

Cahier des charges

Synthèse vocale de numéros téléphoniques

Equipe:

CUMMINGS Thibaud
GOLETTA Michael
SOLEYMANKHANI Hossein
DIALLO Négué

Table de révision

Revision	Date	Auteur	Modifications apportés
1	12 Février 2010	T. CUMMINGS	Copie initiale
2	13 Février 2010	M. GOLETTTO	Ajout Glossaire, Références
3	14 Février 2010	N. DIALLO	Ajout Index
4	17 Février 2010	T. CUMMINGS	MAJ de l'analyse fonctionnelle

Table des matières

1	INTRODUCTION.....	3
1.1	CONCEPTS DE BASE	3
1.2	CONTEXTE	3
2	DESCRIPTION DE LA DEMANDE	4
2.1	OBJECTIFS.....	4
2.2	ANALYSE FONCTIONNELLE	4
3	CONTRAINTES	5
3.1	CONTRAINTES DE DELAIS	5
3.2	CONTRAINTES D'ENCADREMENT	5
3.3	CONTRAINTES TECHNIQUES	5
4	DEROULEMENT DU PROJET	6
4.1	PLANIFICATION.....	6
4.2	RESSOURCES.....	6
4.3	ORGANISATION	6
5	DESCRIPTION DETAILLE DU PROJET	7
6	GLOSSAIRE	9
7	REFERENCES.....	9
8	INDEX	10

1 INTRODUCTION

1.1 Concepts de base

La synthèse vocale aussi connue sous le nom de "Text-to-speech" (TTS) est une technique informatique de reproduction artificielle de la voix humaine; elle permet par des procédés de synthèse sonore de créer de la parole artificielle à partir de n'importe quel texte. Cela permet ainsi à l'ordinateur de prononcer de manière intelligible et agréable un texte enregistré sous forme de fichier ou entré directement au clavier.

Dans notre situation, nous utiliserons la synthèse vocale de numéro téléphonique dans le domaine des serveurs vocaux. Aujourd'hui, les serveurs vocaux sont utilisés dans la téléphonie professionnelle, la communication, l'industrie, les animations télévisées... l'exemple le plus commun serait celui des répondeurs "vous êtes sur le répondeur du...".

Le projet est limité à la restitution de numéros téléphoniques.

1.2 Contexte

Dans le cadre d'études en informatique à l'Université Paris Descartes, nous avons constitué une équipe de quatre personnes afin de développer une application qui nous a été proposée. Il est demandé de réaliser une application de serveur vocal de renseignements téléphoniques. Le module projet de programmation permet d'acquérir de l'expérience en matière de développement d'application. Cela permet de dépasser le stade des travaux pratiques et met en valeur le travail d'équipe.

De plus, l'équipe de développement est confrontée à la demande d'un client dont elle doit satisfaire les besoins. Dans notre situation, c'est en fait l'encadrant qui joue le rôle du client. Il s'agit donc d'une vraie mise en situation en entreprise ou nous participerons à toutes les phases de développement d'un projet.

2 DESCRIPTION DE LA DEMANDE

2.1 Objectifs

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, nous devons 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.

2.2 Analyse Fonctionnelle

- disposer d'une fonction permettant la synthèse des numéros
 - F1.1:** lire une chaîne d'entrée \Leftrightarrow numéro
 - F1.2:** segmenter numéro par défaut selon norme française: par paires
 - F1.3:** utiliser des séparateurs pour segmenter les numéros en groupes de 1 à 4 chiffres
 - accepter le caractère "+" seulement en début de chaîne
 - utiliser les caractères " , . - " comme séparateurs et seulement ceux-ci
 - limiter cette segmentation à 4 chiffres maximum
 - F1.4:** proposer une voix masculine ou féminine
 - F1.5:** proposer trois vitesses de lecture
 - F1.6:** jouer malgré caractères non acceptés en entrée
- fournir des messages à l'application
 - F2.1:** fournir des messages d'erreur
 - F2.2:** fournir des messages d'avertissement
 - F2.3:** renvoyer un message de fin de synthèse

3 CONTRAINTES

3.1 Contraintes de délais

Les recettes ont lieu au terme des douze semaines, soit la semaine du 26/04/2010.

3.2 Contraintes d'encadrement

Des réunions hebdomadaires d'environ vingt minutes sont programmées. Elles ont pour but de donner l'avancement du projet au client, de confirmer ses attentes et d'organiser les réunions suivantes. Elles donneront systématiquement lieu à la rédaction de comptes-rendus déposés sur Plenadis dans les 24h qui suivent la réunion.

