

---

**Projeto de Tópicos**

**Avançados Em**

**Computação II**

---

**Tópicos Avançados Em Computação II**

**2024 – FACET/UFGD**

**Projeto: Dragons**

**Equipe: Alyson Machado, Mateus Irala, Otávio Vieira  
e Vinicius Stefanés**

## Conteúdo

<b>1 INTRODUÇÃO</b>	<b>3</b>
<b>2 MOTIVAÇÃO</b>	<b>3</b>
<b>3 VISÃO DA SOLUÇÃO</b>	<b>4</b>
<b>4 ESCOPO DA SOLUÇÃO</b>	<b>4</b>
4.1 PRINCIPAIS ENTREGÁVEIS	4
<b>5 LIMITES E RESTRIÇÕES DA SOLUÇÃO</b>	<b>4</b>
Funcionalidades Excluídas	4
Premissas	5
Restrições	5
<b>6 DESCRIÇÃO DOS USUÁRIOS</b>	<b>5</b>
<b>7 CRONOGRAMA INICIAL</b>	<b>6</b>
<b>8 INTRODUÇÃO</b>	<b>7</b>
8.1 ESPECIFICAÇÃO	7
8.1.1 <i>Identificação dos requisitos funcionais do sistema</i>	7
8.1.2 <i>Identificação dos requisitos não funcionais do sistema</i>	8
8.1.3 <i>Identificação dos casos de uso</i>	8
Casos de Uso do Dragons	8
<b>9 ATORES</b>	<b>10</b>
<b>10 CASOS DE USO</b>	<b>10</b>
[UC01] Jogar	10
[UC02] Spawnar Dragão	11
[UC03] Colisão	11
[UC04] Controlar Dragão	11
[UC05] Spawner Comida	12
[UC06] Adicionar Porto	12
[UC07] Game Over	12
[UC08] Comer Fruta	13
[UC09] Aumentar Tamanho	13
<b>11 DIAGRAMA DE CLASSES</b>	<b>14</b>
<b>12 DIAGRAMA DE SEQUÊNCIA</b>	<b>15</b>
<b>13 DIAGRAMA DE ATIVIDADES</b>	<b>16</b>
<b>14 REFERÊNCIAS</b>	<b>16</b>

## 1 Introdução

O projeto "Dragons" é uma reinterpretação moderna do clássico Snake Game, que introduz novas dinâmicas e um modo multiplayer competitivo. Enquanto mantém a essência do jogo original, onde o objetivo é crescer ao comer maçãs e evitar colisões, "Dragons" adiciona uma camada extra de estratégia e interação entre jogadores.

No "Dragons", dois jogadores controlam dragões em um campo de jogo repleto de desafios. Além das colisões com as paredes e os próprios corpos, agora eles devem evitar colidir com o corpo do oponente. Esse aspecto competitivo é potencializado pela introdução de novas mecânicas, como paredes espalhadas pelo mapa e túneis (buracos) que permitem aos dragões se deslocarem de uma parte do campo para outra, criando novas oportunidades de ataque e defesa.

Essas inovações não apenas aumentam a complexidade e a diversão do jogo, mas também incentivam o desenvolvimento de novas estratégias, tornando cada partida única. A mecânica tradicional de comer maçãs para crescer continua presente, servindo como uma ferramenta essencial para a sobrevivência e a vitória. O jogador ganha ao sobreviver mais tempo, forçando o oponente a colidir com algum obstáculo.

Neste relatório, exploraremos os detalhes do desenvolvimento do "Dragons", desde a concepção e design até a implementação das mecânicas de jogo. Analisaremos como as novas características foram integradas para criar uma experiência de jogo envolvente e desafiadora, mantendo a nostalgia e a simplicidade do Snake Game original.

## 2 Motivação

O projeto "Dragons" surgiu na disciplina de tópicos avançados II com o objetivo de criar um jogo inspirado no clássico Snake Game, agregando novas dinâmicas e um modo multiplayer competitivo. O jogo original, conhecido por sua simplicidade e viciante jogabilidade, carece de elementos que incentivem a interação direta entre jogadores e a elaboração de estratégias mais complexas.

