



## Dark Continent

A aventura pelo continente escuro

## Requisitos:

### Universo:

Em um universo 2D a aventura começa, nesse jogo você é um herói arqueólogo, que está em busca de alguns itens perdidos, que você descobriu durante um estudo. O universo apresenta as classes e o menu de opções para o jogador. O universo contém 2 mapas (floresta e ruínas), então fundo das batalhas e exploração muda durante a jogatina.

### Jogador:

O jogador possui nome, nível, status, itens, inventário, recursos (nesse mundo ouro se chama greedies), EXP, vida, mana e classe. Recursos ajudam o jogador a comprar novos itens, seja armas, armadura, poções ou itens-chaves para missões. É possível obter recursos de duas formas, através de exploração do mapa, alguns locais têm itens e greedies e também pelas batalhas. Ao vencer batalhas o jogador ganha greedies e EXP. Ao perder o jogador perde greedies e EXP, nunca podendo cair de nível. Um novo jogador começa nível 1 e com nível normal de recursos e itens. O jogador deve estar presente no universo, explorando os mapas, ou em batalhas, se sua vida chegar a 0 o jogo te dá duas opções reiniciar do último ponto em que o jogo foi salvo ou desistir, ao escolher desistir o jogo termina e elimina o jogador do universo. Durante a parte de exploração é possível realizar as seguintes ações: mover jogador, abrir baús, pular, usar itens, entrar em combate (encostando em um inimigo), ao estar em combate o jogador pode realizar as seguintes ações: Atacar alvo, usar habilidades, usar itens, fugir. O jogador pode fugir a qualquer momento. O jogador deve eliminar todos os inimigos para sair vitorioso.

### Status:

Nesse jogo possui um total de 5 status que o jogador pode aumentar de acordo com o seu nível, cada nível se ganha 3 pontos, porém a cada 5 níveis se ganha 5 pontos. Os 5 status são: Força, Agilidade, Inteligência, Vitalidade e Sorte.

Força aumenta seu dano com itens corpo-a-corpo da classe de guerreiro em x dano a cada nível e aumenta um pouco a quantidade de vida.

Agilidade aumenta o dano de itens longo alcance e adagas em x dano e aumenta a "velocidade", o jogador com maior agilidade tem o primeiro turno.

Inteligência aumenta o dano de armas e magias que dão dano mágico em x dano e aumenta a quantidade de mana.

Vitalidade aumenta a bastante quantidade de vida do jogador em um total de x e reduz um pouco o dano tomado.

Sorte aumenta a chance de causar um acerto crítico (ataque causa o dobro de dano) em x%.

### Classes:

Existem nesse universo 3 classes: Guerreiros, Arqueiros e Magos. Guerreiros são a classe das armas de curto alcance corpo-a-corpo como lanças, clavas, machados e espadas e armaduras pesadas, tem como atributos principais a força e vitalidade, tem 10 de vida base. Arqueiros são a classe das armas de longo alcance (bestas, arcos, aljvas) e adagas e armaduras médias e leves, tem como atributos principais

agilidade e sorte, tem 8 de vida base. Magos são a classe das armas mágicas (cajados, varinhas e orbes) e magias e armaduras leves, tem como atributos principais Inteligência e sorte, tem 8 de vida base.

O jogador é formado por uma imagem em pixel-art de acordo com sua classe, o jogador possui um ponto central que diz a sua posição no mapa e um vetor de posição que condiz com o lado que está andando. Quando um jogador for abatido a imagem saltará um pouco e ficará deitada. A animação de ataque depende da sua arma, porém será única para toda uma classe de armas, Exemplo: Armas corpo-a-corpo o jogador andar até o alvo e fará uma animação de corte ou batida, armas de longo alcance andar até a frente do alvo e atirá uma flecha, magias o jogador andar até a frente do alvo conjurará a magia, se for fogo, aparecerá fogo no pé do inimigo, etc.

O jogador que escolher guerreiro começará com uma espada que dá 2 de força e tem 4 de dano base, um escudo que dá 1 de vitalidade e uma armadura pesada que dá 2 de vitalidade. O jogador que escolher Arqueiro começará com um Arco que dá 2 de agilidade e tem 4 de dano base e uma aljava que dá 1 de sorte e uma armadura leve que dá 1 de agilidade e 1 de sorte. O jogador que escolher mago começará com um cajado que dá 2 de inteligência e tem 4 de dano base e uma armadura leve que dá 2 de inteligência e 1 de sorte.