Un minimum de 6 heures de travail hebdomadaires est imposé, les horaires de travail doivent être affichés sur la plate-forme Plenadis chaque semaine.

3.3 Contraintes techniques

La librairie doit être une librairie compilable entièrement codé en langage C, c'est-à-dire au format *librairie.h*. Celle-ci doit être compatible, en priorité sur les systèmes d'exploitation linux (toutes distributions courantes).

La plateforme d'exécution/d'utilisation de la librairie est donc un micro-ordinateur muni d'une des distributions courantes de linux.

La fonction de la librairie étant très simple, elle ne nécessite pas une machine très puissance, symboliquement nous nous contenterons de définir une configuration minimale, un Pentium 4 (2003) avec 512Mo de RAM.

Par contre il est évident que la machine devra être composé d'un carte son capable de lecture de sons de qualité audio compressés au format .wav, échantillonnés à 44,100 kHz.

Aucune contraintes de performances ou d'optimisation d'utilisation du système d'exploitation ou de la machine n'ont été imposés.

La librairie utilisée pour la gestion du son doit être libre pour une exploitation du programme sans frais.

Un gestionnaire de sources SVN devra être utilisé par l'équipe afin de partager le même code et pour une meilleure coordination.

4 DÉROULEMENT DU PROJET

4.1 Planification

Spécification (25 Janvier - 1er Mars)

- définition objectifs/besoins avec client
- rédaction cahier des charges
- rédaction cahier des recettes
- rédaction plan de développement

Développement (1er Mars - 5 Avril)

- conception/architecture
- spécification des modules
- développement de chaque module en // enregistrements vocaux
- développement de l'environnement de test
- test unitaires de chaque module
- intégration
- rédaction des manuels d'installation/utilisation + autres documents

Validation (5 Avril – 12 Avril)

Livraison (12 Avril)

- produit
- recettes avec client

4.2 Ressources

Trois applications seront majoritairement utilisées lors du projet:

Eclipse : environnement de développement intégré pour la conception de la librairie et de l'application de test.

Google documents : logiciel en ligne permettant un travail collaboratif sur des documents textes.

DropBox : service de stockage et de partage de fichiers en ligne utilisé comme dossier collaboratif/partagé.

4.3 Organisation

CUMMINGS Thibaud (*chef de projet*)

GOLETTO Michael (*chef de développement*)

SOLEYMANKHANI Hossein (*enregistrement*)

DIALLO Négué (*documentation*)

Chaque membre de l'équipe travaille sur différents modules composant la fonction de la librairie.

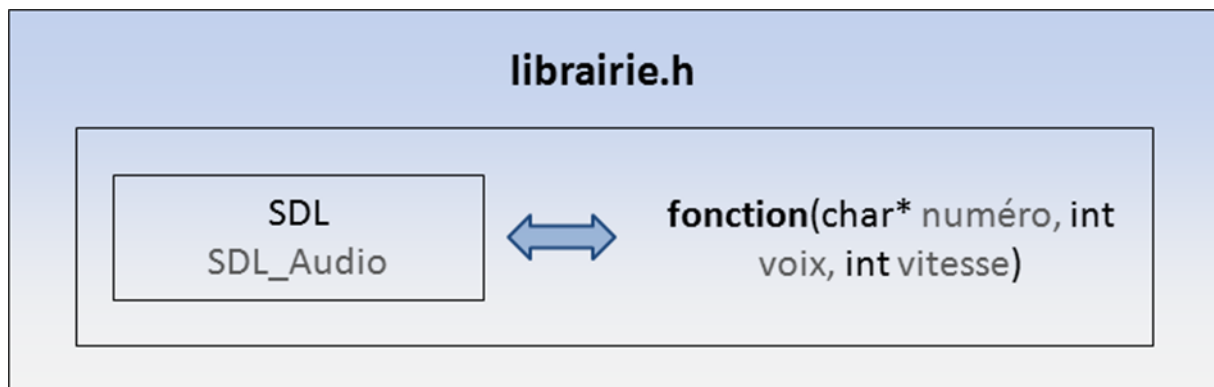
5 DESCRIPTION DÉTAILLÉ DU PROJET

Librairie

La librairie ne contient qu'une seule fonction, le projet n'est donc composé que d'un seul module

Application

L'application de test utilisera donc la seule fonction de la librairie



```
include<stdio.h>
include<librairie.h>
...

void main() {
    ...
    void fonction(char* numéro, int voix, int vitesse);
    ...
}
```

Three vertical arrows point from the function signature in the code block to the corresponding parameters in the function signature shown in the diagram above: **numéro**, **voix**, and **vitesse**.

