**Rémi LEFAIVRE**

**Mauve, Music Player Project**

**Version française :**

Bonjour, je suis le responsable du Front End de ce projet. Ce document à pour but d’expliquer les démarches suivies afin d’aboutir à la réalisation de ce projet de mon côté.

Tout d’abord, le graphiste de notre groupe nous a confectionné une maquette sous Photoshop afin d’avoir un visuel idéal sur lequel se basé. Notre but était de nous rapprocher le plus possible de notre idée et nous avons fait au mieux à l’aide de nos connaissances et du temps qui nous était donné.

L’apparence proposée par le lecteur de musique a été codé en FXML, une alternative au HTML qui prend en compte un aspect plus graphique ainsi qu’une librairie prédisposée afin de pouvoir créer l’affichage de notre lecteur. Grâce à *JavaFX Scene Builder 2.0*, j’ai pu faire en sorte d’assembler, à la même manière qu’Arthur sur Photoshop, les éléments qui constituent *mauve*.Afin de pouvoir interagir avec les actions de l’utilisateur, j’ai dû créer des variables correspondant aux différents boutons présents sur l’interface.

Les variables citées précédemment sont utilisées dans la classe FXMLDocumentController qui répertorie toutes les fonctions utiles au bon fonctionnement du lecteur. Différentes fonctions comme *Play, Pause, Previous* ou bien encore *Next* ont été ajouté dans cette classe. En plus de cette classe, le fichier FXMLDocument est agrémenté du code définissant l’apparence du lecteur en utilisant les mêmes bases que le HTML. Ce dernier est également lié à un fichier Style.css qui permet d’aller légèrement plus loin dans la personnalisation de l’interface, et notamment au niveau des boutons (transparence, images…).

De plus, la classe MediaPlayer permet de mettre en place une scène vierge dans laquelle s’affichera le lecteur. Cette classe va rentrer en résonnance avec la classe Client, liée au serveur par le biais de port et d’IP que notre responsable BackEnd, Hugo, a pris soin d’instaurer. Des fonctions telles que le téléchargement de musiques ou de répertoire dans un tableau dynamique y sont prévues.

Grâce à ce lecteur, nous sommes censé pouvoir lire des musiques à travers un serveur de stockage. Par défaut, une musique sera téléchargée dans le dossier de ce projet intitulé MediaPlayer2 (ne surtout pas renommer ce dossier !).

Sachant que le serveur est hébergé par notre responsable BackEnd, Hugo MATH, et il faudra que celui-ci le mette à disposition si vous voulez essayer notre projet, vous pouvez pour cela le contacter !

Notre équipe vous souhaite de faire bon usage *mauve* et on espère qu’il vous plaira ! Merci de votre confiance.

English Version :

Hi, I’m in charge of the FrontEnd of this project. This document is created to explain every single steps we made to conclude to the realization of this project by my side.

First, our graphist built for us a prototype with Photoshop in order to have an ideal visual on which we could have focused on. Our goal was to get as close as possible to this idea and we worked the best we could with our knowledges and the time that was given.

The apparence showed by the music player was coded in FXML language, analternative to HTML language which count in his library a graphic aspect and a built-in package in order to create our player’s form. Thanks to *JavaFX Scene Builder 2.0*, I managed to build it as easy as we can do with Photoshop and then I set up every single element to build mauve. In order to interact with the user’s actions, I had to create variables. One variable was linked to one button present on the interface.

The variables I talked about upper are used in the class FXMLDocumentController which is filled with all the functions used to make the player working good. There are different functions like Play, Pause, Previous or even Next that were added to this class. We also have the file FXMLDocument which, in his code, defines the player’s apparence by using same basics as HTML language. This last one is also linked to a Style.css file which allows us to go a little bit further in the customization of the interface and particularly with buttons (transparency, pictures…).

Moreover, the MediaPlayer class create a blank scene which will be filled with the player and will show it. This class will be connected to the Client class linked to the server through port and IP address that our BackEnd responsible built. Functions such as downloading musics or create a library to put musics in are prepared.

Thanks to this player, we are supposed to read some musics through a storage server. By default, a music will be downloaded in the project’s folder called MediaPlayer2 (Do not rename this folder !).

The server is hosted by Hugo MATH, the person in charge of the BackEnd and he will need to set the server up in order to make the project working correctly and you can contact him for that !

Our team hopes that you will use mauve as well as you can and we hope that you will like it ! Thanks for your confidence.