**Instituto Superior de Educação Aldo Muylaert.**

**Técnico de Informática**

Desyrê do Nascimento Ananias e Maria Clara S. De Sousa

**MODELAGEM DE DADOS II**

(Jogo RPG)

**Campos Dos Goytacazes**

**2025**

Sumário

[1. Apresentação das Classes 3](#_Toc210676678)

[1.1 Classe Jogador 3](#_Toc210676679)

[1.2 Classe Personagem 4](#_Toc210676680)

[1.3 Classe Guardião\_do\_Tempo (herda de Personagem) 5](#_Toc210676681)

[1.4 Classe Guerreiro (herda de Personagem) 5](#_Toc210676682)

[1.5 Classe Cavalheiro (herda de Personagem) 5](#_Toc210676683)

[1.6 Classe Inimigo (herda de Personagem) 6](#_Toc210676684)

[1.7 Classe Monstro (herda de Inimigo) 6](#_Toc210676685)

[1.8 Classe Sombrio (herda de Inimigo) 6](#_Toc210676686)

[1.9 Classe Bruxa (herda de Inimigo) 7](#_Toc210676687)

[1.10 Classe Habilidade 7](#_Toc210676688)

[1.11 Classe Local 8](#_Toc210676689)

[1.12 Classe Castelo (herda de Local) 8](#_Toc210676690)

[1.13Classe Floresta (herda de Local) 9](#_Toc210676691)

[1.14Classe Comunidade (herda de Local) 9](#_Toc210676692)

[1.15 Classe Torre (herda de Local) 9](#_Toc210676693)

[1.16 Classe Templo (herda de Local) 9](#_Toc210676694)

[1.17 Classe Caverna (herda de Local) 10](#_Toc210676695)

[1.18 Classe Item 10](#_Toc210676696)

[1.19 Classe Arma (herda de Item) 10](#_Toc210676697)

[1.20 Classe Armadura (herda de Item) 11](#_Toc210676698)

[1.21 Classe Poção (herda de Item) 11](#_Toc210676699)

[1.22 Classe Livro (herda de Item) 11](#_Toc210676700)

[1.23 Classe NPCs 11](#_Toc210676701)

[1.24 Classe Missão 12](#_Toc210676702)

[1.25 Classe Inventário 12](#_Toc210676703)

[1.26 Classe Loja 13](#_Toc210676704)

[1.27 Classe Mestre 13](#_Toc210676705)

[1. Diagrama de Caso de Uso 14](#_Toc210676706)

[2. Caso de Uso Estendido 14](#_Toc210676707)

[3. Requisitos Funcionais e Não Funcionais 14](#_Toc210676708)

[4. Diagrama de classe 14](#_Toc210676709)

[5. Diagrama de Sequências 14](#_Toc210676710)

### 

### Apresentação das Classes

# 1.1 Classe Jogador

Representa o usuário humano que participa do jogo. É responsável por controlar personagens, salvar progresso e interagir com o sistema.

Atributos:

id\_jogador: Identificador único do jogador.

nome\_usuario: Nome de login utilizado no sistema.

email: Endereço de contato do jogador.

senha: Código usado para autenticação e segurança.

Métodos:

criar\_personagem(): Cria um novo personagem jogável.

selecionar\_personagem(personagem): Permite escolher um personagem existente.

iniciar\_jogo(): Inicia uma nova sessão de jogo.

salvar\_progresso(): Armazena o progresso atual.

carregar\_progresso(): Recupera um estado salvo do jogo.

encerrar\_jogo(): Finaliza a partida atual.

visualizar\_perfil(): Exibe informações do jogador.

trocar\_senha(): Permite alterar a senha de login.

desconectar(): Finaliza a sessão ativa.

# 1.2 Classe Personagem

Classe base que representa qualquer personagem do jogo, controlado ou não pelo jogador. Define atributos e comportamentos comuns a todos.

Atributos:

Nome: Identificação do personagem.

Nível: Mede experiência e poder.

Força: Capacidade física e dano em combate.

Pontos\_de\_Vida: Quantidade de energia vital.

Ponto\_Fraco: Vulnerabilidade do personagem.

Item: Objeto carregado ou em uso.

