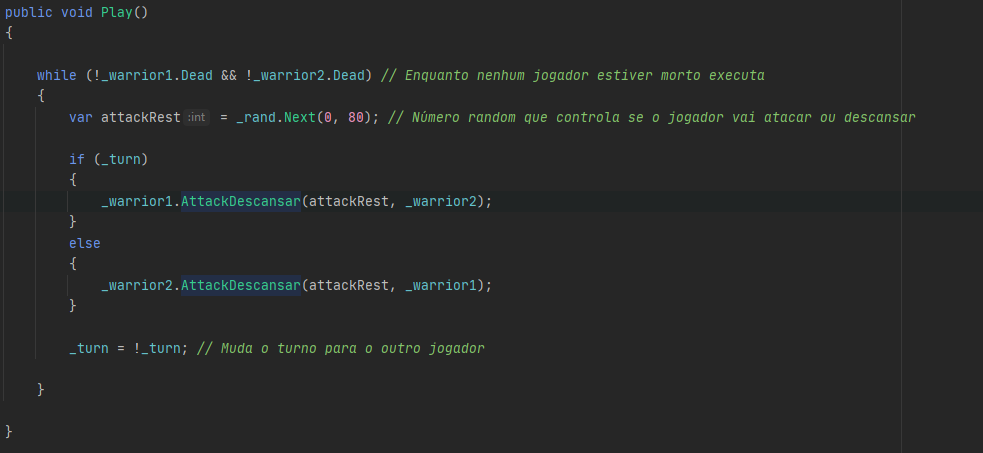
Como os ataques dependem da stamina o jogador tem a opção de descansar para poder restaurar alguma stamina no turno, abdicando de atacar.



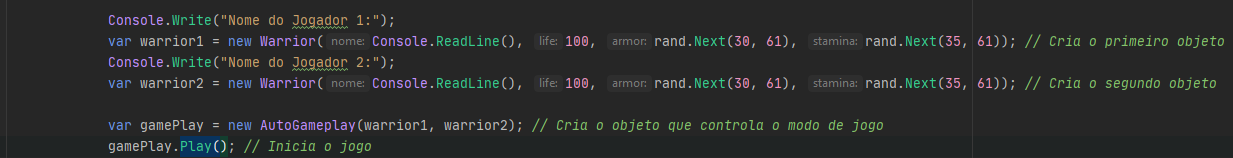
O método Morreu, é invocado quando um dos jogadores morre e apenas escreve no 'cmd' que jogador morreu e que jogador ganhou.



Cada jogador tem o seu turno e pode escolher atacar ou descansar, o método acima é o método responsável por essa escolha e por invocar o método respetivo, este método recebe uma variável com o nome probs, se o valor desta variável for maior que a stamina do jogador, o jogador escolhe descansar, se não, o jogador vai atacar o inimigo.



Agora já sabemos quais as possibilidades de cada jogador, mas para isso é necessário que o jogo comece, é por isso que o método Play existe dentro da Class AutoGameplay, este método vai verificar de quem é que é a vez de jogar e vai chamar o método AttackDescansar do respetivo jogador, passando as probs, que são um valor gerado automáticamente entre 0 e 80 que vão ajudar a determinar no método do jogador se deverá atacar ou descansar.



Antes do jogo começar é necessário que os jogadores sejam criados e que o próprio jogo comece, por isso assim que o programa executa, cada jogador é criado com um nome à escolha pelo utilizador, é atribuído o valor da armor e da stamina aleatóriamente e é criado o jogo com cada um dos jogadores.