

## MODELO PARA O DOCUMENTO DE REQUISITOS DO PROJETO

### REQUISITOS FUNCIONAIS

[RF001]	
<b>Nome:</b>	Iniciar Jogo
<b>Descrição:</b>	O sistema deve permitir que o jogador inicie uma nova partida, carregando o cenário inicial, exibindo o personagem em sua posição inicial e preparando os elementos de jogo (obstáculos e lixos).
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Alta
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"><li>● O jogo deve estar aberto.</li><li>● O jogador deve selecionar a opção de iniciar partida.</li></ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"><li>● O jogo é iniciado no cenário.</li><li>● O personagem aparece ativo e pronto para receber comandos de movimento.</li><li>● Os elementos do cenário (obstáculos e lixos) são carregados.</li></ul>
Fluxos de eventos	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"><li>1. O jogador seleciona a opção "Iniciar".</li><li>2. O sistema carrega o cenário inicial.</li><li>3. O sistema posiciona o personagem no ponto inicial do cenário.</li><li>4. O sistema disponibiliza os elementos do cenário (obstáculos e lixos).</li><li>5. O jogo inicia, permitindo que o jogador controle o personagem.</li></ol>
<b>Fluxo secundário 1:</b>	-

<b>[RF002]</b>	
<b>Nome:</b>	Movimentar o personagem
<b>Descrição:</b>	O sistema deve permitir que o jogador controle o personagem podendo realizar o movimento de pulo ao pressionar a tecla designada (a tecla Espaço).
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Alta
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O jogo deve estar em execução.</li> <li>● O personagem deve estar ativo em um cenário.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O personagem se move conforme os comandos permitidos para o cenário em que se encontra.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador pressiona a tecla de movimento para pular (espaço).</li> <li>2. O sistema identifica a entrada.</li> <li>3. O personagem salta ao pressionar a tecla.</li> </ol>
<b>Fluxo secundário 1:</b>	<ol style="list-style-type: none"> <li>1. Se a tecla pressionada não for válida, nenhuma ação é realizada e o jogo permanece em andamento.</li> </ol>

<b>[RF003]</b>	
<b>Nome:</b>	Coletar lixo
<b>Descrição:</b>	O sistema deve permitir que o jogador colete objetos de lixo espalhados pelos cenários.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Essencial
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O personagem deve se movimentar até a posição do objeto de lixo.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O lixo é removido do cenário e adicionado ao inventário do jogador.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador se aproxima do item.</li> <li>2. O sistema detecta colisão com o objeto.</li> <li>3. O item é coletado e removido do cenário.</li> </ol>
<b>Fluxo secundário 1:</b>	-

<b>[RF004]</b>	
<b>Nome:</b>	Exibir contador de lixo
<b>Descrição:</b>	O sistema deve exibir, em tempo real, um contador na interface do jogo que apresenta a quantidade de lixo coletado pelo jogador, organizado por tipo (ex.: plástico, metal, vidro, papel, orgânico, etc.). O contador deve ser atualizado automaticamente conforme o jogador coleta novos itens. Ao coletar a quantidade pré-definida de lixo (15), o jogador avança para a fase de distribuição do lixo nas lixeiras.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Essencial
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O jogador deve estar em uma partida ativa.</li> <li>● O jogo deve ter itens de lixo disponíveis no cenário para coleta.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O contador permanece visível e atualizado durante toda a partida.</li> <li>● Os valores de lixo coletados são armazenados até o fim da rodada.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador inicia a partida.</li> <li>2. O jogo posiciona diferentes tipos de lixo no cenário.</li> <li>3. O jogador coleta um item de lixo.</li> <li>4. O sistema identifica o tipo do item coletado.</li> <li>5. O contador em tela é atualizado, incrementando na categoria correspondente ao tipo de lixo</li> <li>6. O jogador pode visualizar em tempo real a quantidade total e por tipo de lixo coletado.</li> <li>7. O sistema verifica se a quantidade total de lixo coletado atingiu o limite pré-definido (15).</li> <li>8. Se o limite foi atingido, o sistema interrompe a coleta no cenário e avança o jogador para a fase de distribuição do lixo nas lixeiras.</li> </ol>
<b>Fluxo secundário 1:</b>	-