Habilidade: Poderes e capacidades especiais.

Posição: Localização atual no cenário.

Métodos:

Atacar\_outro\_personagem(alvo): Executa ataque contra outro personagem.

Defender\_ataque(): Reduz ou evita dano recebido.

Usar\_item(item): Utiliza um item do inventário.

Escolher\_personagem(): Permite selecionar o personagem.

Adicionar\_nome(novo\_nome): Modifica o nome do personagem.

Adicionar\_habilidade(): Acrescenta nova habilidade.

# 1.4 Classe Guerreiro (herda de Personagem)

Personagem focado em força física e combate corpo a corpo.

Atributos:

Força: Representa o poder físico.

Defesa: Resistência contra ataques.

Arma: Equipamento principal usado em batalha.

Nível: Grau de evolução do guerreiro.

Métodos:

Atacar(): Executa um golpe físico contra o inimigo.

Defender(): Reduz o dano recebido.

Treinar(): Aumenta seus atributos.

UsarHabilidade(): Ativa uma habilidade especial de combate.

# 1.5 Classe **Mago** (herda de Personagem)

### 

Variante do guerreiro que representa honra e defesa, geralmente com foco em proteger aliados.

**Atributos:**

**CódigoDeHonra:** Regras que o personagem segue.

**Montaria:** Animal ou veículo usado em batalha.

**Lealdade:** Nível de compromisso com o grupo.

**DefesaExtra:** Valor adicional de proteção.

**Métodos:**

**ProtegerAliado():** Reduz o dano que um aliado sofreria.

**Montar():** Sobe em sua montaria.

**Desmontar():** Desce da montaria.

**InspirarAliados():** Concede bônus temporário aos companheiros.

# 1.6 Classe Inimigo (herda de Personagem)

Entidade hostil que se opõe aos jogadores e aliados.

Atributos:

Tipo\_inimigo: Categoria ou espécie.

Loot: Itens obtidos após derrotá-lo.

Ataque: Tipo de golpe ou comportamento ofensivo.

Fraqueza: Vulnerabilidade.

Métodos:

Atacar\_outro\_personagem(alvo): Executa ataque contra um personagem.

Defender\_de\_ataque(): Protege-se de um golpe recebido.

# 1.7 Classe Monstro (herda de Inimigo)

Criatura selvagem, geralmente irracional e poderosa.

Atributos:

Fúria: Aumenta o dano quando irritado.

Natureza\_em\_grupo: Melhor desempenho junto a outros monstros.

Métodos:

rugir(): Intimida inimigos próximos.

atacar\_em\_grupo(): Coordena ataque coletivo.

# 1.8 Classe Sombrio (herda de Inimigo)

Guerreiro que usa poderes obscuros e furtividade para atacar.

Atributos:

EnergiaSombria: Fonte de poder.

Camuflagem: Capacidade de se esconder.

Velocidade: Agilidade em combate.

Precisão: Capacidade de atingir pontos críticos.

Métodos:

AtacarSombras(): Executa um ataque furtivo.

InvocarTrevas(): Cria uma área escura que enfraquece inimigos.

Fugir(): Sai de uma batalha usando camuflagem.

Corromper(): Aplica efeitos negativos em um oponente.

# 1.9 Classe **Bruxa** (herda de Inimigo)

Personagem que utiliza magia e poções para atacar ou auxiliar.

**Atributos:**

**Mana:** Energia mágica.

**Grimório:** Livro de feitiços.

**Poções:** Itens criados para efeitos diversos.

**Sabedoria:** Conhecimento mágico acumulado.

**Métodos:**

**LançarFeitiço():** Executa um feitiço usando mana.

**CriarPoção():** Produz uma nova poção.

**CurarAliado():** Recupera vida de um companheiro.

**EncantarItem():** Aumenta o poder de um equipamento.

# 1.10 Classe Habilidade

Representa as ações especiais ou poderes que um personagem pode executar no jogo.

Atributos:

Nome: Identificação da habilidade.

Descrição: Explica o que a habilidade faz.

Dano: Quantidade de dano que causa (se for ofensiva).

Custo: Valor necessário (mana, energia etc.) para usá-la.

