

# Cahier des charges

## Parler avec son ordinateur

Sébastien PEDREAU  
Christian INGOUFF

Année 2012/2013  
Semestre 3

# Sommaire

<b>1</b>	<b>Origine et nature du projet</b>	<b>2</b>
1.1	Origine et présentation du sujet . . . . .	2
1.2	Objectifs . . . . .	2
1.3	Phases du projet . . . . .	3
1.3.1	Proposition des sujets - Septembre/Octobre 2011 . . . . .	3
1.3.2	Recherche théorique - Novembre 2011/Juin 2012 . . . . .	3
1.3.3	Rédaction et réalisation - Septembre/Janvier 2012 . . . . .	3
<b>2</b>	<b>Expression fonctionnelle du besoin</b>	<b>4</b>
2.1	Expression du besoin . . . . .	4
2.2	Fonctions de services et fonctions de contraintes . . . . .	5
2.2.1	Préface . . . . .	5
2.2.2	Fonctions de services . . . . .	5
2.2.3	Fonctions de contraintes . . . . .	5
2.3	Maquette . . . . .	6
2.3.1	Structure . . . . .	6
2.3.2	Fonctionnement . . . . .	7
2.4	Critères d'appréciation . . . . .	8
<b>3</b>	<b>Ressources</b>	<b>9</b>
3.1	Ressources humaines . . . . .	9
3.2	Ressources financières . . . . .	9
3.3	Ressources informatiques . . . . .	9
<b>4</b>	<b>Conclusion</b>	<b>10</b>

# Chapitre 1

## Origine et nature du projet

### 1.1 Origine et présentation du sujet

L'idée de parler avec les machines remonte très loin dans l'histoire, jusqu'aux mythes de la Grèce antique. L'homme semble toujours avoir voulu créer une sorte d'alter-ego mécanique. Ce projet, bien qu'un peu utopique, semble de plus en plus se préciser dans notre époque moderne, que ce soit dans l'esprit des littéraires (A travers des films comme "Metropolis" ou des livres), ou celui des scientifiques (Les derniers automates réalisés sont de plus en plus perfectionnés).

C'est à travers cette remarque que nous avons trouvé le sujet de notre TIPE : "Parler à son ordinateur". Que se passerait-il si notre machine nous répondait ? Si l'on pouvait tenir une conversation, plus ou moins cohérente, avec notre ordinateur ?



FIGURE 1.1 – Asimo, automate humanoïde

### 1.2 Objectifs

Pour cette seconde année de travail, notre objectif sera donc de faire parler notre ordinateur. Pour cela, nous allons coder un bot informatique destiné à communiquer avec l'utilisateur, c'est-à-dire un chatbot (aussi appelé un chatterbot). Nous le coderons dans le langage "turbo-pascal", et tenterons de le programmer afin qu'il ait une discussion aussi fluide que possible.

## **1.3 Phases du projet**

### **1.3.1 Proposition des sujets - Septembre/Octobre 2011**

Au début de l'année scolaire 2011, nous avons créé deux sujets dans le cadre du Travail d'Initiative Personnel Encadré (TIPE), et nous les avons proposés au corps enseignant de notre établissement. Le sujet ainsi choisi fut "Parler a son ordinateur", tandis que Yannick Le Nir nous fut assigné en tant que professeur référent. Grâce à ses recherches et ses connaissances sur l'Intelligence Artificielle, son aide allait nous être précieuse.

### **1.3.2 Recherche théorique - Novembre 2011/Juin 2012**

La première année scolaire était destinée à des recherches théoriques. Pour notre sujet, ces recherches furent dans un premier temps orientées vers l'Intelligence Artificielle, ainsi que les bots informatiques, plus spécialement les bots de conversation, appelés "Chatbot". Ensuite, nous nous sommes penchés sur des notions de reconnaissance vocale et de synthétisation de la voix. En effet, bien que le projet soit ambitieux, peut être pourrions-nous ajouter des systèmes de reconnaissance vocale et de synthétisation de la voix à notre logiciel.

