

Manuel d'installation

Synthèse vocale de numéros téléphoniques

Equipe:

CUMMINGS Thibaud
GOLETTA Michael
SOLEYMANKHANI Hossein
DIALLO Négué

Table de Révision

Révision	Date	Auteur	Modifications apportées
1	10 Avril 2010	T. CUMMINGS	Version initiale
2	29 Avril 2010	M. GOLETTA	Version finale

Table des matières

1	INTRODUCTION	3
1.1	OBJECTIFS ET METHODES.....	3
1.2	DOCUMENTS DE REFERENCE	3
2	INSTALLATION DE LA LIBRAIRIE	4
2.1	INSTALLATION SUR LA PLATEFORME	4
2.1.1	Pré requis	4
2.1.2	Eclipse	4
2.2	INCLUSION DANS LE CODE SOURCE	5
3	LANCEMENT DE L'APPLICATION DE TEST	7
4	GLOSSAIRE	8
5	REFERENCES.....	8
6	INDEX	9

1 INTRODUCTION

Ce document énonce les procédures nécessaires à la mise en place de la librairie dans l'environnement de développement et à son bon fonctionnement. Pour l'application de test, le document fournira les étapes à suivre pour une bonne exécution de celle-ci.

1.1 Objectifs et méthodes

Il est demandé de livrer une librairie codé en langage C permettant la synthèse vocale de numéro téléphoniques ainsi qu'une application qui teste cette librairie. La librairie qu'il est demandé de fournir doit comporter une fonction permettant la reconstruction vocale d'un numéro téléphonique. Afin de restituer vocalement des numéros de téléphones, il est nécessaire d'implémenter un algorithme de synthèse vocale. A partir d'une série de nombres en entrée, l'algorithme devra être capable de générer le fichier son correspondant à ces nombres par une voix reconstituée.

1.2 Documents de référence

Le deux documents de référence ici sont le cahier des charges (S1), le cahier de recette (S2) et le plan de développement (S3)

2 INSTALLATION DE LA LIBRAIRIE

2.1 Installation sur la plateforme

Pré-Requis :

- Le programme n'est pas supporté par les plateforme 64 Bits
- Le manuel d'installation est spécifique aux distributions linux (la démarche spécifique pour Ubuntu est détaillée).

2.1.1 Pré requis

Pour pouvoir profiter pleinement de la librairie Synthèse_Vocale utilisant « Simple DirectMedia Layer » (plus connu sous le nom de SDL), vous devez installer les paquets suivants : **libsdl1.2debian**, **libsdl1.2debian-all** et **libsdl1.2-dev**.

Si vous êtes sous Ubuntu, lancer le Terminal : Applications > Accessoire > Terminal, Puis rentrer les lignes suivant (votre mot de passe vous sera demander afin de permettre l'installation) :

```
sudo apt-get install libsdl1.2debian  
sudo apt-get install libsdl1.2debian-all  
sudo apt-get install libsdl1.2-dev
```

(**N.B** : Le paquet libsdl1.2-dev est nécessaire que si vous souhaitez utiliser la librairie dans un environnement de développement. Vous pouvez ne pas l'installer si vous ne désirez que tester l'application de test.)

Une fois ces paquets installés, vous pouvez quitter le Terminal.

2.1.2 Eclipse

Si vous avez déjà Eclipse installé, passez directement à l'étape 2.2.

Rendez vous sur <http://www.eclipse.org/downloads/> et téléchargez la version compatible avec votre système de « Eclipse IDE for C/C++ Developers (79 MB) » (Eclipse CDT).

Procédez à l'installation en suivant les consignes fournit sur le site.

2.2 Inclusion dans le code source

1. Lancez maintenant Eclipse (sous Ubuntu, Applications > Programmation > Eclipse).
2. Créez un nouveau projet C (File > New > C Projet), puis rentrez le nom désiré pour votre projet, sélectionnez « Empty Projet » et « Linux GCC ».
3. Créez un nouveau fichier source : (File > New > Source File), dans « source folder », mettez le nom du projet précédent crée, et dans « Source File » mettez le nom que vous désirez avec l'extension .c (exemple main.c).
4. À l'aide de votre explorateur de fichier (Nautilus sous Ubuntu), copiez l'ensemble des fichiers situé dans Application > Code du CD-ROM, puis retournez sous Eclipse et collez-les dans le projet.
5. Il faut maintenant lier la librairie SDL au projet. Pour cela, faites clic droit sur la projet, puis *Proprieties*, dans le sous menu *C/C++ Build*, sélectionnez *Setting*. Dans *Librairies* de *GCC C Linker*, ajoutez *SDL* à *Librairies (-l)* et le répertoire de votre librairie SDL (sous Ubuntu : */usr/lib*) dans *Library search path (-L)*
6. Dans votre fichier source, afin d'inclure les sources de la librairie, insérez en début du fichier la ligne :

```
#include «Lib_Synthese/Synthese_Vocale.h»
```

