

Padrão State

Padrão Comportamental

I. Problema: Implementando um Player de Áudio

Recebemos a tarefa para implementar um player de áudio cujos controles se comportam de forma diferente dependendo do atual estado de reprodução do player. A interface da aplicação (gráfica ou texto), basicamente, vai responder a 4 ações: **play** (reproduzir música), **stop** (parar a música), **next** (avançar para a próxima música) e **prev** (voltar à música anterior). A Figura 1 ilustra a interface gráfica básica de uso do player.

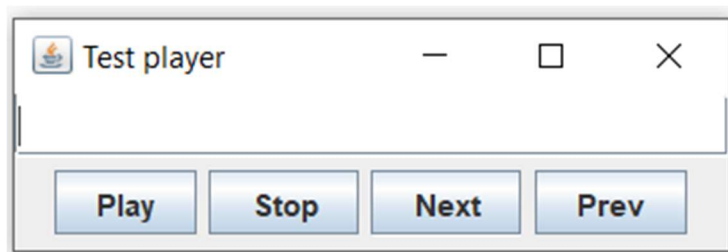


Figura 1. Ilustração da interface do player mp3.

O pressionamento de cada botão do **Player** se comporta de maneira diferente de acordo com o **estado atual** de reprodução do player. Os estados que podem ser assumidos pelo player são:

- a) **ready** (pronto para reproduzir)
- b) **playing** (reproduzindo)
- c) **locked** (reprodução interrompida)

Cada estado define suas próprias regras de comportamento em resposta aos eventos de pressionamento dos botões **stop**, **play**, **next** e **previous**:

Modo **ready**:

- Se receber o evento **stop**, muda para o estado de **locked**;
- Se receber o evento **play**, muda para o estado de **playing** e inicia o playback;
- Se receber o evento **next**, não faz nada, pois está travado;
- Se receber o evento **previous**, não faz nada, pois está travado.

Modo **playing**:

- Se receber o evento **stop**, muda para o estado de **locked** e para a reprodução. Além disso, armazena a trilha corrente que estava sendo reproduzida;
- Se receber o evento **play**, muda para o estado de **ready** e efetua uma pausa na reprodução;
- Se receber o evento **next**, reproduz a próxima música;
- Se receber o evento **previous**, reproduz a música anterior

Modo **locked**:

- Se receber o evento **stop**, caso o player esteja reproduzindo música, alterna para o estado **ready** e interrompe a execução; Caso contrário, informa que está "travado".
- Se receber o evento **play**, muda para o estado de **ready**.
- Se receber o evento **next**, não faz nada, pois está travado;
- Se receber o evento **previous**, não faz nada, pois está travado

O **Player** controla internamente o status de reprodução da música, a playlist definida para reprodução e a música atual que está sendo tocada. Quando um player é instanciado, ele começa no estado **ready** e tem seu status de reprodução igual a **false**. Além disso, realiza o povoamento automático de uma playlist com 12 músicas. Algumas operações de responsabilidade do Player são elencadas a seguir:

- a) **setPlaying()**: ativa o status de "estar em reprodução" para true ou false
- b) **isPlaying()**: sinaliza se o player está em reprodução ou não
- c) **startPlayback()**: informa qual trilha está sendo reproduzida
- d) **nextTrack()**: avança para a próxima trilha e retorna a trilha em reprodução
- e) **previousTrack()**: retrocede para a trilha anterior e retorna a trilha em reprodução

II. Tarefa

Neste exercício, observe os seguintes pontos:

1. A definição de como o padrão **State** foi configurado no Player para mudar de comportamento de acordo com as ações executadas e lógica da transição de estados apresentada neste documento.
2. O Diagrama de Classe para facilitar a visão geral da solução.