<b>[RF005]</b>	
<b>Nome:</b>	Desviar obstáculos
<b>Descrição:</b>	O sistema deve permitir que o jogador evite obstáculos no cenário (ex.: carros, árvores caídas, pedras).
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Média
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O obstáculo deve estar posicionado no cenário.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O jogador passa pelo obstáculo sem colisão.</li> <li>● Caso colida, a fase atual reinicia.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador se move em direção ao obstáculo.</li> <li>2. O jogador desvia a tempo.</li> <li>3. O jogo continua normalmente.</li> </ol>
<b>Fluxo secundário 1:</b>	<ol style="list-style-type: none"> <li>1. O jogador não desvia.</li> <li>2. O sistema reinicia a fase atual e esvazia o inventário.</li> </ol>

<b>[RF006]</b>	
<b>Nome:</b>	Reiniciar cenário
<b>Descrição:</b>	O sistema deve reiniciar automaticamente o cenário do jogo sempre que o jogador colidir com um obstáculo. Dessa forma, a partida é reiniciada, garantindo a continuidade do jogo.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Média
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O jogador deve estar em uma partida ativa.</li> <li>● O cenário deve conter obstáculos ou elementos que possam causar colisão.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O cenário é reiniciado imediatamente após a colisão.</li> <li>● O contador de lixo coletado e a pontuação retornam ao valor inicial.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador inicia a partida.</li> <li>2. O jogador percorre o cenário e coleta itens de lixo.</li> <li>3. O jogador colide com um obstáculo.</li> <li>4. O sistema detecta a colisão.</li> <li>5. O sistema reinicia o cenário, reposicionando o jogador e os elementos do jogo em seu estado inicial.</li> </ol>
<b>Fluxo secundário 1:</b>	<ol style="list-style-type: none"> <li>1. O jogador desvia do obstáculo.</li> <li>2. O jogo continua normalmente.</li> </ol>

<b>[RF007]</b>	
<b>Nome:</b>	Classificar lixo nas lixeiras
<b>Descrição:</b>	O sistema deve permitir que o jogador descarte o lixo coletado em lixeiras específicas (plástico, papel, vidro, metal, orgânico) no final de cada fase.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Alta
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O jogador deve ter lixo coletado no inventário.</li> <li>● Lixeira deve estar disponível no cenário.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● Lixo é removido do inventário e contabilizado na lixeira correta.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador interage com uma lixeira.</li> <li>2. O sistema verifica o tipo de lixo.</li> <li>3. O lixo é corretamente classificado.</li> <li>4. O cenário é atualizado (melhora visual).</li> </ol>
<b>Fluxo secundário 1:</b>	<ol style="list-style-type: none"> <li>1. O jogador interage com uma lixeira incorreta.</li> <li>2. O sistema detecta a inconsistência.</li> <li>3. O sistema exibe uma mensagem de erro.</li> <li>4. O lixo permanece no inventário do jogador.</li> </ol>

<b>[RF008]</b>	
<b>Nome:</b>	Exibir e transformar cenários
<b>Descrição:</b>	O sistema deve exibir três cenários (cidade, floresta e oceano), que são transformados após a coleta e correta classificação do lixo.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Essencial
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>• O jogador deve completar a coleta e classificação de todos os lixos do cenário atual.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>• O cenário é transformado em sua versão limpa (ex.: cidade sem poluição, floresta com árvores saudáveis, oceano limpo).</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador completa a classificação correta de todos os lixos.</li> <li>2. O sistema transforma o cenário exibindo sua versão limpa.</li> </ol>
<b>Fluxo secundário 1:</b>	<ol style="list-style-type: none"> <li>1. Se houver lixo não classificado, o cenário não é transformado.</li> </ol>

