



deti

universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

UNIVERSIDADE DE AVEIRO
MESTRADO EM ENGENHARIA INFORMÁTICA

Projeto 2

INTERAÇÃO MULTIMODAL

DOCENTES:

PROF. ANTÓNIO TEIXEIRA

PROF. MIGUEL SILVA

PROF. SAMUEL SILVA

António Ramos Nº Mec 101193
João Pedro Pereira Nº Mec 106346

21 de maio de 2022

Conteúdo

1	Introdução	2
2	Execução	3
2.1	Execução	3
2.2	Geração da playlist	3
2.3	Iniciar mmiframework	3
2.4	Iniciar os programas	3
3	Instruções	4
3.1	Inicialização	4
3.2	Volume	4
3.3	Mudar música	4
3.4	Pausa	4
3.5	Velocidade	5
3.6	Repetição	5
3.7	Aleatória	5
3.8	Silenciar	5
3.9	Nome da música	6
3.10	Tempo da música	6
3.11	Pedir a musica	6
3.12	Terminar	6
4	Desenvolvimento	7
5	Bibliografia	8

1 Introdução

O objetivo deste trabalho é desenvolver uma aplicação que tenha interação com uma outra aplicação, usando tecnologias de reconhecimento de voz.

A aplicação escolhida pelo nosso grupo foi VLC media player, onde temos que colocar as instruções de voz a comunicar com a aplicação.

Para a execução deste projeto é necessário a aplicação VLC media player, Visual Studio, mmiframework fornecido pelo professor, e as bibliotecas da Microsoft de reconhecimento de voz. Existem certas ações para a aplicação criada ser executada, que serão apresentadas neste relatório.

2 Execução

2.1 Execução

Com o objetivo de criar uma simulação de uma instalação de serviço que consiste em dois servidores em série (tandem), cada um com sua própria fila FIFO. Um cliente que completa o serviço no servidor 1 segue para o servidor 2, enquanto um cliente que conclui o serviço no servidor 2 sai da instalação.

Os tempos entre chegadas dos clientes ao servidor 1 são variáveis aleatórias exponenciais com média de 1 minuto.

Os tempos de atendimento dos clientes no servidor 1 são variáveis aleatórias exponenciais com média de 0,7 minuto, e no servidor 2 são variáveis aleatórias exponenciais com média de 0,9 minuto.

2.2 Geração da playlist

Para iniciar com as músicas, é necessário ter uma playlist, para tal foi criada uma solução para criar automaticamente essa playlist. Para ser criada corretamente essa playlist, é necessário colocar as músicas na pasta “music” na raiz do projeto. Existem alguns requisitos que se deve cumprir nas músicas adicionadas na pasta. Esses requisitos são:

- Não colocar acentos nem no nome da música nem no artista.
- Para ser bem reconhecido a música a que quer colocar, é aconselhado que a música tenha o nome em português
- O nome do ficheiro deve conter primeiro o nome do artista/banda e de seguida o nome da música, separado por um hífen “-”.

2.3 Iniciar mmiframework

Aceder à pasta “IM” e abrir a linha de comando com o ficheiro que tem o nome de “CMD”. Escrever “java -jar mmiframeworkV2.jar”, isto fará que inicie a aplicação java fornecida pelo professor.

2.4 Iniciar os programas

Agora apenas é necessário abrir o projeto “AppGui” e “speechModality”. Executar ambos os projetos e começar a dar instruções de voz.

3 Instruções

3.1 Inicialização

Para iniciar a música é necessário diz alguma destas instruções:

- Iniciar/Começar playlist
- Iniciar/Começar música

Após esta instrução é iniciada a playlist com todas as músicas que estão na pasta "music"

3.2 Volume

Para fazer alguma alteração no volume é necessário diz alguma destas instruções:

- Reduzir/Diminuir/Descer/baixar o volume/som
- Aumentar/Subir o volume/som
- mais alto
- mais baixo
- pode colocar/por a musica mais alto

3.3 Mudar música

Para fazer alguma alterações na música a ouvir é necessário diz alguma destas instruções:

