

txt2midi

Bruno S. Ardenghi Gonçalves George Jia Chen Qian Jorgefran Souza Batista

Universidade Federal do Rio Grande do Sul
Instituto de Informática
INF01120 — Técnicas de Construção de Programas
Prof. Marcelo Pimenta

2025/2

Interface Gráfica e Design System

► Stack Tecnológica:

- Desenvolvido inteiramente em **Python** com paradigma Orientado a Objetos.
- Migração estratégica: de *Tkinter* (protótipo) para o ecossistema **GNOME**.

► Objetivos de UX/UI:

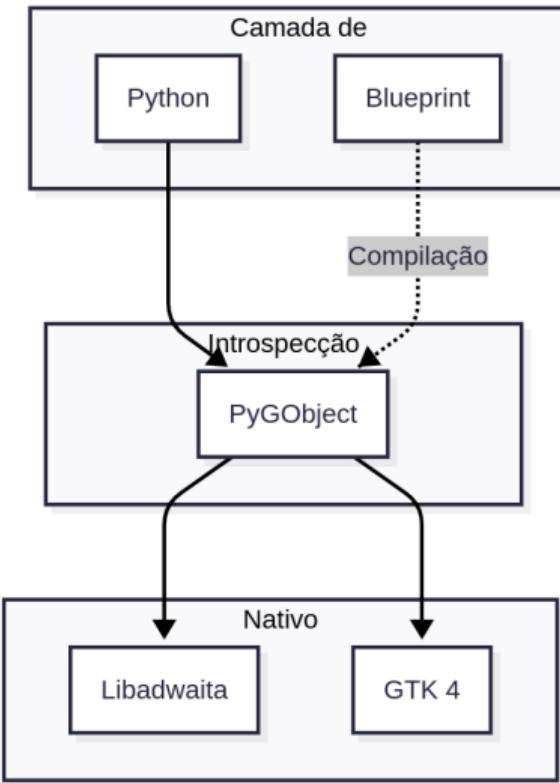
- Aderência visual nativa ao ambiente Ubuntu/Linux.
- Polimento estético superior aos toolkits padrões do Python.

► Fluxo de Execução:

Comunicação síncrona entre interface e lógica de negócios.



Arquitetura da Interface e Bibliotecas



- ▶ **GTK 4 (The GIMP Toolkit):**
 - ▶ Utilização de aceleração de hardware (GPU) para renderização.
 - ▶ Gerenciamento moderno de eventos e widgets.
- ▶ **Libadwaita:**
 - ▶ Implementação das *Human Interface Guidelines* (HIG) do GNOME.
 - ▶ Widgets adaptativos e responsivos "out-of-the-box".
- ▶ **PyGObject (Binding):**
 - ▶ Camada de introspecção (*GObject Introspection*).
 - ▶ Gera classes Python dinamicamente a partir de bibliotecas C.

Blueprint: UI Declarativa

```
using Gtk 4.0;
using Adw 1;

template $ConfigPanel : Adw.
    PreferencesGroup {
    title: _("Parâmetros iniciais");

    Adw.SpinRow spin_bpm {
        title: _("BPM");
        adjustment: Adjustment {
            lower: 20; upper: 400;
            step-increment: 5; value: 120;
        };
    }
}
```

► Linguagem de Marcação:

- Uso da nova sintaxe *Blueprint* do GNOME.
- Sintaxe concisa, similar a CSS/JSON, compilada para XML.

► Separação de Responsabilidades:

- **View:** Definida estaticamente nos arquivos Blueprint.
- **Controller:** Lógica injetada via Python e templates compostos.
- Facilita manutenção e leitura do código visual.

Sintaxe e Modos de Operação

Comando	Função
A–G [#,b]	Notas Musicais e Acidentes
R, P	Pausa (<i>Rest</i>)
An, Rn	Duração ($1/n$)
.	Ponto de aumento (+50%)
Ln	Define duração padrão
On	Define Oitava absoluta (0-10)
>, <	Sobe/Desce oitava relativa
Tn	Define BPM (Tempo)
Vn	Define Volume (0-127)
In	Define Instrumento (General MIDI)

► Modo Padrão (Texto Livre):

- Segue estritamente o mapeamento do enunciado.
- Focado em simplicidade.

► Modo MML:

- Abas independentes (multitarefa).
- Controle granular de oitavas e dinâmicas.
- Permite composições complexas.

Importação e Persistência

- ▶ **Conversão de Binário para Texto:**
 - ▶ O sistema lê o arquivo .mid e o transforma de volta em código editável.
- ▶ **Atualização Automática da Interface:**
 - ▶ Ao carregar um arquivo, o BPM, Volume e Instrumento da tela se ajustam sozinhos aos valores do MIDI.
- ▶ **Fluxo de "Salvar e Carregar":**
 - ▶ O arquivo MIDI serve como o "save file" do projeto.
 - ▶ Permite fechar o programa e continuar editando a música depois exatamente de onde parou.

