

PROJETO FINAL

Relatório

Laboratórios de Informática

Telmo Sauce, Bárbara Moreira, Ricardo Covelo,
Rodrigo Loureiro



PROJETO FINAL

Laboratórios de Informática

DETI

Universidade de Aveiro

Telmo Sauce, Bárbara Moreira, Ricardo Covelo, Rodrigo Loureiro
(104056)barbarammoreira@ua.pt, (102576) rlr@ua.pt
(102668) ricardo.covelo11@ua.pt, (104428)telmobelasauce@ua.pt

11/07/2021

Índice

1	Introdução	2
2	HTML	3
2.1	index.html	3
2.2	music.html	4
2.3	excertos.html	5
2.4	mix.html	6
2.5	about.html	7
3	script.js & link.py	8
4	Base de Dados	9
5	Resultados	10
5.1	Funcionamento dos Votos	10
5.2	Funcionamento do Browse	10
6	Conclusões Finais	12

Capítulo 1

Introdução

No âmbito da cadeira de Laboratórios de Informática (LABI) , para o projeto final foi-nos atribuída a tarefa da criação de um site capaz de reproduzir músicas, misturá-las e ainda adicionar e reproduzi-las. Assim dividimos o nosso relatório em seis tópicos sendo o primeiro a Introdução. No capítulo *html* explicamos cada uma das quatro páginas web presentes no nosso trabalho e recorremos a imagens ilustrativas como complemento dessa mesma explicação. Destacámos ainda dois capítulos para dois temas de grande importância sendo estes mesmos os ficheiro de código *script.js*, que contém as funções javascript utilizadas, e *link.py*, o nosso principal ficheiro python, e a DataBase (DB) onde explicámos a obtenção de dados da nossa base de dados. Por ultimo, no capítulo *Resultados* apresentámos imagens que comprovam o bom funcionamento do nosso código e no capítulo *Conclusões Finais* as respectivas percentagens atribuídas a cada membro do grupo assim como um breve texto sobre a realização deste projeto.

Capítulo 2

HTML

A nossa Interface Web é composta por cinco páginas criadas através de HyperText Markup Language (HTML), Cascading Style Sheets (CSS) e Java Script (JS). Páginas essas que iremos listar de seguida.

2.1 index.html

Este ficheiro de código trata-se simplesmente do login no nosso site. O utilizador entra, inicia a sua sessão escrevendo o seu *username* no retângulo por baixo do texto "*Nickname*" e, clicando no botão "*Start*", entra no site. Para este efeito, no nosso ficheiro *link.py*, a função *login()* verifica se o username inserido é ou não válido, se for avança para a nossa primeira página caso contrário a página reenicia e é lhe pedido novamente que seja inserido o nome.



Figura 2.1: Login do index.html

2.2 music.html

Após o login o utilizador entra na nossa página *music.html*. A mesma apresenta na forma de uma tabela a listagem das músicas disponíveis, o respetivo artista, dois botões que permitem a votação na mesma e o número de votos. É de notar.... (escrever caso não tenhamos conseguido fazer a cena dos votos). O funcionamento do botão *PLAY* é assegurado pela função *play()*. A abertura desta página é realizada através da função python *music()*.

Coluna da tabela que conta o número de votos				
#	Artist	Music	Botões que permitem reproduzir o som/música selecionado(a)	Botão correspondente aos votos negativos
0	Marshmello	Friends	PLAY	▽ ▲
0	Outro	Outra	PLAY	▽ ▲

Botão correspondente aos votos positivos

Figura 2.2: Tabela do music.html

2.3 excertos.html

Correspondente á página com a nossa *excertos_table*, esta página tem disponibilizada vários excertos de música podendo os mesmos ser adicionados através da barra do *Browse* em baixo da mesma. Assim que adicionado, através do botão *Add*, é criada uma nova *row* com o mesmo ficheiro, sendo possível a sua reprodução. Estes excertos serão os mesmos a ser usadas posteriormente no misturador. Feito o "pedido" de adição de um novo ficheiro, o conteúdo do mesmo é enviado para a nossa DB sendo inserida uma nova linha na nossa tabela com os dados do ficheiro. Este processo é garantido pelas funções *onChange()*, recebe o ficheiro, *sendFile()*, executa a função *upload()*, responsável pelo upload do nosso ficheiro e criação de uma nova linha, e envia a informação para a DB, que posteriormente será utilizada no mixer.

