

Em busca do filho perdido

Alunos: Arthur Felipe Luciani Corrêa e Rafael Antônio Bressanini

Data: 18/05/2021

Resumo:

O jogador controlará Rollo em uma jornada para resgatar seu filho sequestrado, nessa grande missão Rollo passará cidades, receberá auxílio e fará grandes amigos, porém terá que passar por imprevistos. O jogador guiará Rollo e suas decisões podem mudar o destino do jogo, com situações diferentes.

Descrição do jogo

Rollo e seu filho estavam fazendo uma viagem para cidade de Edda quando no caminho foram surpreendidos por três bandidos, que os assaltaram e sequestram o filho de Rollo. Então Rollo começa uma jornada em busca do filho, essa jornada será guiada pelo jogador, as decisões tomadas pelo jogador influenciam no jogo, e podem melhorar ou piorar as chances de sucesso.

O jogador precisará trabalhar para conseguir suas moedas, e conseguir comprar ferramentas e armas para enfrentar os perigos do jogo.

O jogo foi dividido em cidades (Edda, Dagr, Forseti, Harald, Ingvar, Siefried) e momentos marcantes como emboscadas, para uma melhor organização. Em cada cidade o jogador terá tarefas diferentes e poderá optar por fazê-las ou não, além de descobrir pistas para jornada em resgate do seu filho, algumas cidades podem ser melhores que outras. O jogo também conta com algumas emboscadas, nas quais o jogador só sobreviverá se tiver os equipamentos necessários para superar as dificuldades.

Desenvolvemos um sistema de inventário que o jogador poderá acessar em determinados momentos do jogo, no inventário o jogador poderá ver seus equipamentos e informações sobre eles, além de verificar sua quantidade de moedas. Os itens armazenados no inventário poderão ser comprados ou adquiridos em determinadas ocasiões, e são fundamentais para o jogador continuar sua jornada, uma vez que os itens necessários, perderá o jogo.

```
>Aceitar ferramentas?
[1]Sim
[2]Nao
1
Rollo: Ficarei muito grato se voce fizer isso.
Ferreiro: Nao precisa agradecer, e boa sorte com seu filho.

    >Inventário
Você pode consultar seu inventário para saber todos os itens que você tem
    >Fazer outra missão
Você pode fazer a outra missão disponível, no seu caso a missão 2
    >Continuar
Você continua a história do jogo independente de ter feito todas as missões ou não

    [0]Consultar Inventário
    [1]Fazer outra missão
    [2]Continuar
0

>Inventário

>Moedas: 0

>Espada nivel 1
    Dano: 40

>Armadura nivel 1
    Proteção: 50

>Item não encontrado

>Item não encontrado

>Item não encontrado

>Item não encontrado

>Item não encontrado
```

Descrição do código

O jogo segue uma ordem dentro de uma função geral (int main), essa ordem são as cidades e imprevistos que jogador superará. Quase todas as cidades estão estruturadas em um loop While, que faz a função de deixar o jogador preso para fazer as missões da cidades, até selecionar a opção de continuar, e continuar sua jornada para a próxima cidade. As missões ficam dentro de um laço Switch Case e são acessadas a partir da escolha da missão

Ex loop While, para continuar ou fazer outra missão:

```
[1]Fazer outra missão
[2]Continuar
```

Ex Switch Case:

```
[1] Falar com o ferreiro
[2] Falar com o vendedor de alimentos
```

Para melhor entendimento da história algumas cidades terão ordem de missões definidas e o jogador não poderá selecionar qual missão deseja fazer antes.

Funções

funcaoContinuarOuParar: Esta função serve para apresentar as opções de visitar o inventário, fazer outra missão ou continuar seu caminho.

Ex:

```
>Inventário
Você pode consultar seu inventário para saber todos os itens que você tem
>Fazer outra missão
Você pode fazer a outra missão disponível, no seu caso a missão 2
>Continuar
Você continua a história do jogo independente de ter feito todas as missões ou não

[0]Consultar Inventário
[1]Fazer outra missão
[2]Continuar

0

>Inventário

>Moedas: 0

>Espada nível 1
    Dano: 40

>Armadura nível 1
    Proteção: 50

>Item não encontrado

>Item não encontrado

>Item não encontrado

>Item não encontrado

>Item não encontrado

[1]Fazer outra missão
[2]Continuar
```

verificacao: Verifica se a opção digitada pelo jogador é válida, por exemplo, se as opções são 1 e 2, enquanto o jogador não digitar uma das duas opções o jogo não avançará.

Ex:

```
[1]Falar com o ferreiro
[2]Falar com o vendedor de alimentos
8

[1]Falar com o ferreiro
[2]Falar com o vendedor de alimentos
7

[1]Falar com o ferreiro
[2]Falar com o vendedor de alimentos
1
Rollo: Olá amigo você tem um minuto para me ajudar?
```

criaAsOpcoes: Verifica os itens que o jogador possui e partir daí cria as opções, e faz a verificação se o número digitado está correto

Ex:

[1]Lutar	Opção bloqueada(você não tem os itens necessários)
[2]Fugir com o cavalo	[2]Fugir com o cavalo

inventário: Mostra os itens do jogador e informações adicionais, como o dano de uma espada, ou proteção de uma armadura. Se o jogador não tiver algum item aparece item não encontrado.

Ex:

```
>Inventário
>Moedas: 25
>Espada nivel 1
    Dano: 40
>Armadura nivel 1
    Proteção: 50
>Item não encontrado
Armadilha(só pode ser usada em momentos especiais)
>Item não encontrado
>Sela para o cavalo
>Pães: 2
```

Dificuldades

Uma grande dificuldade que encontramos foi para criar a função `criaAsOpcoes` pois não sabíamos como poderíamos fazer o jogo mostrar as opções que o jogador poderia escolher a partir dos itens coletados pelo jogador além de também precisar fazer a verificação se o número digitado era aceitável. Outra grande dificuldade foi para descobrir como faríamos a estruturação das cidades e como colocar cada missão eventos dentro delas.

Mas a maior dificuldade foi quando ficamos completamente presos em uma parte do código, pois na hora de encaixar os códigos juntos não funcionavam, mesmo tendo revisado todos os erros possíveis, o que nos deixou bastante desanimados. Porém depois de reescrever uma parte do código tudo voltou a funcionar.