GtkSourceView: Syntax Highlighting

```
<context id="standard">
  <include>
    <context id="bpm-plus" style-ref="keyword">
      <match>BPM\+</match>
    </context>
    <context id="notes" style-ref="note">
      <match case-sensitive="false">[A-H]</match>
    </context>
    <context id="rest" style-ref="comment">
      <match>;</match>
    </context>
  </include>
</context>
```

► Widget Especializado:

- ▶ Uso da biblioteca GtkSourceView (padrão em editores como gedit).

► Definição de Linguagem (.lang):

- ▶ Criação de arquivos XML personalizados.
- ▶ Mapeamento de tokens via Expressões Regulares (RegEx).

► Integração Visual:

- ▶ O código adapta as cores automaticamente ao tema do sistema (Claro/Escuro).
- ▶ Melhora a legibilidade dos comandos musicais.

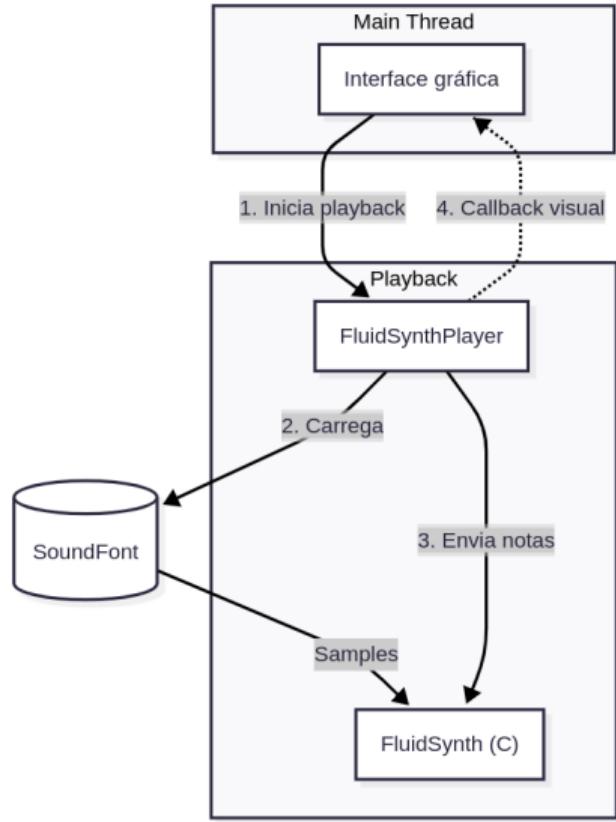
Engine de Áudio e SoundFonts

► FluidSynth & pyfluidsynth:

- ▶ Integração direta via biblioteca dinâmica (DLL/.so).
- ▶ Elimina latência de chamadas de sistema (subprocessos).

► Tecnologia SoundFont (.sf2):

- ▶ Síntese baseada em amostras (*samples*).
- ▶ Flexibilidade: O usuário pode carregar bancos de sons personalizados.
- ▶ Incluímos dois para demonstração.



Feedback Visual em Tempo Real

Texto de entrada

Insira o texto para geração da música.

```
B8 > C#8 < B4 A4 G#8 A4 > F#8 < A8 > E8  
F#8 D8 A8 B8 D8 > C#8 < D8 D8 > D8 < D8  
D8 > E8 < D8 F#8 < A8 > E8 F#8 D8 A8 B8  
D8 D8 D8 D8 D8 D4 A4 E8 E4 E4 D8 C#4  
D4 E2 A4 E4 E8 E4 D8 C#4 D4 E2 R8 G#4 E4  
E4. E8 A4 E4 E4 C#4 D2 D2 E4. < A8 A8 A8  
> F#4 E4. < A8 A8 A8 > F#4 E4. E8 E8 E8  
E8 E8 E4 C#4 < B4 > C#4 C#4. < A8 A8 A8  
> F#4 E4. < A8 A8 A8 > A4 E4. E8 E8 E8  
E8 E8 E4 C#4 < B4 A4 A2 A4. A8 A4 > C#4
```

► Sistema de Callbacks:

- O sintetizador notifica a interface a cada evento de nota.

► Mapeamento Texto-Áudio:

- O algoritmo identifica o "chunk"(trecho de texto) responsável pela nota atual.
- Aplica uma tag de estilo visual (highlight) instantâneo.

► Experiência do Usuário:

- Permite depuração visual da composição.
- Facilita o acompanhamento pedagógico da execução.

Obrigado!