<b>O problema é...</b>	O Snake Game clássico não oferece uma experiência multiplayer rica e interativa. Falta diversidade de mecânicas que tornem cada partida única e estratégica.
<b>Que afeta...</b>	Os jogadores que esperam um jogo que mantenha a essência do Snake Game, mas que também ofereça uma competição direta entre jogadores, com novos desafios e possibilidades estratégicas
<b>O impacto disto é...</b>	A falta de inovação pode levar à desmotivação dos jogadores, que podem optar por jogos mais modernos e complexos. Para os desenvolvedores, há uma perda de oportunidade de se destacar no mercado de retrô-games
<b>A solução seria...</b>	<ul style="list-style-type: none"><li>• <b>Modo Multiplayer Competitivo:</b> Introduzir um modo onde dois jogadores possam competir diretamente, controlando dragões.</li><li>• <b>Novas Mecânicas de Jogo:</b> Adicionar elementos como paredes espalhadas pelo mapa e túneis (buracos) que</li></ul>

	<p>permitem transposição de áreas do campo, aumentando a complexidade e as oportunidades estratégicas.</p> <ul style="list-style-type: none"> <li>• <b>Essência do Jogo Original:</b> Manter a mecânica tradicional de comer maçãs para crescer, que será crucial para a sobrevivência e vitória.</li> <li>• <b>Desafios Adicionais:</b> Incluir novos obstáculos e elementos de jogo que incentivem o desenvolvimento de novas estratégias a cada partida.</li> </ul>
--	--

### 3 Visão da Solução

Proporcionar uma experiência de jogo competitiva e estratégica, modernizando o clássico Snake Game com novas dinâmicas e um modo multiplayer envolvente.

### 4 Escopo da Solução

Nessa seção, estão descritas as principais necessidades para a aplicação desenvolvida:

Necessidades	Categoria
1. Modo multiplayer competitivo com dois jogadores	<i>Essencial</i>
2. Manter a mecânica tradicional de comer maçãs para crescer	<i>Essencial</i>
3. Sistema de pontuação e determinação de vencedor	<i>Importante</i>
4. Interface de usuário intuitiva e responsiva	<i>Importante</i>

#### 4.1 Principais entregáveis

**Aplicação de Jogo "Dragons":** O jogo completo com todas as funcionalidades especificadas, pronto para uso pelos jogadores.

**Relatório Final:** Um documento abrangente que cobre todas as fases do desenvolvimento, desde a concepção até a implementação e testes.

### 5 Limites e Restrições da Solução

#### Funcionalidades Excluídas

- **Modo Online:** A funcionalidade de permitir que jogadores se conectem e joguem pela internet não será implementada nesta versão do jogo. Esta funcionalidade pode ser planejada para projetos futuros, considerando a complexidade adicional de desenvolver e manter um sistema de multiplayer online estável e seguro.
- **Suporte para Plataformas Móveis:** O jogo "Dragons" será desenvolvido apenas para desktop (Windows, MacOS e Linux) e não incluirá suporte para dispositivos móveis nesta fase do projeto. A adaptação para plataformas móveis pode ser considerada em versões futuras.

- **Modos de Jogo Avançados:** Modos de jogo adicionais, como torneios, campanhas ou desafios específicos, não serão desenvolvidos nesta versão inicial. Eles poderão ser considerados para expansões ou futuras versões do jogo.

## Premissas

- **Recursos Limitados:** O projeto "Dragons" será desenvolvido com recursos limitados de tempo e equipe, o que impõe a necessidade de priorizar funcionalidades essenciais e importantes.
- **Foco no Multiplayer Local:** A premissa é que o jogo será jogado principalmente por dois jogadores no mesmo dispositivo, focando na experiência multiplayer local.
- **Compatibilidade com Desktops:** O jogo deve ser compatível com as principais plataformas de desktop, garantindo uma experiência de jogo fluida e sem bugs.

## Restrições

- **Orçamento Limitado:** O desenvolvimento do projeto deve se manter dentro do orçamento alocado, o que pode restringir a implementação de funcionalidades adicionais.
- **Prazos Estritos:** O projeto deve ser concluído dentro do prazo definido pela disciplina de Engenharia de Tópicos Avançados da Computação II, exigindo um gerenciamento eficiente do tempo e dos recursos.
- **Tecnologia Utilizada:** A implementação deve ser feita utilizando as tecnologias e linguagens de programação python usando o pygame, como plataforma.

## 6 Descrição dos Usuários

### 1. Jogadores Casuais

- **Perfil:** Indivíduos que jogam ocasionalmente para entretenimento. Geralmente, buscam jogos que sejam fáceis de aprender e jogar, mas que ainda proporcionam diversão e desafio.
- **Desafios:**
  - Falta de jogos simples e divertidos que ofereçam uma experiência multiplayer local.
  - Dificuldade em encontrar jogos que sejam fáceis de entender, mas que mantenham o interesse a longo prazo.
- **Necessidades:**
  - Interface intuitiva e fácil de usar.
  - Experiências de jogo rápidas e envolventes.