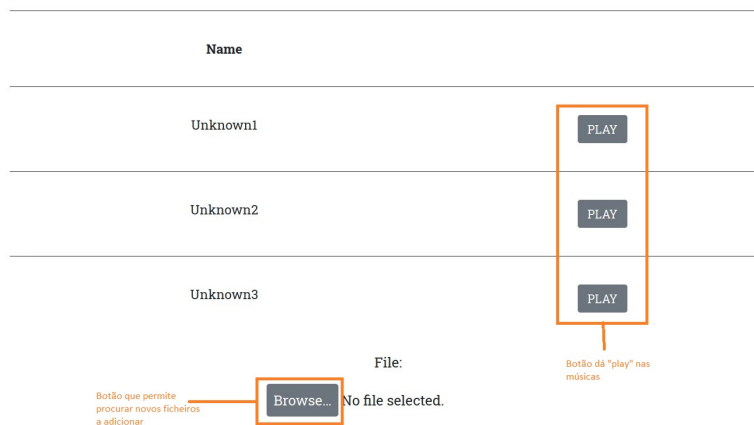


Figura 2.3: Tabela de Excertos

2.4 mix.html

Esta é a página mais importante do nosso projeto, e também a mais trabalhosa, trata-se então de um misturador de sons. No lado esquerdo encontramos várias opções que nos permitem escolher os sons que pretendemos usar (sons estes que já estão listados no `excertos.html`), do lado direito encontram-se os efeitos sonoros disponíveis sendo que, para cada uma deles, existe uma função que o "liga" à música escolhida do lado esquerdo. Estas funções no entanto apenas são aplicadas no servidor através das funções `sendpauta()` e `upload_pauta()` entre cliente e servidor.

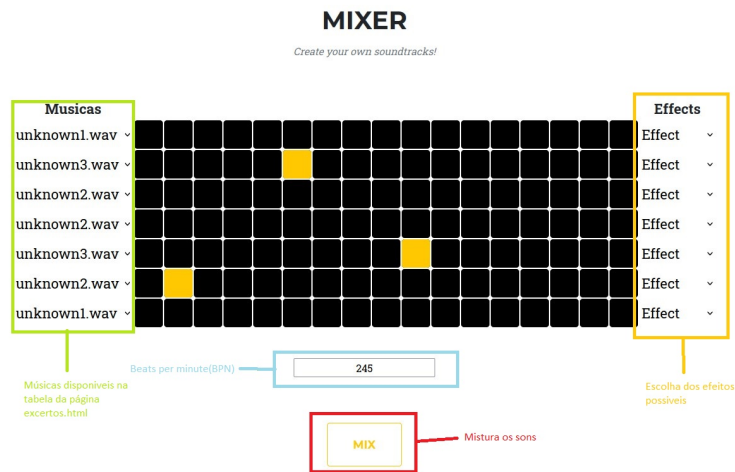


Figura 2.4: Misturador

2.5 about.html

Página web com a apresentação do grupo e as contruibuições que cada um teve para a boa realização do projeto. Contém também o link do repositório no CodeUa e os diferentes recursos usados para a boa realização do trabalho.

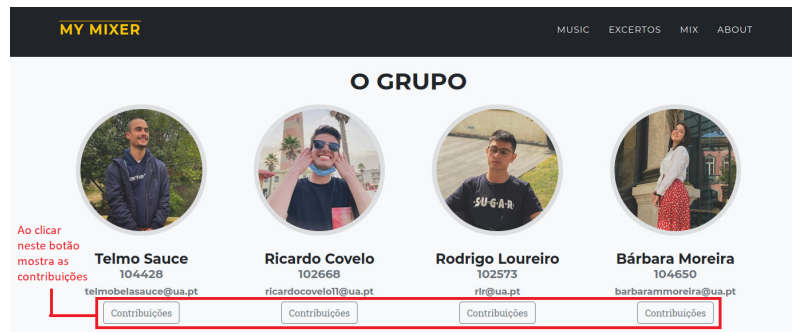


Figura 2.5: O nosso grupo na página about

Capítulo 3

script.js & link.py

Com uma grande importância para o bom funcionamento do nosso trabalho, estes dois ficheiros de código funcionam como "fio" de ligação para todo o restante código e site. Através de um Application Programming Interface (API) é nos possível comunicar entre python e javascript. No *script.js* encontram-se todas as funções em javascript usadas, desde às correspondentes á barra de ferramentas, às responsáveis pelo funcionamento do nosso mixer. O mesmo tem grande importância visto que é o principal responsável por chamar os API e por criar algumas tabelas de forma dinâmica, pois estas vão buscar a informação á base de dados. O *link.py* está portanto responsável por enviar/receber estruturas de dados, como JSON, atualizar a nossa DB e interligar os nossos ficheiros HTML.