TempoRecarga: Tempo necessário para reutilizar a habilidade.

Métodos:

Ativar(): Executa o efeito da habilidade.

Melhorar(): Aumenta o poder ou reduz o custo da habilidade.

ExibirDetalhes(): Mostra informações completas sobre a habilidade.

# 1.11 Classe Local

Representa lugares dentro do universo do jogo.

Atributos:

Recursos: Itens e materiais disponíveis.

Clima: Condições ambientais.

Tipo: Categoria do local (castelo, floresta, etc).

Métodos:

Adicionar\_tipo(Novo\_tipo): Define novo tipo de local.

Entrar\_local(personagem): Faz o personagem entrar.

Sair\_local(personagem): Faz o personagem sair.

Explorar\_local(): Procura recursos e segredos.

Interagir\_objeto(objeto): Permite interagir com objetos do ambiente.

# 1.12 Classe Castelo (herda de Local)

Local fortificado, centro de poder e defesa.

Atributos:

Estrutura: Descrição da construção.

Segurança: Nível de proteção.

Cômodos: Quantidade e tipo de salas internas

# 1.13Classe Floresta (herda de Local)

Ambiente natural repleto de vegetação e mistérios.

Atributos:

Períodos: Fases que afetam o local (estações, clima).

Densidade: Grau de cobertura vegetal.

# 1.14Classe Comunidade (herda de Local)

Assentamento populacional com cultura e organização própria.

Atributos:

Tamanho: Extensão ou população.

Cultura: Tradições locais.

Conflitos: Disputas internas.

Mistérios: Segredos do local.

# 1.15 Classe Torre (herda de Local)

Construção vertical com vários andares e desafios.

Atributos:

Níveis: Quantidade de andares.

Possui\_armadilhas: Indica se há perigos.

Métodos:

mudar\_de\_andar(): Move o personagem entre os níveis.

salvar\_armadilhas(): Registra armadilhas existentes.

# 1.16 Classe Templo (herda de Local)

Local sagrado de poder espiritual ou mágico.

Atributos:

Energia\_mística: Fonte de poder.

Sanctuario: Área segura e protegida.

Métodos:

Curar\_personagem(): Recupera vida.

Obter\_bencao(): Concede bênçãos especiais.

# 1.17 Classe Caverna (herda de Local)

Ambiente subterrâneo, escuro e perigoso.

Atributos:

Escuridão: Reduz visibilidade.

Perigos\_ocultos: Presença de armadilhas e criaturas.

Métodos:

Explorar\_caverna(): Investiga o ambiente.

Acender\_tocha(): Aumenta a visibilidade.

# 1.18 Classe Item

Representa objetos utilizáveis ou colecionáveis no jogo.

Atributos:

Qualidade: Estado físico do item.

Raridade: Grau de dificuldade de obtenção.

Valor: Custo em moeda do jogo.

Nome: Identificação textual.

Descrição: Detalhes sobre o item.

Métodos:

Escolher\_nome(Novo\_nome): Renomeia o item.

Usar\_item(Personagem): Aplica o item em um personagem.

# 1.19 Classe Arma (herda de Item)

Item ofensivo usado para causar dano.

Atributos:

Efeitos\_especiais: Propriedades mágicas.

Dano: Quantidade de dano causado.

Alcance: Distância de ataque.

Propriedades: Características (leve, pesada etc).

Métodos:

Usar\_item(personagem): Permite que o personagem utilize a arma.

# 1.20 Classe Armadura (herda de Item)

Descrição:

Equipamento de defesa que reduz dano recebido.

Atributos:

Proteção: Nível de resistência.

Tipo\_armadura: Categoria (leve, pesada etc).

Métodos:

Usar\_item(personagem): Equipa a armadura.

Escolher\_tipo\_armadura(): Define o tipo da armadura.

# 1.21 Classe Poção (herda de Item)

Item consumível que causa efeitos temporários.

Atributos:

Tipo: Efeito da poção (cura, veneno, buff...).

Quantidade: Número de doses disponíveis.

Métodos:

Beber\_pocao(): Usa a poção em si.

Aplicar\_pocao(aliado): Usa em outro personagem.