### 2. Jogadores Competitivos

- **Perfil:** Indivíduos que procuram desafios e oportunidades para competir com outros jogadores. Estes jogadores apreciam a complexidade e a profundidade nas mecânicas de jogo.
- **Desafios:**
  - Falta de jogos que combinam simplicidade e estratégias complexas.
  - Necessidade de jogos que permitam competição direta e interativa entre jogadores.
- **Necessidades:**
  - Mecânicas de jogo que ofereçam várias estratégias e táticas.
  - Sistema de pontuação.
  - Desafios adicionais que aumentem a complexidade do jogo.

### 3. Entusiastas de Jogos Retro

- **Perfil:** Indivíduos que têm uma afinidade por jogos clássicos e retrô, apreciando a nostalgia e simplicidade desses jogos.
- **Desafios:**
  - Dificuldade em encontrar jogos modernos que capturem a essência dos clássicos.
  - Falta de inovações que tragam novos elementos para jogos retrô sem perder a essência original.
- **Necessidades:**
  - Manutenção da mecânica clássica do Snake Game.
  - Novas dinâmicas que agreguem valor ao jogo original.
  - Estilo visual e sons que remetam aos jogos clássicos

## 7 Cronograma Inicial

Esta seção apresenta um cronograma inicial para o projeto, destacando quais serão os principais marcos do projeto, o que conterão e quando eles ocorrerão.

Fases/Marcos do projeto	Entregáveis	Data de início prevista	Data de término prevista
<i>Tema do Projeto(1 semana)</i>	<i>Escolher um tema de jogo para criar um projeto incremental em Python usando a biblioteca Pygame que será desenvolvido ao longo do semestre.</i>	<i>19/03/2024</i>	<i>27/03/2024</i>
<i>Criação do github(1 semana)</i>	<i>Criação do repositório no Github da equipe Adicionar no Github um protótipo inicial que carrega uma imagem de fundo, uma música e pelo menos um objeto que se move com o teclado na tela</i>	<i>27/03/2024</i>	<i>03/04/2024</i>
<i>Criação da apresentação(3 semanas)</i>	<i>Preparar o protótipo para apresentação em sala.</i>	<i>03/04/2024</i>	<i>17/04/2024</i>

<i>Inclusão de itens no projeto (3 semanas)</i>	<i>Incluir no projeto os seguintes itens: a. Código com aplicação de máscaras nos sprites; b. Sprites animados com sequência de imagens; c. Movimentação de fundo de tela (caso tenha); d. Refatoração de código para classes; e. Diagramas UML atualizados.</i>	<i>24/05/2024</i>	<i>08/05/2024.</i>
<i>Últimas melhorias para o projeto(3 semanas)</i>	<i>a. Game com menus (opções, login, cadastro); b. Conexão com base de dados; c. Melhorias na jogabilidade/níveis, etc.</i>	<i>15/05/2024</i>	<i>05/06/2024</i>

## 8 Introdução

Este documento especifica o sistema, as funcionalidades esperadas e os principais diagramas do projeto, ao projeto Dragons referente ao trabalho de tópicos avançados II. O objetivo do documento é fornecer aos desenvolvedores as informações necessárias para o projeto.

### 8.1 Especificação

A seguir são descritos os principais requisitos do sistema bem como os casos de uso derivados.

#### 8.1.1 Identificação dos requisitos funcionais do sistema

A seguir são listados os principais requisitos funcionais do sistema.

<b>F001</b>	Modo multiplayer competitivo para dois jogadores
<b>F002</b>	Implementação de colisão no mapa
<b>F003</b>	Mecânica de comer maçãs para crescer e aumentar a pontuação
<b>F004</b>	Sistema de detecção de colisões entre dragões e obstáculos.
<b>F005</b>	Sistema de pontuação e determinação de vencedor
<b>F006</b>	Implementação de efeitos sonoros e visuais adequados ao tema

#### 8.1.2 Identificação dos requisitos não funcionais do sistema

A seguir são listados os principais requisitos não funcionais do sistema.