#### Combate:

O combate do jogo é baseado em turnos quem tem o primeiro turno é quem tiver a maior quantidade de agilidade, seja jogador ou um inimigo, após isso prossegue para o segundo com maior agilidade e assim por diante até o com menor agilidade.

O sistema de dano a princípio será:

$((\text{dano base da arma} + (0.5 * \text{atributo principal da classe(atacante)})) - (0.2 * (\text{Vitalidade(alvo)})))$ .

Todos do universo que podem entrar em combate tem uma taxa base de acerto crítico de 5%, acertos críticos fazem o ataque causar o dobro de dano tornando a fórmula assim:

$(2 * (\text{dano base da arma} + (0.5 * \text{atributo principal da classe(atacante)})) - (0.2 * (\text{Vitalidade(alvo)})))$ .

O combate idealmente será realizado por uma interface, porém caso não consiga fazer será utilizado o próprio console por comandos. Exemplos de comandos para o console:

```
#attackA1 (ataca o adversário na posição 1);  
#hability1 (usa a habilidade da posição 1);  
#useitem1 (usa o item da posição 1);  
#run (foge).
```

Ao final da batalha, o registro de todas as ações será gravado em um arquivo, para consulta posterior. Se um jogador não digita corretamente seus comandos, a tela é renovada para que o mesmo repita a operação, respeitando o seu tempo limite para ação.

Banco de dados:

O jogo possuirá um banco de dados que armazenará as estatísticas dos jogadores e armas. Sempre que uma nova batalha se inicia, é possível carregar um jogador. O universo decidirá com quantos adversários lutará, podendo ir de 1 até 3, no começo só pode enfrentar até 2. O banco também armazenará os inimigos, que são únicos.

## Lista de substantivos e verbos:

Substantivos	Atribuição	Checagem
Universo	Classe	NA
Jogador (Player)	Classe	NA
Classes	Classe	NA
Menu	Verbalizado	NA
Cenário e jogador	imagens	NA
Nome (Jogador)	Atributo do jogador	NA
Classe (Jogador)	Atributo do jogador	NA
Nível (Jogador)	Atributo do jogador	NA
Status (Jogador)	Atributo do jogador	NA
Itens (Jogador)	Atributo do jogador	NA
Inventário (Jogador)	Atributo do jogador	NA
Recursos (Jogador)	Atributo do jogador	NA
EXP (Jogador)	Atributo do jogador	NA
Vida (Jogador)	Atributo do jogador	NA
Mana (Jogador)	Atributo do jogador	NA
Batalha	Classe	NA
Cenário	Imagens	NA
Ordem dos turnos (Agilidade)	Atributo	NA
Dano	Atributo	NA
Imagem (Jogador)	Classe	NA
Vetor (Jogador)	Atributo	NA
Bando de dados: Batalha	Classe	NA
Quantidade de inimigos	Atributo	NA
Linha de comando (Terminal)	Classe	NA
Ação	Verbalizado	NA

\*\*\*

Verbos	Atribuição	Checagem
Universo mostra as opções (menu)	Membro	NA
Universo inicia a batalha (batalha)	Membro	NA

Universo apresenta os inimigos (view)	Membro	NA
Jogador compra itens (compra)	Membro	NA
Jogador adquire recursos (ganha) - Update()	Membro	NA
Jogador perde recursos (perde) - Update()	Membro	NA
Criar jogador	Membro	NA
Matar jogador	Membro	NA
Mover jogador	Membro	NA
Atacar	Membro	NA
Fugir (run)	Membro	NA
Play (ação de jogador)	Membro	NA
Posicionar inimigos	Membro	NA
ChatAction	Membro	NA
Save	Membro	NA

