

Rafael Ferreira Rigo - 24149

Samuel Rosa Parra - 24155

Francisco da Fonseca Rodrigues

Estrutura de dados

Junho de 2025

Relatório de Projeto – Jogo da Forca em C#

Introdução

Este relatório descreve o desenvolvimento de um jogo da forca utilizando a linguagem C#, com interface gráfica elaborada no Windows Forms. O objetivo principal foi aplicar os conhecimentos de Estrutura de Dados para a criação de uma aplicação lúdica, funcional e com boa interação com o usuário.

Desenvolvimento

Objetivos do Trabalho

O trabalho teve como objetivo implementar um jogo da forca completo com:

- Gerenciamento de palavras através de uma classe dicionário.
- Controle de tempo de jogo via Timer.
- Contagem de erros e acertos.
- Exibição gráfica das etapas da forca.
- Verificação de vitória ou derrota do jogador.
- Interface gráfica interativa com botões de letras e controle de início e fim de jogo.

Cronograma de Desenvolvimento

- **28/05/2025:** Criação inicial do projeto no Visual Studio. Estruturação das classes e definição dos métodos principais.

- **30/05/2025:** Desenvolvimento da classe **Dicionario**, responsável por armazenar e fornecer as palavras do jogo.
 - **31/05/2025:** Implementação do layout gráfico, inserção de imagens correspondentes às fases do jogo (forca).
 - **12/06/2025:** Implementação do evento **Click** dos botões (letras), que identifica a letra clicada, desativa o botão e atualiza o estado do jogo.
 - **13/06/2025:** Programação do **timerTempo**, métodos **btnIniciar**, **Fim**, **Ganhou** e **Perdeu**. Realização de testes finais e ajustes.
-

Descrição dos Métodos Implementados

- **Classe **Dicionario**:**

Responsável por armazenar uma lista de palavras e fornecer aleatoriamente uma delas para o jogo. Garante variedade de palavras a cada nova partida.

- **Layout das Imagens:**

As imagens representam visualmente a evolução da forca a cada erro do jogador. Foram

organizadas em um array de imagens, alteradas conforme o número de erros acumulados.

- **Evento `Click` dos Botões:**

Cada botão representa uma letra do alfabeto. Ao ser clicado, verifica se a letra está contida na palavra secreta. Caso positivo, revela as posições corretas. Caso negativo, incrementa o contador de erros e atualiza a imagem da força.

- **`tmrTempo`:**

Controla o tempo limite de jogo. A cada segundo, reduz o tempo restante. Quando o tempo chega a zero, o jogo termina automaticamente, acionando o método `Perdeu`.

- **`btnIniciar`:**

Responsável por iniciar uma nova partida. Este método reseta as variáveis, escolhe uma nova palavra, limpa o layout e reinicia o timer.

- **Método `Fim`:**

Método que encerra o jogo, desabilitando todos os botões e o timer. Utilizado tanto em caso de vitória quanto de derrota.

- **Método `Ganhou`:**

Exibe uma mensagem de vitória, para o timer e chama o método `Fim`.

- **Método `Perdeu`:**

Exibe uma mensagem de derrota, mostra a palavra correta, para o timer e chama o método `Fim`.

Erros Encontrados e Soluções Aplicadas

- **Erro 1:** Falha na atualização das imagens da força.
 - **Solução:** Implementação correta de um array de imagens e controle do índice conforme o número de erros.
- **Erro 2:** Timer não reiniciava ao clicar em "Iniciar".
 - **Solução:** Inclusão de `timerTempo.Stop()` e `timerTempo.Start()` no método `btnIniciar`.
- **Erro 3:** Letras já clicadas podiam ser clicadas novamente.
 - **Solução:** Desativação do botão após clique usando `btn.Enabled = false`.

Dificuldades Encontradas

- Configuração do Timer para funcionar adequadamente junto com a lógica de vitória e derrota.
- Integração das imagens para representar corretamente as etapas do jogo.
- Sincronização dos estados dos botões com o progresso da palavra secreta.

Fontes de Pesquisa

- Documentação oficial da Microsoft (docs.microsoft.com) – Uso de Timer, PictureBox, eventos Click.
 - Stack Overflow – Dúvidas sobre manipulação de controles Windows Forms.
 - YouTube – Tutoriais de projetos simples de forca em C#.
-

Conclusão

O desenvolvimento deste projeto proporcionou uma experiência prática completa na aplicação dos conceitos de Estrutura de Dados e manipulação de interface gráfica em C#. Aprendemos sobre controle de eventos, uso de timers, manipulação de imagens, além da importância de uma boa estruturação de código para facilitar manutenções futuras. O jogo resultante cumpre os requisitos propostos e funciona de forma estável e interativa.