<b>RNF001</b>	Desempenho: O jogo deve rodar com fluidez em hardware moderado
<b>RNF002</b>	Manutenibilidade: Código bem documentado e modularizado
<b>RNF003</b>	Estabilidade: O jogo deve ser robusto e minimamente suscetível a crashes

### 8.1.3 Identificação dos casos de uso

## Casos de Uso do Dragons

### 1. Introdução

Este documento apresenta os casos de uso do jogo "Dragons", um remake do clássico Snake Game com novas dinâmicas e um modo multiplayer competitivo.

### 2. Identificação dos Atores

- **Jogador 1:** Usuário que controla um dragão no jogo.
- **Jogador 2:** Usuário que controla outro dragão no jogo.

### 3. Casos de Uso

#### UC01: Iniciar Jogo

- **Descrição:** Inicia uma nova partida de "Dragons".
- **Atores:** Jogador 1, Jogador 2
- **Pré-condições:** Nenhuma.
- **Fluxo Principal:**
  - O jogador seleciona a opção "Iniciar Jogo" no menu principal.
  - O sistema exibe a tela de jogo com dois dragões em posições iniciais opostas.
  - Os jogadores podem controlar seus dragões usando as setas do teclado.
  - O jogo termina quando um dos jogadores colide com uma parede, outro jogador ou seu próprio corpo.
  - O sistema exibe a tela de fim de jogo com o placar final.
- **Fluxos Alternativos:**
  - [FA01]: Se nenhum jogador pressionar uma tecla após 5 segundos, o sistema exibe uma mensagem de aviso e reinicia a partida.
- **Fluxos de Erro:**
  - [FE01]: Se ocorrer um erro durante a inicialização do jogo, o sistema exibe uma mensagem de erro e retorna ao menu principal.

#### UC02: Jogar

- **Descrição:** Permite que os jogadores controlem seus dragões durante a partida.
- **Atores:** Jogador 1, Jogador 2
- **Pré-condições:** O jogo deve estar iniciado (UC01).
- **Fluxo Principal:**
  - Os jogadores pressionam as setas do teclado para controlar seus dragões.
  - Os dragões se movem na direção indicada pelas setas.
  - Se um dragão comer uma maçã, ele cresce em tamanho.



- Se um dragão colidir com uma parede, outro jogador ou seu próprio corpo, ele perde a partida.
- **Fluxos Alternativos:**
  - [FA01]: Se um jogador pressionar a tecla "Esc" durante a partida, o sistema exibe um menu de pausa com opções para retomar o jogo, ir ao menu principal ou sair do jogo.
- **Fluxos de Erro:**
  - [FE01]: Se ocorrer um erro durante a partida, o sistema exibe uma mensagem de erro e retorna ao menu principal.

### UC03: Pausar/Retomar Jogo

- **Descrição:** Permite pausar e retomar o jogo.
- **Atores:** Jogador 1, Jogador 2
- **Pré-condições:** O jogo deve estar em andamento (UC02).
- **Fluxo Principal:**
  - O jogador pressiona a tecla "Esc" durante a partida.
  - O sistema exibe um menu de pausa com opções para retomar o jogo, ir ao menu principal ou sair do jogo.
  - Se o jogador selecionar a opção "Retomar Jogo", o sistema volta para a tela de jogo e a partida continua.
- **Fluxos Alternativos:**
  - [FA01]: Se o jogador pressionar a tecla "Esc" novamente enquanto o menu de pausa estiver aberto, o menu é fechado e o jogo retorna para a tela de jogo.

### UC04: Ir ao Menu Principal

- **Descrição:** Retorna ao menu principal do jogo.
- **Atores:** Jogador 1, Jogador 2
- **Pré-condições:** O jogo deve estar em andamento (UC02) ou pausado (UC03).
- **Fluxo Principal:**
  - O jogador seleciona a opção "Menu Principal" no menu de pausa (UC03) ou pressiona a tecla "Esc" na tela de jogo (UC02).
  - O sistema exibe o menu principal do jogo.
- **Fluxos Alternativos:**
  - [FA01]: Se o jogador confirmar a ação de sair do jogo no menu principal, o sistema fecha o jogo.

### UC05: Sair do Jogo

- **Descrição:** Sai do jogo "Dragons".
- **Atores:** Jogador 1, Jogador 2
- **Pré-condições:** O jogo deve estar no menu principal (UC04).
- **Fluxo Principal:**
  1. O jogador seleciona a opção "Sair do Jogo" no menu principal.

## 9 Atores

A tabela abaixo descreve brevemente cada ator da aplicação.