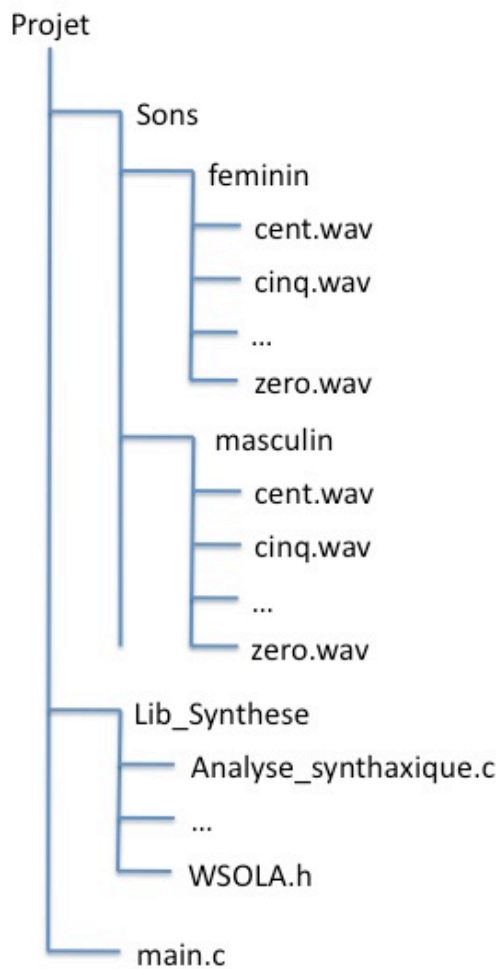
7. Pour utiliser la fonction dans votre fonction main, ajoutez la ligne :

```
Synthese_Vocale();
```

Aucun paramètre n'est nécessaire, cette fonction se chargera elle-même de demander les informations tels que le numéro, la voix et la vitesse nécessaires au bon fonctionnement du programme.

7. Pour lancer le programme, vous devez le compiler puis l'exécuter.
Pour compiler, cliquez sur le menu déroulant Project, puis Build.
Pour l'exécuter, cliquez sur Run puis Run.
Afin de voir l'exécution du programme, allez dans l'onglet Console d'Eclipse.

Exemple de programme et de son architecture :



```
// On inclue la librairie.
#include "Lib_Synthese/Synthese_Vocale.h"

int main(int argc, char *argv[]){

    // Appel à la fonction de la librairie.
    Synthese_Vocale();

    return 0;
}
```

3 LANCEMENT DE L'APPLICATION DE TEST

Pour les distributions Linux:

1. Localiser le répertoire où se situe l'application de test (en général, clic droit puis propriété vous donnera les informations nécessaires).
2. Ouvrez un Terminal (sous Ubuntu, Applications puis Accessoire et enfin Terminal). Entrez la ligne suivante :
`cd Répertoire/application/de/test`
3. Exécutez l'application grâce à la ligne (test est le nom de l'application) :
`./test`

4 GLOSSAIRE

-C-

CDT : Acronyme ayant pour référence le plugin d'*Eclipse* (*C/C++ Development Toolkit*)

-E-

Eclipse : Environnement de développement intégré (IDE) libre, extensible et polyvalent (supporte plusieurs langages).

-N-

Nautilus : Explorateur de fichier par défaut sous Ubuntu

-S-

SDL : Simple DirectMedia Layer, librairie multimedia.

-U-

Ubuntu : Système d'exploitation basé sur *Debian*, une des nombreuses distributions *GNU/Linux*

5 REFERENCES

Cahier des charges *S1*

Cahier de recette *S2*

Plan de développement *S3*

Documentation interne du code *R2*

6 INDEX

CDT, 4

Eclipse, 8

Librairie, 3, 9

SDL, 3

Ubuntu, 4, 7