### **1.3.3 Rédaction et réalisation - Septembre/Janvier 2012**

Lors de la seconde année du TIPE, l'objectif était la création d'un poster explicatif concernant notre sujet, ainsi qu'un cahier des charges complet de la réalisation de notre projet. Dans notre cas, le cahier des charges portera donc sur le fonctionnement de notre chatbot, sur ses fonctions de service et ses fonctions contraintes, ainsi qu'un bref manuel d'utilisation.

## Chapitre 2

# Expression fonctionnelle du besoin

### 2.1 Expression du besoin

Voici un premier diagramme de type "bête à cornes" pour l'expression du besoin. Il permet d'aborder le sujet et de le décrire rapidement.

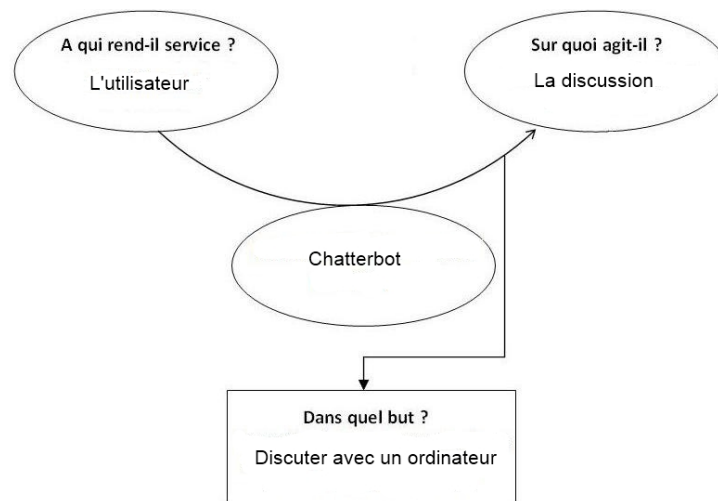


FIGURE 2.1 – Diagramme "bête-à-cornes"

## 2.2 Fonctions de services et fonctions de contraintes

### 2.2.1 Préface

Nous traiterons dans cette partie les modalités de notre réalisation, à travers la fonction de services et les fonctions contraintes. La fonction de service (qui est ici la fonction principale) est la raison qui nous pousse à créer notre logiciel, tandis que les fonctions contraintes servent à montrer les limites et les nécessités qui seront à prendre en compte lors de la réalisation de notre projet.

### 2.2.2 Fonctions de services

Le tableau suivant résume les différentes composantes de la fonction principale 1, à savoir la discussion avec l'utilisateur.

Fonctions de services	Fonctions techniques	Solutions techniques
Discussion (FP1)	Avoir une discussion fluide	Algorithme pour analyser la phrase et trouver la meilleure réponse
	Avoir systématiquement une réponse	Base de données exhaustive couvrant un maximum de questions

### 2.2.3 Fonctions de contraintes

Le tableau suivant résume les contraintes de la fonction principale 1.

Fonctions de contraintes	Critères	Niveaux
Facile d'utilisation (FC1)		Être facile d'utilisation pour un utilisateur lambda
Fonctionnement (FC2)	Operating System (OS)	Fonctionne sous les différents OS utilisés
Esthétique (FC3)		Être agréable visuellement pour un utilisateur lambda
Codage (FC4)	langage turbo-pascal	Être codé dans un langage informatique qui nous est connu

## 2.3 Maquette

Dans cette partie, nous présenterons la structure et le fonctionnement de notre chatbot.

### 2.3.1 Structure

Voici une image présentant la structure du logiciel lors de son utilisation :