Ator	Descrição
<b>Usuário que controla o dragão azul</b>	Este jogador utiliza as setas do teclado para mover o dragão pela tela, coletar maçãs e evitar colisões com paredes, o outro jogador ou seu próprio corpo. O objetivo do jogador dragão azul é sobreviver por mais tempo que o jogador dragão vermelho e alcançar a maior pontuação.
<b>Usuário que controla o dragão vermelho</b>	Este jogador utiliza o teclado para mover o dragão pela tela, coletar maçãs e evitar colisões com paredes, o outro jogador ou seu próprio corpo. O objetivo do jogador dragão vermelho é sobreviver por mais tempo que o jogador do dragão azul e alcançar a maior pontuação.

## 10 Casos de Uso

Essa seção apresenta todos os requisitos funcionais da aplicação, especificados como casos de uso.

### Diagrama de casos de uso

#### [UC01] Jogar

- **Prioridade:**
  1. ☒ *Essencial*
  2. ☐ *Importante*
  3. ☐ *Desejável*
- **Ator(es):** Jogador
- **Descrição:** Iniciar uma partida do jogo "Dragons".
- **Pré-condições:**
  1. O jogador está no menu principal.
  2. O sistema está operacional.
- **Pós-condições:**
  1. Partida iniciada com dragões e comida no campo de jogo.
- **Fluxo Principal:**
  1. O jogador seleciona a opção de jogar no menu principal.
  2. O sistema inicia uma nova partida.
  3. Inclui [UC02] Spawnar Dragão.
  4. Estende [UC05] Spawner Comida.
  5. Estende [UC06] Adicionar Porto.
  6. Jogador controla o dragão (Estende [UC04] Controlar Dragão).
  7. Partida termina quando ocorre uma colisão (Estende [UC03] Colisão).
  8. Sistema exibe o resultado da partida.

#### [UC02] Spawnar Dragão

- **Prioridade:**
  1. ☒ *Essencial*
  2. ☐ *Importante*

- 3. ☐ Desejável
- **Ator(es):** Sistema
- **Descrição:** Criar um dragão no campo de jogo.
- **Pré-condições:**
  1. A partida foi iniciada.
- **Pós-condições:**
  1. Dragão aparece no campo de jogo.
- **Fluxo Principal:**
  1. O sistema cria o dragão na posição inicial.
  2. O Dragão está pronto para ser controlado pelo jogador.

### **[UC03] Colisão**

- **Prioridade:**
  1. ☒ Essencial
  2. ☐ Importante
  3. ☐ Desejável
- **Ator(es):** Sistema
- **Descrição:** Detectar colisões durante o jogo.
- **Pré-condições:**
  1. Dragão em movimento.
- **Pós-condições:**
  1. Colisão detectada, a partida termina.
- **Fluxo Principal:**
  1. O sistema monitora o movimento do dragão.
  2. Detecta colisão com paredes, corpos ou outros dragões.
  3. Estende [UC07] Game Over.

### **[UC04] Controlar Dragão**

- **Prioridade:**
  1. ☒ Essencial
  2. ☐ Importante
  3. ☐ Desejável
- **Ator(es):** Jogador
- **Descrição:** O jogador controla o movimento do dragão.
- **Pré-condições:**
  1. A partida foi iniciada.
  2. Dragão foi criado.
- **Pós-condições:**
  1. Dragão se move conforme controle do jogador.
- **Fluxo Principal:**
  1. O jogador usa controles para mover o dragão.
  2. O sistema atualiza a posição do dragão no campo de jogo.

### **[UC05] Spawner Comida**

- **Prioridade:**
  1. ☒ Essencial
  2. ☐ Importante
  3. ☐ Desejável
- **Ator(es):** Sistema
- **Descrição:** Gerar comida no campo de jogo.

- **Pré-condições:**
  1. A partida foi iniciada.
- **Pós-condições:**
  1. Comida aparece no campo de jogo.
- **Fluxo Principal:**
  1. O sistema gera comida aleatoriamente no campo de jogo.
  2. O jogador pode comer a comida (Estende [UC08] Comer Fruta).

### **[UC06] Adicionar Porto**

- **Prioridade:**
  1. ☐ Essencial
  2. ☐ Importante
  3. ☒ Desejável
- **Ator(es):** Sistema
- **Descrição:** Adicionar portos (túneis) no campo de jogo.
- **Pré-condições:**
  1. A partida foi iniciada.
- **Pós-condições:**
  1. Portos aparecem no campo de jogo.
- **Fluxo Principal:**
  1. O sistema cria portos aleatoriamente no campo de jogo.

