

**UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMATICA**

**CAIO FELIPE FERREIRA NUNES
GUSTAVO MACHADO SILVA
LUCAS ROSSI KLEIN
YASMIN AGNES SIMÃO**

**TRABALHO PRÁTICO
FASE 1**

**Disciplina: Técnicas de Construção de Programas (INF01120)
Professor: Marcelo Soares Pimenta**

Porto Alegre, setembro de 2025.

Contents

1	INTRODUÇÃO	3
2	REQUISITOS FUNCIONAIS	4
3	INTERFACE DE USUÁRIO	5
4	DEFINIÇÕES DE CLASSES	8
4.1	TextHandler	8
4.2	Instrument	8
4.3	Note	8
4.4	Volume	9
4.5	MusicalContext	9
4.6	Action	9
4.7	ActionMapper	9

1 INTRODUÇÃO

Nesse trabalho será projetado e desenvolvido um software que recebe um texto, string de caracteres, e gera uma música, baseada em um mapeamento de caractere para uma nota ou mudança em algum parâmetro musical, como instrumento, oitava ou volume.

Este trabalho está sendo realizado como atividade avaliativa da disciplina de Técnicas de Construção de Programas, e tem como objetivo desenvolver e avaliar as habilidades dos estudantes acerca da construção de software de qualidade.

Nesta primeira fase serão entregues as especificações das funcionalidades do sistema, a definição das classes que serão implementadas posteriormente e um croqui da interface do software com o usuário que também será implementada nas fases seguintes. Note que essas especificações podem sofrer algumas alterações durante as próximas etapas, apesar de serem pensadas com fatores de qualidade do software, como confiabilidade e modularidade, em mente.

2 REQUISITOS FUNCIONAIS

O Software gerador de música é dependente de um texto para leitura, além de recursos que possibilitem a mudança na configuração musical do sistema, funções essas que devem ser levadas em conta na montagem de classe e estruturação do código. Sua funcionalidade é especificada de acordo com a seguinte lista de requisitos funcionais:

- O sistema deve ler um texto de entrada contendo comandos musicais.
- O sistema deve identificar o caractere atual do texto.
- O sistema deve verificar se há mais caracteres a serem processados.
- O sistema deve associar comandos de texto a ações musicais.
- O sistema deve atualizar a ação associada a um comando existente.
- O sistema deve remover a ação associada a um comando.
- O sistema deve conter mapeamento para executar a ação correspondente a um comando.
- O sistema deve permitir alterar a nota musical atual.
- O sistema deve permitir inserir um silêncio na música.
- O sistema deve permitir dobrar o volume atual.
- O sistema deve permitir trocar o instrumento musical em execução.
- O sistema deve manter o estado atual da execução musical (instrumento, volume e nota).
- O sistema deve fornecer métodos para consultar o instrumento, volume e nota atuais.
- O sistema deve fornecer métodos para modificar o instrumento, volume e nota atuais.
- O sistema deve reproduzir a música a partir do contexto musical definido.

3 INTERFACE DE USUÁRIO

Para a produção das telas foi pensada em uma aplicação composta por apenas uma tela, tendo ela dois formatos: sem o texto inserido (Figura 1), com o texto inserido (Figura 3). A motivação por trás dessa escolha foi decidida pelo grupo após analisar os requisitos do enunciado da aplicação. Com o auxílio da plataforma Figma foram montadas os modelos iniciais de como seriam apresentadas as telas ao usuários.



Figure 1: Tela Inicial - Sem música gerada (Fonte: Os autores)

Na primeira tela é apresentado ao usuário 3 blocos: um para inserir o texto, um com funções para alterar alguns parâmetros da música (Escala Musical, Tonalidade, BPM, etc.) e outra com duas opções o resultado da som gerado e outra com um guia de uso do sistema (Figura 2). As demais telas aqui presentes também serão compostas dos mesmos três blocos, possuindo apenas algumas alterações no bloco mais à direita, composto pelo resultado da música gerada e guia de uso.

🎵 Gerador de Música a partir de Texto
 Digite um texto e converta-o em uma sequência de notas musicais usando parâmetros personalizáveis

Texto de Entrada

Digite o texto que será convertido em música...

Parâmetros Musicais

Escala Musical ▼
Tonalidade ▼

Tempo (BPM): 120
 Lento (60) ————— Rápido (200)

Oitava Base: 4
 Grave (2) ————— Agudo (6)

Duração da Nota: 0.5s
 Curta (0.1s) ————— Longa (2.0s)

🎵 Gerar Música

Resultados Guia de Uso

📖 Guia de Mapeamento Pré-definido

🎵 Notas Musicais

- A Nota Lá
- B Nota Si
- C Nota Dó
- D Nota Ré
- E Nota Mi
- F Nota Fá
- G Nota Sol
- H Nota Si Bemol

🔊 Silêncio e Controle

a,b,c,d,e,f,g,h Silêncio ou Pausa

Espaço Dobra volume (máx: 200%)

Outras consoantes Repete nota anterior ou pausa

🎹 Mudanças de Instrumento

! Bandoleon (MIDI #24)

O,o,i,i,u,u Gaita de Foles (MIDI #110)

Digitos pares Instrumento atual + valor do dígito

Digitos ímpares Tubular Bells (MIDI #15)

Nova linha Ondas do Mar (MIDI #123)

; Tubular Bells (MIDI #15)

, Órgão (MIDI #17)

🔊 Controles de Oitava

? ou . Aumenta uma oitava (volta ao padrão se no limite)

📖 Exemplo de Uso

Texto: "ABC def!G H"

Resultado:

- A = Nota Lá
- B = Nota Si
- C = Nota Dó
- Espaço = Dobra volume
- d,e,f = Pausas
- ! = Muda para Bandoleon
- G = Nota Sol (com Bandoleon)
- Espaço = Dobra volume novamente
- H = Nota Si Bemol

Figure 2: Tela Inicial - Guia de uso da aplicação (Fonte: Os autores)

Ao selecionar o “Guia de Uso” o usuário conseguirá ter acesso às funcionalidades de cada caractere e um breve exemplo de uso (Figura 2). Por último temos a tela com a música gerada (Figura 3), aqui o bloco do “Resultado” é alterado com algumas informações sobre a música gerada sendo elas: notas geradas, sequência de notas, detalhes das notas. Além disso é apresentado um novo bloco logo abaixo dos detalhamentos da música com o player da música com duas opções, “Reproduzir” e “Pausar”.