# 1.22 Classe Livro (herda de Item)

Objeto de aprendizado e desenvolvimento de habilidades.

Atributos:

Título: Nome do livro.

Conhecimento\_contido: Informação ou habilidade ensinada.

Métodos:

Ler(): Lê o conteúdo.

Aprender\_habilidade(): Adquire nova habilidade.

### 1.23 Classe NPCs

Personagens não controlados por jogadores, com papel narrativo.

Atributos:

Nome: Identificação.

Papel: Função na história.

Diálogo: Texto de fala.

Métodos:

Oferecer\_missao(): Propõe tarefas.

Reagir\_ao\_jogador(): Interage com o usuário.

Falhar(): Cancela ou não completa uma ação.

### 1.24 Classe Missão

Objetivos do jogo que rendem recompensas.

Atributos:

Objetivo: Meta a ser cumprida.

Recompensa: Benefício ao concluir.

Status: Estado atual (ativa, concluída, falha).

Métodos:

Aceitar(): Inicia a missão.

Recusar(): Rejeita a missão.

Concluir(): Finaliza com sucesso.

Falhar(): Marca como fracassada.

### 1.25 Classe Inventário

Armazena e organiza os itens do personagem.

Atributos:

Itens: Lista de objetos.

Capacidade\_itens: Limite máximo de armazenamento.

Métodos:

adicionar\_item(): Adiciona um novo item.

remover\_item(): Exclui um item.

listar\_itens(): Mostra conteúdo.

trocar\_unidade\_moedas(): Converte moedas.

remover\_moeda(): Gasta valor.

adicionar\_moeda(): Adiciona moeda ao saldo.

### 1.26 Classe Loja

Sistema de comércio interno do jogo.

Atributos:

Itens\_disponíveis: Lista de produtos à venda.

Moeda: Tipo de moeda usada.

Métodos:

Comprar(): Adquire item.

Vender(): Transfere item em troca de moeda.

Negociar(): Faz barganhas ou trocas especiais.

### 1.27 Classe Mestre

Representa o Mestre do Jogo (Game Master), narrador e controlador do mundo. É responsável por conduzir a história, criar eventos e definir recompensas.

Atributos:

tempoProvisacao: Controle do tempo narrativo.

criatividadeentidade: Capacidade de criar histórias e situações.

Métodos:

criarEvento(tipo): Gera um novo evento.

gerenciarNPCs(npcs): Controla personagens não jogáveis.

salvarMissao(): Armazena informações de missões.

definirRecompensa(missao): Define prêmios.

definirResultado(local): Determina consequências no cenário.

2. Diagrama de Caso de Uso

### D:\Segundo ano\Trabalho interdiciplinar\Diagrama de caso de uso RPG-Página-1.drawio.png

### 3. Layout dos casos de uso:

|  |  |
| --- | --- |
| **Use case** | Gerenciar Sessão |
| **Atores:** | Jogador |
| **Tipos:** | Primário |
| **Propósitos:** | O jogador inicia o jogo |
| **Resumo** | O jogador pode logar, salvar ou carregar seu progresso, controlando o acesso ao sistema. |
| **,Referência cruzada:** | Classes: *Jogador* |

|  |  |
| --- | --- |
| **Sequência típica de eventos** | |
| **Ação do ator** | **Resposta do sistema** |
| 1. Jogador acessa a tela inicial do jogo. | 2. O sistema exibe opções de login e carregamento. |
| 3. Jogador escolhe “Logar Usuário”. | 4. O sistema valida as credenciais. |
| 5. Jogador seleciona “Carregar Progresso”. | 6. O sistema carrega os dados salvos. |
| 7. Jogador inicia o jogo. | 8.O sistema inicia a sessão ativa. |

|  |
| --- |
| **Sequências alternativas:** |
| **Linha 4 :** Se as credenciais estiverem incorretas, o sistema exibe mensagem de erro e solicita nova tentativa. |
| **Linha 6:** se não houver progresso salvo, o sistema pergunta se o jogador deseja começar uma nova partida. |