### **[UC07] Game Over**

- **Prioridade:**
  1. ☒ Essencial
  2. ☐ Importante
  3. ☐ Desejável
- **Ator(es):** Sistema
- **Descrição:** Finalizar a partida quando ocorrer uma colisão.
- **Pré-condições:**
  1. Colisão detectada.
- **Pós-condições:**
  1. A partida terminou.
  2. O resultado da partida é exibido.
- **Fluxo Principal:**
  1. Sistema detecta colisão (Incluir [UC03] Colisão).
  2. O sistema finaliza a partida.
  3. O sistema exibe o resultado da partida.

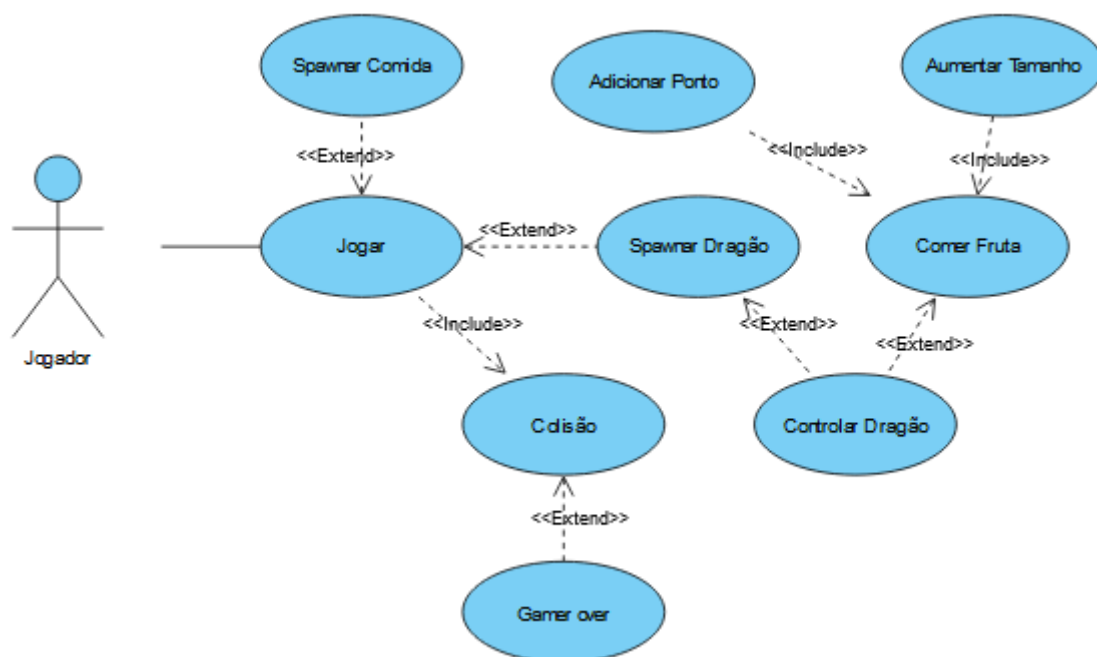
### **[UC08] Comer Fruta**

- **Prioridade:**
  1. ☒ Essencial
  2. ☐ Importante
  3. ☐ Desejável
- **Ator(es):** Jogador
- **Descrição:** Dragão come a fruta e aumenta de tamanho.
- **Pré-condições:**
  1. A comida foi gerada no campo de jogo.
- **Pós-condições:**
  1. Dragão aumenta de tamanho.