Capítulo 4

Base de Dados

É de notar que todas as tabelas presentes nas nossas páginas web são criadas numa função python, *list()*, através da informação implementada na base de dados (*database.execute*). Dependendo do "type" que esta nossa função recebe, a mesma reconhece qual página se trata e origina a nossa tabela conforme os dados fornecidos na database, dados estes que são traduzidos num json e enviados para o javascript e posteriormente passados para HTML.

A função python *vote()* trata de buscar os dados presentes em *people* e *row* até lá guardados na DB e atualizá-los. Posto isto se o valor recebido for de "1" o número de votos no id correspondente aumenta, caso contrário se o valor recebido é -1 e maior que "0", não podem haver votos negativos, o número de votos diminui. Caso nenhuma destas condições se verifique, é enviada uma mensagem de erro. Estes valores são posteriormente atualizados e guardados da nossa DB, como anteriormente referido.

Capítulo 5

Resultados

Apresentaremos agora, algumas imagens que correspondentes aos resultados obtidos neste nosso projeto.

5.1 Funcionamento dos Votos



Figura 5.1: Contagem de votos positivos, "likes"@"



Figura 5.2: Contagem de votos negativos, "dislikes"@"

5.2 Funcionamento do Browse

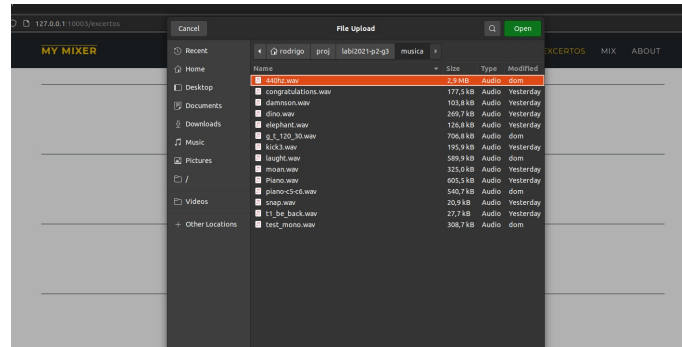


Figura 5.3: Funcionamento do Browse

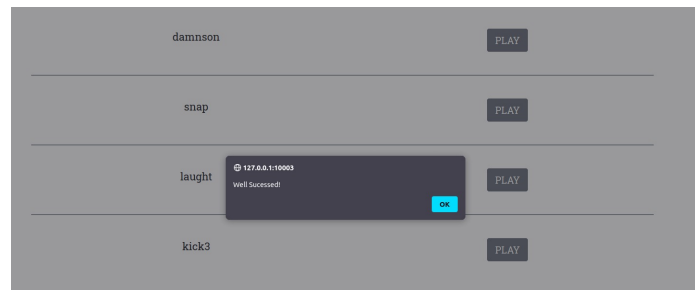


Figura 5.4: Adição do ficheiro



Figura 5.5: Criação da nova row

Capítulo 6

Conclusões Finais

Com este projeto foi possível consolidar ainda melhor os conhecimentos adquiridos ao longo deste primeiro ano nas aulas de LABI, bem como melhorar as nossas capacidades de trabalho em grupo.

Ultrapassámos diversas dificuldades em grupo e cremos ter feito um bom trabalho. Posto isto, e perante o percurso na realização deste mesmo projeto achamos que as percentagens atribuídas a cada membro devam ser as seguintes:

- Telmo Sauce - 25%
- Rodrigo Loureiro - 25%
- Bárbara Moreira - 25%
- Ricardo Covelo - 25%

Acrónimos

LABI Laboratórios de Informática

API Application Programming Interface

HTML HyperText Markup Language

JS Java Script

CSS Cascading Style Sheets

DB DataBase

Bibliografia

- [1] (). «W3 Schools», URL: <https://www.w3schools.com/>.
- [2] (). «Stack Overflow», URL: <https://stackoverflow.com/>.
- [3] (). «GitHub», URL: <https://github.com/>.
- [4] (). «Stack Exchange», URL: <https://stackexchange.com/>.
- [5] (). «Geeks for Geeks», URL: <https://www.geeksforgeeks.org/>.
- [6] (). «MDN Web Docs», URL: <https://developer.mozilla.org/en-US/>.