🎵 Gerador de Música a partir de Texto

Digite um texto e converta-o em uma sequência de notas musicais usando parâmetros personalizáveis

Texto de Entrada

Hello World!

Parâmetros Musicais

Escala Musical: **Maior** Tonalidade: **C**

Tempo (BPM): 135
Lento (60) Rápido (200)

Oitava Base: 4
Grave (2) Agudo (6)

Duração da Nota: 1.1s
Curta (0.1s) Longa (2.0s)

🎵 Gerar Música

🎵 Notas Geradas

🎵 12 notas ⌚ 6.1s

12 Total de Notas 11 Notas Musicais 1 Pausas 6.1s Duração

Sequência de Notas

B4 F4 F4 F4 undefined4 f C4 undefined4 E4 F4 E4 A1

Detalhes das Notas

B4 #1	493.9 Hz	0.49s
F4 #2	349.2 Hz	0.49s
F4 #3	349.2 Hz	0.49s
F4 #4	349.2 Hz	0.49s
--	NaN Hz	

Reprodução

▶ Reproduzir □ Parar

Progresso 6.1s

Pronto para reproduzir

Figure 3: Tela Resultados - Com música gerada (Fonte: Os autores)

4 DEFINIÇÕES DE CLASSES

Com o objetivo de implementar as ideias propostas nos requisitos funcionais e na interface de usuário, projetamos classes para integrar as diversas funcionalidades do nosso software. A Figura 4 nos mostra as classes e suas relações.

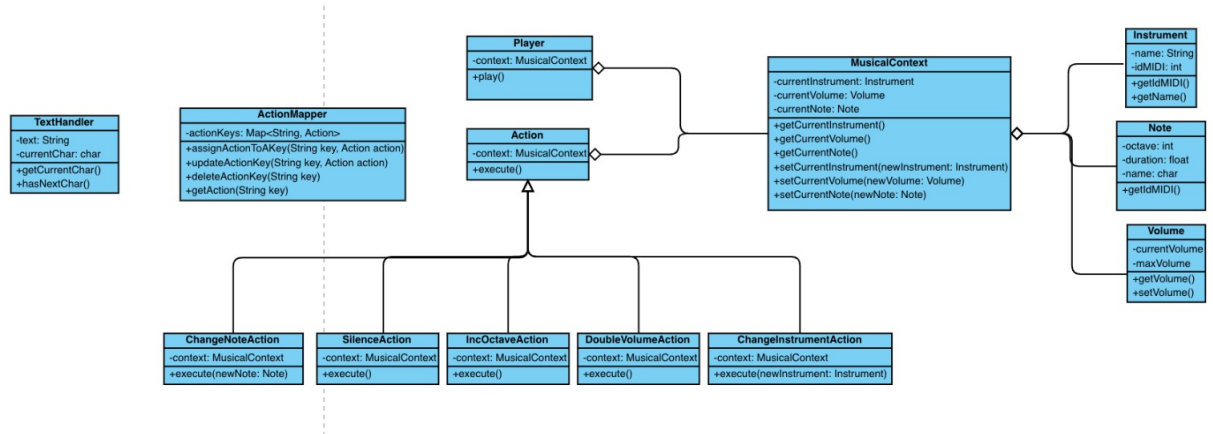


Figure 4: Diagrama de classes (Fonte: Os autores)

Conforme apresentado na Figura 4, as classes pensadas envolvem diferentes funcionalidades que são integradas entre as mesmas. Com o objetivo de manter cada classe com um escopo específico, o funcionamento do software foi separado em sete classes principais e outras classes filhas.

4.1 TextHandler

A classe TextHandler é responsável por lidar com a string inserida pelo usuário. Suas responsabilidades envolvem percorrer a string retornando o char atual e salvando a posição e informar se há um próximo char ou a string chegou no final.

4.2 Instrument

A classe Instrument tem como responsabilidade guardar as propriedades que identificam um instrumento e os tipos de instrumento que estão disponíveis no software.

4.3 Note

A classe Note define as propriedades de uma nota musical que envolvem oitava, duração, entre outras. A classe tem como objetivo estipular o que o software necessita para lidar com notas musicais.

4.4 Volume

A classe Volume define as propriedades que controlam o volume da aplicação, como volume atual, volume máximo, obter volume e definir volume.

4.5 MusicalContext

A classe MusicalContext define um tipo de objeto que tem como propriedades um instrumento, uma nota, e um volume. A função da classe é manter essas propriedades, habilitar a obtenção e definição delas.

4.6 Action

A classe Action é responsável por definir uma ação de alteração dentro de uma ou mais propriedade do contexto musical. A classe possui classes filhas e cada uma das filhas representa uma ação específica.

4.7 ActionMapper

Essa classe ActionMapper tem como objetivo mapear um char específico a uma ação específica. Ela conta com um HashMap de char para Action e funções para definir, atualizar e excluir um mapeamento.