- podes mudar/trocar/alterar a musica para a seguinte/proxima
- podes mudar/trocar/alterar para a anterior
- proxima musica
- musica anterior
- podes coloca/por a musica seguinte/anterior

3.4 Pausa

Para pausar música, é necessário diz alguma destas instruções:

- podes pausar a musica
- pausa a musica
- podes continuar a musica

- continua a musica

3.5 Velocidade

Para fazer alguma alteração na velocidade é necessário diz alguma destas instruções:

- podes aumentar/subir a velocidade
- Reduzir/Diminuir/Descer/baixar a velocidade
- mais rapido
- mais lento/devagar
- podes coloca/por a musica mais rapido

3.6 Repetição

Para fazer alguma alteração na velocidade é necessário diz alguma destas instruções:

- podes repetir a musica
- podes não repetir a musica
- deixar de repetir
- desativar repetição

3.7 Aleatória

Para fazer escolher as músicas aleatoriamente é necessário diz alguma destas instruções:

- podes escolher tu
- escolhe tu
- desativar/inativar/cancelar a musica aleatoria
- parar de escolher as musicas

3.8 Silenciar

Para fazer com que fique sem som, mode mute é necessário diz alguma destas instruções:

- silencio
- sem som
- cala te

Para retirar este modo, tem as seguintes instruções:

- retornar o som
- com som
- podes cantar

3.9 Nome da música

Para descobrir o nome da músicas, pode-se dizer alguma destas instruções:

- como se chama esta musica
- qual o nome

3.10 Tempo da música

Para descobrir o tempo que a musica tem, pode-se dizer alguma destas instruções:

- qual a duracao da musica
- quanto tempo tem a musica

3.11 Pedir a musica

Para pedir para tocar certa musica, pode-se dizer alguma destas instruções:

- Colocar esta musica "nome da musica"(nome do artista opcional)
- Põe a musica "nome da musica"(nome do artista opcional)

3.12 Terminar

Para terminar o VLC, dizer alguma destas instruções:

- Xau
- desliga por favor
- Vou embora
- Fechar a aplicação

4 Desenvolvimento

Numa fase inicial foi decidido pelo grupo, fazer as instruções básicas do vLC, tais como:

- Iniciar musica
- Pausar/continuar
- Mudar Volume
- Mudar para a seguinte/anterior

Numa fase onde já tínhamos estas últimas instruções a funcionar, começamos a implementar instruções, que retorna-se valores. Então foram criadas as instruções para retornar o nome da música e a duração da música.

Mais para a frente, verificamos que seria possível pedir para tocar a música dizendo o nome, onde avançamos o desenvolvimento, utilizando uma gramática dinâmica, que faz a leitura da pasta "music" e recria o ficheiro "*Dynamic_grammar.grxml*", onde se refere a *referenciado nas regras do* *ptG.grxml*, o ficheiro principal da gramática.

Com isto feito, pretendemos que o processo seja o mais automático possível e verificamos que para inicializar com uma playlist nova, tínhamos que criar a playlist no VLC e só depois iniciar o nosso desenvolvimento. Então para contornar isto, verificamos que o ficheiro de uma playlist do VLC é um ficheiro XSPF, que é um ficheiro de formato XML e como já tínhamos realizado algo para gerar um ficheiro de formato XML, decidimos fazer o mesmo para o ficheiro da playlist.

No final, verificamos que para certos valores de reconhecimento continham por vezes informação correta para a instrução que foi mencionada. Para tal, implementamos um processo do programa questionar se realmente o que entendeu foi o que ele disse, e caso responda sim, então será executado o mesmo.

5 Bibliografia

1. [https://docs.microsoft.com/en-us/previous-versions/office/developer/communication-server-2007/bb857891\(v=office.12\)](https://docs.microsoft.com/en-us/previous-versions/office/developer/communication-server-2007/bb857891(v=office.12))
2. <https://wiki.videolan.org/Talk:Console/>
3. <https://www.nextivr.com/creating-dynamic-grxml-grammars-with-c-and-asp-net.html>