## Detalhamento de requisitos e identificação de Classes:

1. Os Players receberam uma ação de morrer. Através desta ação os mesmos modificarão o status "alive", que permitirá a sua permanência no universo.
2. O jogador recebeu um vetor de referência.
3. Vetores, são entidades matemáticas representadas por uma origem (ponto) e um ângulo.
4. Os Pontos são entidades matemáticas representadas por duas coordenadas polares: ângulo e raio.
5. As entidades possuirão uma função de colisão, que vai iniciar um combate em caso de colisão com jogador.
6. O atributo "combat" é uma variável que indicará se o jogador está ou não em combate.
7. O atributo "damage" é a variável que armazenará a força do ataque das entidades. Esta variável deverá ser ajustada sempre que o Player escolher atacar ou trocar de arma.
8. O membro "print" fará a impressão das informações de classe no terminal. O membro "show" fará a plotagem da entidade no espaço.
9. O membro "rotate" realizará a rotação do Vetor em questão.
10. A classe "Guerreiro" foi criada.
11. A classe "Arqueiro" foi criada.
12. A classe "Mago" foi criada.
13. Em cada batalha, será registrada a quantidade de inimigos, a quantidade de rodadas, e as ações dos jogadores. Após as batalhas, o jogo deverá gravar um arquivo para conferência posterior.
14. As ações dos jogadores serão tratadas antes de serem validadas, por uma operação denominada chatAction(). Em cada batalha será registrado o apelido do vencedor (Red Eagle, Blue Lizard, etc).
15. As regras físicas do jogo pertencem ao Universo. Os jogadores realizarão suas ações e o Universo tratará de validar as ações e suas consequências. Propriedades tais como colisões, atritos envolvidos, dentre outras que se

fizerem necessárias. Ou seja, o Universo realizará as alterações no exército de cada jogador.

16. O Universo possui uma Tela (Screen), que será responsável pela apresentação dos elementos gráficos que estiverem habilitados.
17. Os jogadores possuem um status de jogo, para o Universo saber se um jogador está na batalha ou se o mesmo fugiu, ou ainda se morreu.
18. O Universo possui um Terminal (Terminal) para receber as ações dos jogadores, apresentá-las aos mesmos, analisar as ações e executá-las no Universo.
19. A Tela (Screen) também é responsável pelos áudios do jogo: trilha sonora, tema da batalha, efeitos sonoros, movimento e fuga.
20. As ações serão representadas por um modelo chamado Action. Cada modelo terá sua Action que, no final da batalha, será registrada no banco de dados.