La seule fonction disponible dans la librairie a pour but de synthétiser vocalement un numéro entrée par l'utilisateur.

En entrée, la fonction prend en compte trois variables :

- une chaîne de caractères correspondant au numéro à synthétiser
- un entier correspondant à la voix voulu
- un entier correspondant à la vitesse de lecture

La chaîne est limitée à 255 caractères (ANSI) et n'acceptera que les chiffres de 0 à 9 ainsi de 5 autres caractères : le point « . », la virgule « , », l'espace « », le tiret « - » et le signe mathématique plus « + ».

Tout autre caractère et caractères suivant le 255^{ième} sera supprimé lors de la procédure/analyse de la chaîne et ne sera donc pas lu/synthétisé. ex: 063r85hg987/45 sera lu comme 06 38 59 87 45.

Un retour à la ligne *Entrée* détermine la fin de la saisie.

Des messages d'erreurs seront affichés à l'écran lors de présence d'un caractère non accepté ou d'une longueur de chaîne supérieur à 255 caractères.

Malgré les erreurs de saisie la fonction tente de lire/synthétiser le numéro entré.

Une succession non séparée de chiffres sera interprétée par paires (norme française) c.à.d. 0352398579 sera lu comme 03 52 39 85 79.

Les caractères « , » « . » « - » « » ont pour utilité d'assurer des regroupements de chiffres. Chacun de ceux-ci sont donc des séparateurs pour déterminer les regroupements voulus.

Ces regroupements ne sont possibles que par groupes de 1 à 4, pas plus.

Quelconque regroupement dépassant 4 caractères sera donc interprété par paires (interprétation par défaut). ex : 09 5678964 => 09 5 67 89 64

Pour éviter les confusions et minimiser les risques d'erreurs il y a possibilité d'utiliser des caractères différents dans la même chaîne. ex: 01-586-9342-52 01.586.9342.52 01,586,9342,52 01 586 9342 52 et 01-586.9342,52 seront lus comme "zéro un. cinq-cents quatre-vingt six. neuf mille trois cent quarante deux. cinquante deux". Cela dit il est conseillé d'utiliser un seul séparateur à la fois.

Le plus « + » n'est utilisable qu'en début de chaîne et sera lu tel qu'il est. Tout caractère plus « + » placé ailleurs qu'en première position de la chaîne ne sera pas reconnu et donc pas lu/synthétisé.

L'entier déterminant la voix utilisé prend ses valeurs dans 0 et 1. Les deux voix possibles sont une voix féminine (0) et une voix masculine (1).

La vitesse de lecture du numéro est également déterminée par un entier. Les trois vitesses possible correspondent au rythme de lecture du numéro: lent (1), normal (0), rapide (2).

Ces deux choix sont unitaires; il est évident qu'il est impossible de choisir plusieurs vitesses ou les deux voix en même temps.

Un message d'affichage à l'écran est émis une fois la synthèse du numéro correctement exécuté.

6 GLOSSAIRE

-M-

Mo : Méga-Octet, unité de stockage informatique.

-R-

RAM : Random Access Memory, mémoire vive.

-S-

SDL : Simple DirectMedia Player est une bibliothèque libre permettant la gestion d'évènements.

SVN : Subversion, système de gestion de versions au cours du développement.

-W-

wav : extension de fichier audio, il permet de contenir des formats variés

7 REFERENCES

Livres

Le langage C Norme ANSI de *Brian W. Kernighan* et *Dennis M. Ritchie*

Documentation en ligne

Explication théorique de la gestion du son par SDL (anglais) :

<http://osdl.sourceforge.net/main/documentation/rendering/SDL-audio.html>

Gestion du son par SDL (français) : <http://loka.developpez.com/tutoriel/sdl/sons/>

8 INDEX

Application, 7

Développement, 6

Descartes, 3

DropBox, 6

Eclipse, 6

Google Documents, 6

Langage C, 5, 9

Linux, 5

Livraison, 6

Librairie, 7

Plenadis, 5

SVN, 5

SDL, 7

Synthèse vocale, 3 , 4

Serveur vocal, 3

Spécifications, 6

Text-to-speech, 3

Validation, 6