|  |  |
| --- | --- |
| **Use case** | Gerenciar Personagem |
| **Atores:** | Jogador |
| **Tipos:** | Primária |
| **Propósitos:** | Permitir que o jogador crie, visualize e gerencie seus personagens. |
| **Resumo** | O jogador cria um novo personagem, visualiza seus atributos e atualiza informações. |
| **Referência cruzada:** | Classes: *Personagem, Habilidade, Missão* |

|  |  |
| --- | --- |
| **Sequência típica de eventos** | |
| **Ação do ator** | **Resposta do sistema** |
| 1. Jogador seleciona “Criar Personagem”. | 2. O sistema solicita nome e atributos. |
| 3. Jogador insere as informações. | 4. O sistema cria o personagem e salva. |
| 5. Jogador escolhe “Visualizar Status” | 6. O sistema exibe atributos, nível e habilidades. |

|  |
| --- |
| **Sequências alternativas:** |
| **Linha 1 :** Se o jogador já tiver o número máximo de personagens, o sistema exibe aviso. |
| **Linha 5** : Caso o personagem esteja inativo, o sistema informa que ele não pode ser visualizado |

|  |  |
| --- | --- |
| **Use case** | Explorar o Mundo |
| **Atores:** | Jogador |
| **Tipos:** | Primário |
| **Propósitos:** | Permitir que o jogador explore o mapa, interaja com locais e entre em combates. |
| **Resumo** | O jogador escolhe locais, explora regiões e pode iniciar batalhas ou encontrar NPCs. |
| **Referência cruzada:** | Classes: *Local, Missão, NPCs* |

|  |  |
| --- | --- |
| **Sequência típica de eventos** | |
| **Ação do ator** | **Resposta do sistema** |
| 1. Jogador seleciona “Explorar o Mundo”. | 2. O sistema exibe o mapa do jogo. |
| 3. Jogador escolhe “Selecionar Local”. | 4. O sistema mostra detalhes e possíveis interações. |
| 5. Jogador confirma entrada no local. | 6. O sistema inicia o evento correspondente (exploração ou combate). |

|  |
| --- |
| **Sequências alternativas:** |
| **Linha 2 :** Se o local estiver bloqueado, o sistema solicita completar uma missão antes. |
| **Linha 5 :** Se o jogador estiver com pouca vida, o sistema recomenda descansar antes de entrar em combate. |

|  |  |
| --- | --- |
| **Use case** | Interagir com NPCs |
| **Atores:** | Jogador |
| **Tipos:** | Secundário |
| **Propósitos:** | Permitir que o jogador converse e receba missões de NPCs. |
| **Resumo** | O jogador pode iniciar diálogos, receber missões ou comprar itens de NPCs. |
| **Referência cruzada:** | Classes: *NPC, Missão, Loja* |

|  |  |
| --- | --- |
| **Sequência típica de eventos** | |
| **Ação do ator** | **Resposta do sistema** |
| 1. Jogador aproxima-se de um NPC. | 2. O sistema exibe opções de interação. |
| 3. Jogador escolhe “Conversar”. | 4. O NPC apresenta falas pré-definidas. |
| 5. Jogador seleciona “Receber Missão”. | 6. O sistema adiciona a missão ao inventário do jogador. |

|  |
| --- |
| **Sequências alternativas:** |
| **Linha 3 :**  Caso o NPC esteja indisponível, o sistema informa “Este personagem não pode conversar agora.” |
| **Linha 5 :** Se o jogador já tiver a missão ativa, o sistema impede aceitar novamente |

|  |  |
| --- | --- |
| **Use case** | Gerenciar Inventário |
| **Atores:** | Jogador |
| **Tipos:** | Secundário |
| **Propósitos:** | Permitir que o jogador visualize, use ou descarte itens. |
| **Resumo** | O jogador pode gerenciar seus itens, incluindo consumíveis e equipamentos. |
| **Referência cruzada:** | Classes: *Inventário, Item, Personagem* |

|  |  |
| --- | --- |
| **Sequência típica de eventos** | |
| **Ação do ator** | **Resposta do sistema** |
| 1. Jogador abre o menu de inventário | 2. O sistema exibe os itens disponíveis. |
| 3. Jogador seleciona “Usar Item”. | 4. O sistema aplica o efeito do item. |
| 5. Jogador escolhe “Descartar Item”. | 6. O item é removido do inventário. |