<b>[RF009]</b>	
<b>Nome:</b>	Avançar para o próximo cenário
<b>Descrição:</b>	O sistema deve avançar automaticamente para o próximo cenário após a conclusão do anterior.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Essencial
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O cenário atual deve estar limpo e concluído.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O próximo cenário é carregado.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador conclui o cenário atual.</li> <li>2. O sistema carrega o próximo cenário automaticamente.</li> </ol>
<b>Fluxo secundário 1:</b>	<ol style="list-style-type: none"> <li>1. Se o cenário não for concluído, o jogo permanece no mesmo estágio.</li> </ol>

<b>[RF010]</b>	
<b>Nome:</b>	Exibir tela de finalização
<b>Descrição:</b>	O sistema deve exibir uma tela de finalização com mensagem educativa ao término do último cenário.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Essencial
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O jogador deve concluir todos os cenários.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O sistema apresenta mensagem educativa e encerra o jogo.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	<ol style="list-style-type: none"> <li>1. O jogador conclui o último cenário.</li> <li>2. O sistema exibe tela de finalização com mensagem educativa.</li> </ol>
<b>Fluxo secundário 1:</b>	-

## REQUISITOS NÃO FUNCIONAIS

[RNF01]	
<b>Nome:</b>	Interface gráfica em 2D
<b>Descrição:</b>	O jogo deve apresentar gráficos bidimensionais (2D).
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Alta
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"><li>• O jogo deve estar em execução.</li></ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"><li>• O cenário é exibido em estilo 2D.</li></ul>
Fluxos de eventos	
<b>Fluxo principal:</b>	-
<b>Fluxo secundário 1:</b>	-

[RNF02]	
<b>Nome:</b>	Usabilidade intuitiva na jogabilidade
<b>Descrição:</b>	O jogo deve ser intuitivo, garantindo que os jogadores compreendam facilmente a dinâmica e o objetivo principal a partir de uma breve descrição inicial. A jogabilidade deve ser fluida, permitindo que os usuários interajam normalmente com os controles e cumpram os objetivos do jogo sem dificuldades ou confusão. O design dos elementos de interface e as interações devem ser autoexplicativos.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Alta
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● Jogador deve ter acesso ao jogo.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● Jogador comprehende facilmente os comandos e objetivos.</li> </ul>
Fluxos de eventos	
<b>Fluxo principal:</b>	-
<b>Fluxo secundário 1:</b>	-

<b>[RNF03]</b>	
<b>Nome:</b>	Feedback visual e sonoro para ações do jogador
<b>Descrição:</b>	O sistema deve fornecer respostas visuais (animações, mudanças gráficas) e sonoras (efeitos de som) às ações do jogador.
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Média
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● O jogador realiza uma ação (movimento, coleta, reciclagem).</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● O sistema responde com feedback correspondente.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	-
<b>Fluxo secundário 1:</b>	-

<b>[RNF04]</b>	
<b>Nome:</b>	Mensagens educativas relacionadas aos ODS
<b>Descrição:</b>	O sistema deve exibir mensagens de conscientização ambiental alinhadas aos Objetivos de Desenvolvimento Sustentável (ODS).
<b>Atores:</b>	Jogador
<b>Prioridade:</b>	Média
<b>Entradas e pré-condições:</b>	<ul style="list-style-type: none"> <li>● Jogador deve concluir cenários ou ações importantes.</li> </ul>
<b>Saídas e pós-condições:</b>	<ul style="list-style-type: none"> <li>● Mensagem educativa é apresentada na tela.</li> </ul>
<b>Fluxos de eventos</b>	
<b>Fluxo principal:</b>	-
<b>Fluxo secundário 1:</b>	-