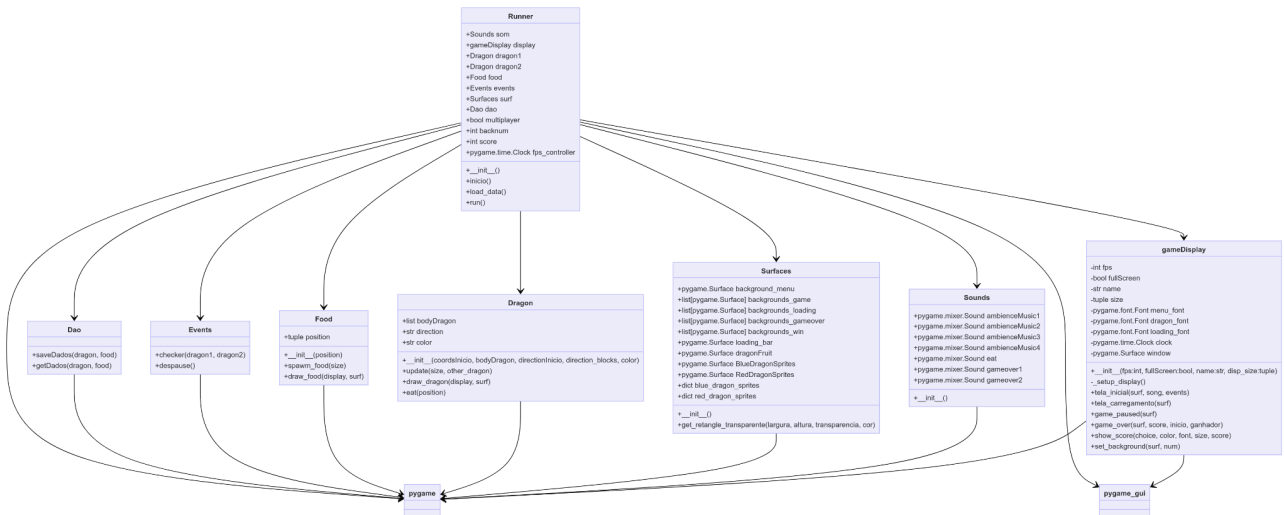
- **Fluxo Principal:**
  1. O jogador move o dragão para comer a fruta.
  2. O sistema detecta que a fruta foi comida.
  3. O sistema aumenta o tamanho do dragão (Inclui [UC09] Aumentar Tamanho).

## [UC09] Aumentar Tamanho

- **Prioridade:**
  1. ☒ Essencial
  2. ☐ Importante
  3. ☐ Desejável
- **Ator(es):** Sistema
- **Descrição:** Aumentar o tamanho do dragão após comer a fruta.
- **Pré-condições:**
  1. Dragão comeu a fruta.
- **Pós-condições:**
  1. O tamanho do dragão é aumentado.
- **Fluxo Principal:**
  1. O sistema aumenta o tamanho do dragão.