|  |
| --- |
| **Sequências alternativas:** |
| **Linha 3 :**  Se o item for de uso restrito, o sistema informa que o personagem não pode utilizá-lo. |
| **Linha 5 :** Se o item for essencial, o sistema solicita confirmação antes de excluir. |

|  |  |
| --- | --- |
| **Use case** | Entrar em Combate |
| **Atores:** | Jogador |
| **Tipos:** | Primário |
| **Propósitos:** | Permitir batalhas entre o jogador e inimigos. |
| **Resumo** | O jogador enfrenta inimigos usando ataques físicos e habilidades especiais. |
| **Referência cruzada:** | Classes: Personagem, Habilidade, Inimigo |

|  |  |
| --- | --- |
| **Sequência típica de eventos** | |
| **Ação do ator** | **Resposta do sistema** |
| 1. Jogador entra em uma área de combate. | 2. O sistema inicia o modo de batalha. |
| 3. Jogador seleciona “Atacar Inimigo”. | 4. O sistema calcula dano e atualiza a vida do inimigo. |
| 5. Jogador usa “Habilidade Especial”. | 6. O sistema aplica o efeito da habilidade. |
| 7. Jogador escolhe “Defender”. | 8. O sistema reduz o dano recebido no próximo ataque. |

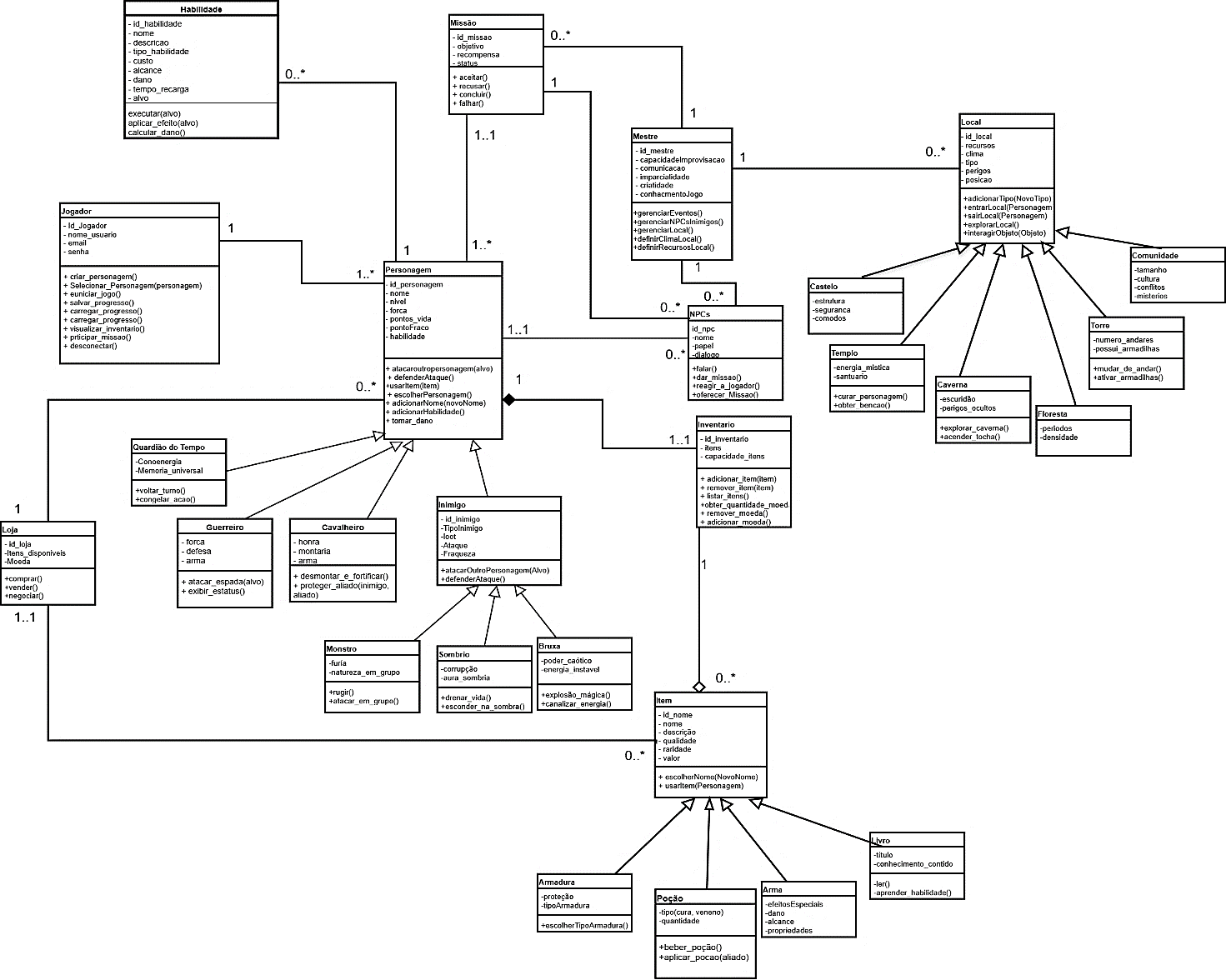
|  |
| --- |
| **Sequências alternativas:** |
| **Linha 3 :**  Se o jogador estiver sem energia, o sistema impede o ataque. |
| **Linha 5 :** Se a habilidade estiver em recarga, o sistema exibe o tempo restante. |