## VIEWERS

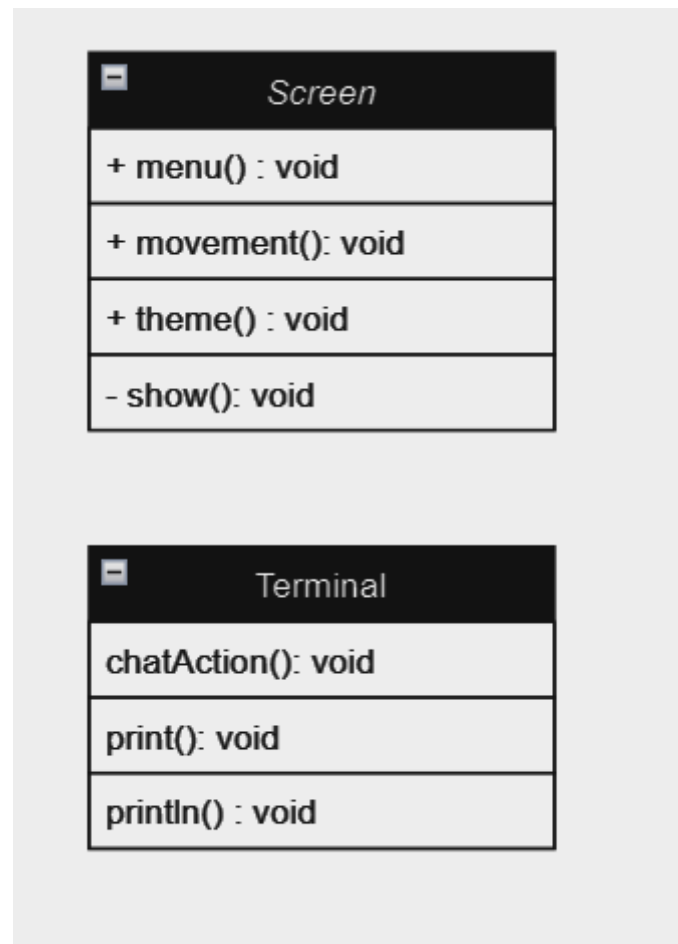


Figura 1 – MODELS VIEW

## CONTROLLERS

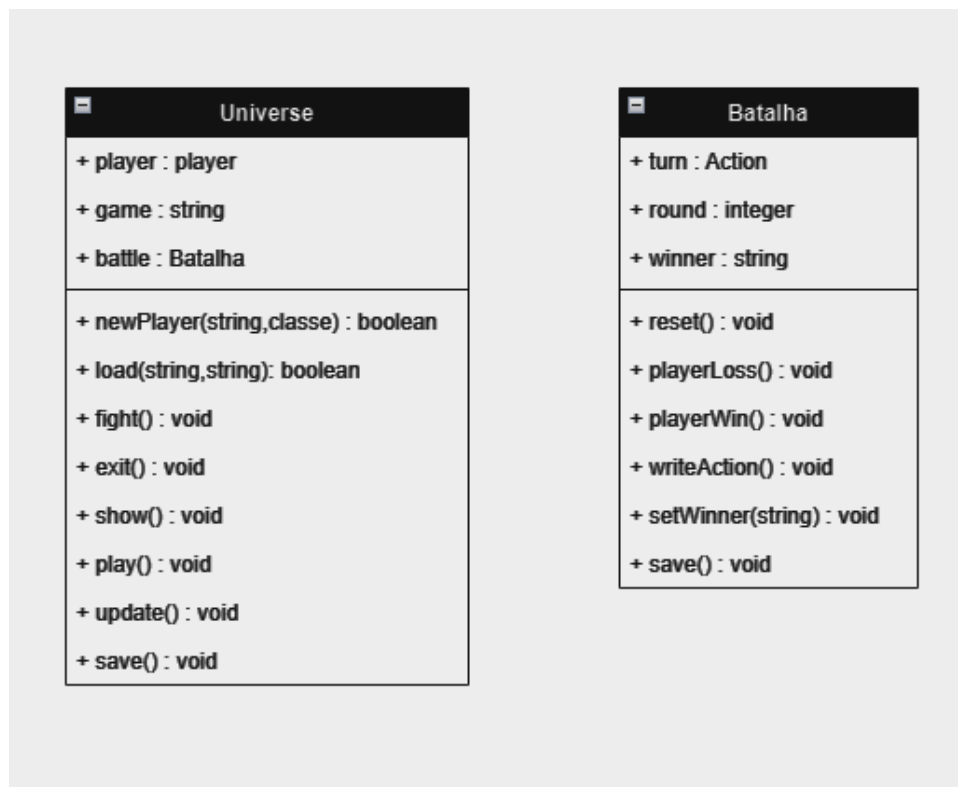


Figura 2 – CONTROLLERS

## MODELS

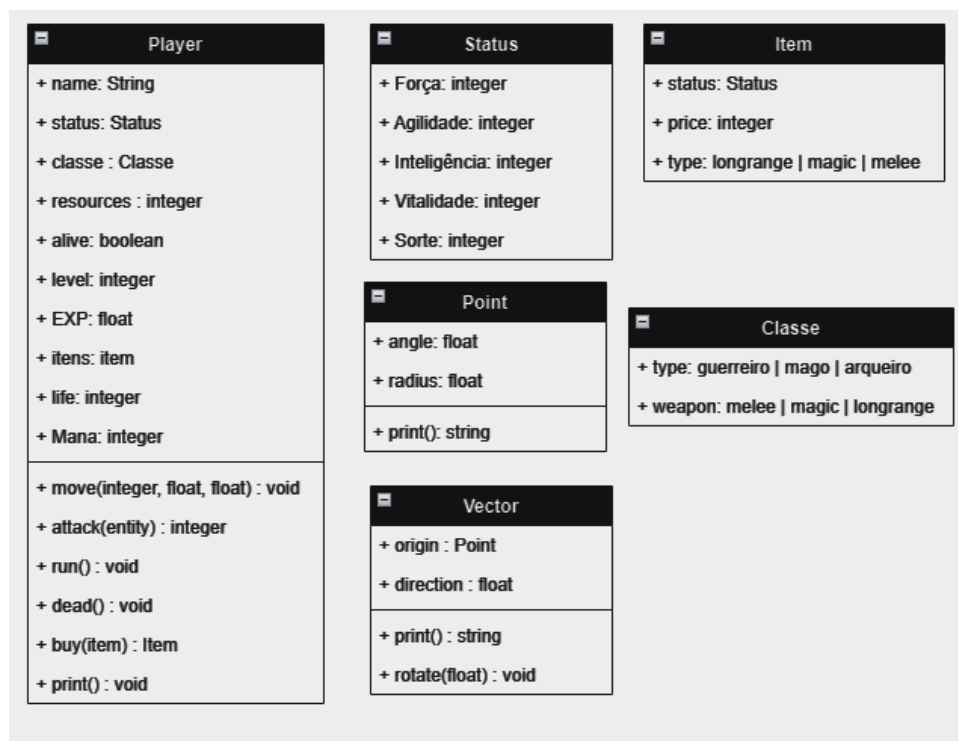


Figura 3 – MODELS

## DIAGRAMA DE CLASSES

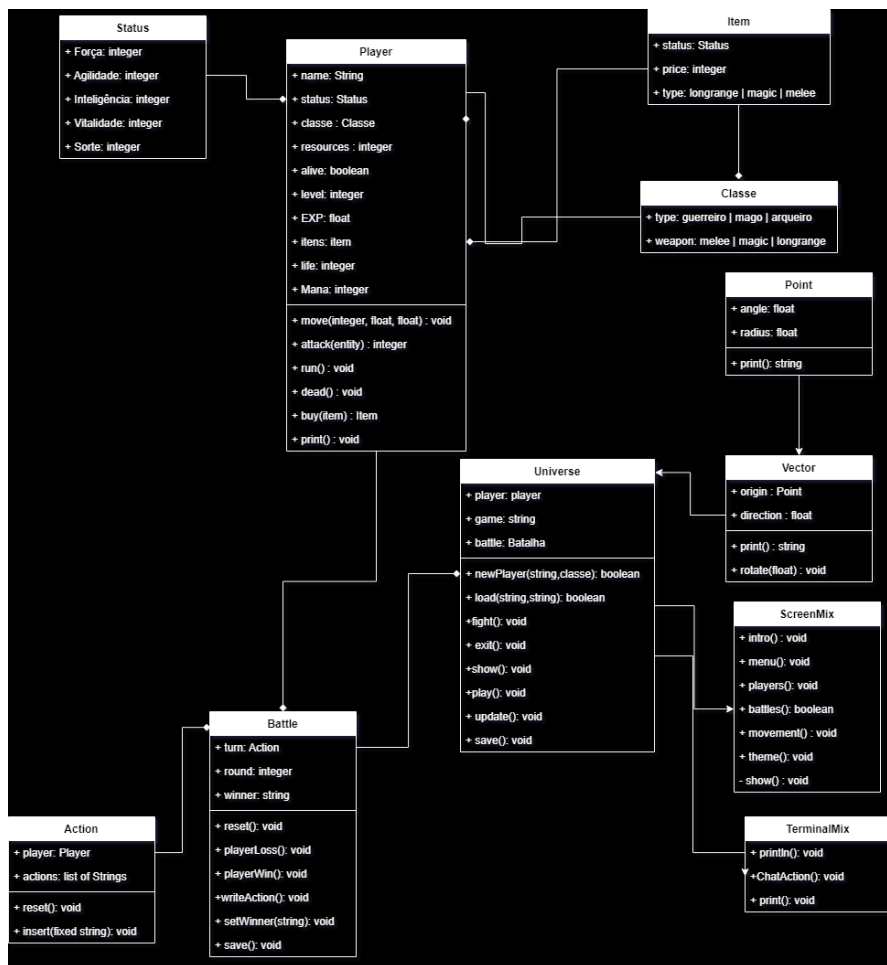


Figura 4 – Diagrama de Classes

## TABELA DE CONSTRUTORES

Classe	Construtor
Universe	vazio
Player	(nome,classe)
Battle	(Action,Player)
Action	Vazio
Terminal	Mixin
Screen	Mixin
Classe	(type,weapon)
Item	(status,price,type)
Vector	(Point,Point)

## DIAGRAMA DE CASO DE USO