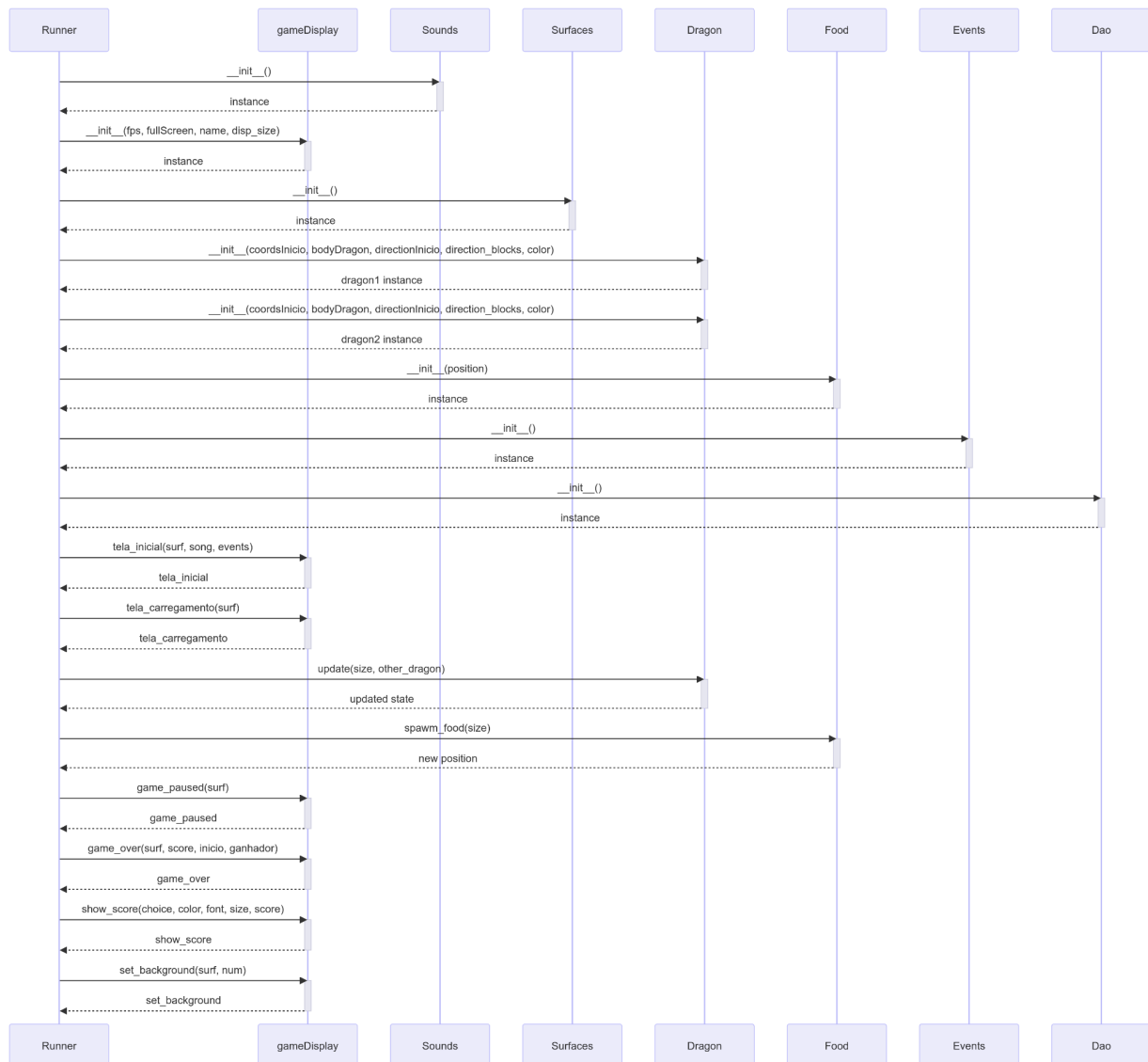
## 11 Diagrama de Classes



Todos os relacionamentos são realizadas apenas pela classe Runner

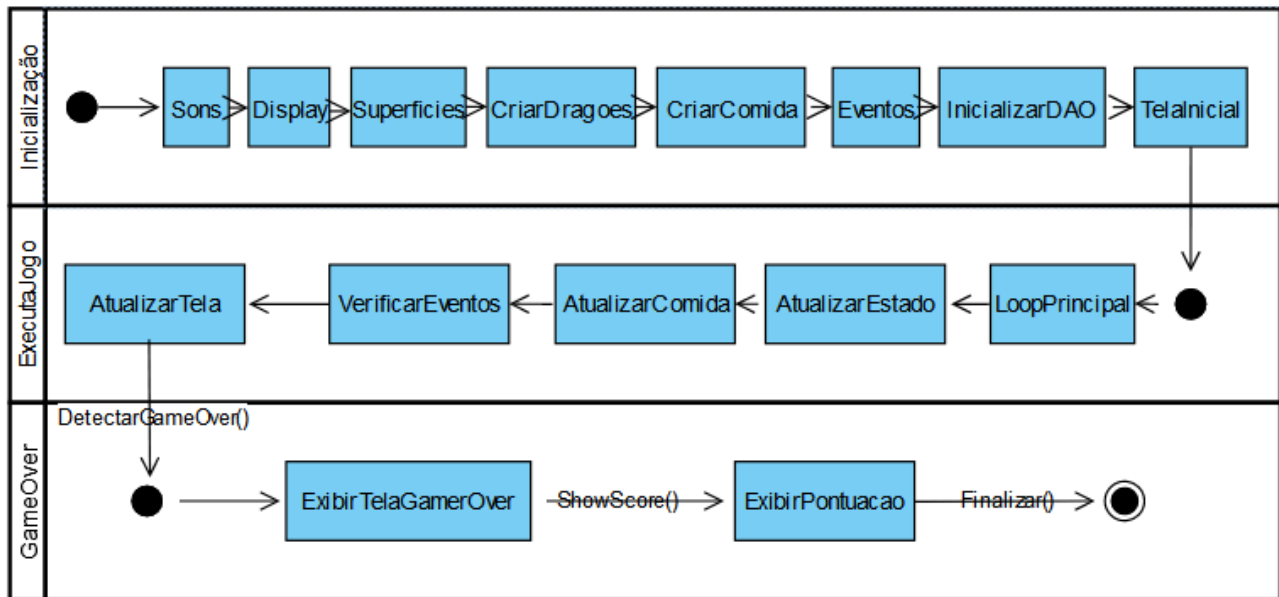
## 12 Diagrama de Sequência

Diagrama de sequência para quando o jogo é iniciado:



## 13 Diagrama de Atividades

O objetivo desta seção é apresentar o fluxo de execução dos processos que compõem a aplicação utilizando o diagrama de atividades.



## 14 Referências

Nesta seção, são apresentadas as referências utilizadas para a elaboração deste documento.

[Regras da ABNT](#)