### 4. Requisitos Funcionais e Não Funcionais

| Código | Descrição |
| --- | --- |
| RF01 | O sistema deve permitir que o jogador faça login para acessar o jogo. |
| RF02 | O sistema deve permitir que o jogador crie um personagem informando nome e atributos. |
| RF03 | O sistema deve permitir que o jogador visualize o status do personagem (nível, vida, força, etc.). |
| RF04 | O sistema deve permitir que o jogador explore o mundo, selecionando locais diferentes no mapa. |
| RF05 | O sistema deve permitir que o jogador interaja com NPCs, podendo conversar, receber missões ou comprar itens. |
| RF06 | O sistema deve permitir que o jogador receba missões e as acompanhe até a conclusão. |
| RF07 | O sistema deve permitir que o jogador entre em combate com inimigos durante a exploração. |
| RF08 | O sistema deve permitir que o jogador ataque inimigos utilizando ataques físicos. |
| RF09 | O sistema deve permitir que o jogador use habilidades especiais durante o combate. |
| RF10 | O sistema deve permitir que o jogador defenda-se de ataques inimigos, reduzindo o dano recebido. |
| RF11 | O sistema deve permitir que o jogador gerencie o inventário, visualizando e organizando seus itens. |
| RF12 | O sistema deve permitir que o jogador use, descarte ou consuma itens do inventário. |
| RF13 | O sistema deve permitir que o jogador compre e venda itens com NPCs comerciantes. |
| RF14 | O sistema deve permitir salvar e carregar o progresso do jogador. |
| RF15 | O sistema deve permitir que o jogador encerre a sessão com segurança. |

| Código | Descrição |
| --- | --- |
| RNF01 | O sistema deve ser desenvolvido com interface intuitiva e de fácil navegação, adequada para o público-alvo. |
| RNF02 | O sistema deve garantir segurança das informações do jogador, com autenticação de login e criptografia de senha. |
| RNF03 | O tempo de carregamento de qualquer tela ou funcionalidade não deve exceder 5 segundos. |
| RNF04 | O sistema deve ser compatível com plataformas desktop e web, permitindo acesso multiplataforma. |
| RNF05 | O sistema deve permitir salvar automaticamente o progresso em intervalos configuráveis. |
| RNF06 | O sistema deve possuir banco de dados relacional, armazenando informações de jogadores, personagens, missões e itens. |
| RNF07 | O sistema deve ter design responsivo, ajustando-se a diferentes resoluções de tela. |
| RNF08 | O sistema deve possuir documentação técnica e manual do usuário para suporte e manutenção. |
| RNF09 | O sistema deve manter integridade referencial entre os dados de jogador, personagem e inventário. |
| RNF10 | O sistema deve permitir atualizações e expansões futuras sem comprometer dados anteriores. |

### 5. Diagrama de classe



### 6. Diagrama de Sequências