FIGURE 2.2 – Exemple du logiciel en cours d'utilisation

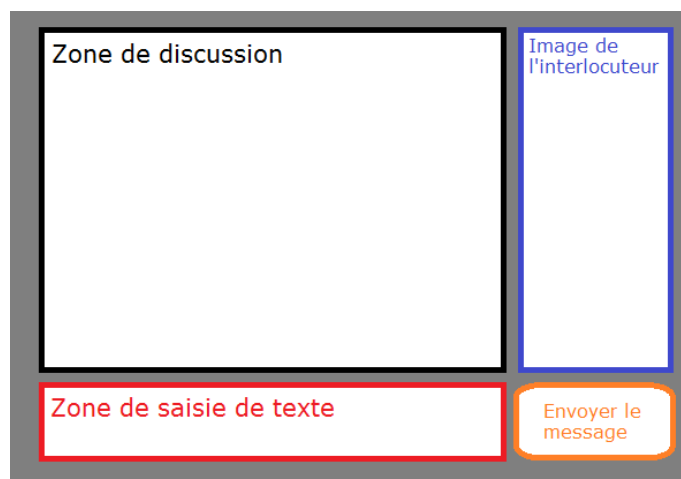


FIGURE 2.3 – Maquette du logiciel

### 2.3.2 Fonctionnement

L'utilisation du logiciel amène d'abord l'utilisateur sur une page menu permettant de lancer une discussion. Cette page de menu permet de sélectionner le niveau de conversation (qui déterminera l'algorithme de réponse ainsi que l'image de l'interlocuteur machine), avant de lancer la conversation.

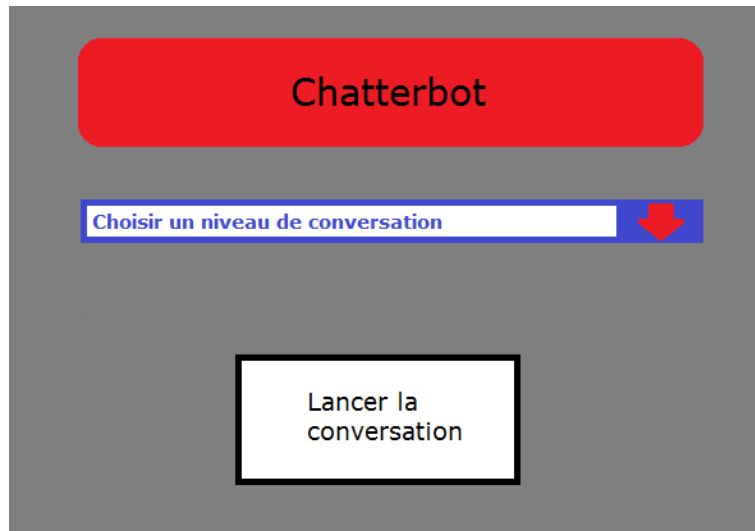


FIGURE 2.4 – Maquette du menu

Pour converser avec la machine, il suffit de rentrer dans la zone de saisie de texte, par l'intermédiaire de l'interface clavier, une phrase. L'ordinateur analysera cette phrase. En fonction de certains mots-clés, du type de question (phrase interrogative par exemple), il formulera une réponse selon un algorithme, celui-ci étant choisi dans une liste d'algorithme prédéfinie par l'utilisateur, lorsque ce dernier est sur le menu.

Nous parlons ici de système expert, c'est-à-dire un système capable de reproduire les mécaniques de la pensée d'un expert (en l'occurrence, un expert en conversation). Ce système expert est constitué de l'interface utilisateur présentée plus tôt, d'une base de connaissances, ainsi que d'un moteur d'inférence, à savoir un algorithme censé reproduire le comportement d'un "expert" au cours d'une conversation. L'algorithme sera choisi parmi les suivants :

Algorithme du psychologue : Selectionne une phrase aléatoirement parmi des phrases pré-enregistrées présentes dans la base de connaissances (remplie préalablement par les développeurs). Ces phrases sont volontairement vagues (comme "pouvez-vous préciser votre pensée ?") pour répondre à un maximum de phrases.

Algorithme par mots-clés : Dès que l'algorithme reconnaît un mot-clés présent dans sa base de connaissances (préalablement remplie par les développeurs), il associera une phrase qu'il connaît en rapport avec le mot en question. Si l'utilisateur lui demande son film préféré, le logiciel reconnaîtra "FILM", "PREFERE" ainsi que le type de phrase interrogative, et ira chercher dans sa base de connaissance la phrase pré-enregistrée : "Mon film préféré est Robin des bois.". Cet algorithme nécessitera donc une base de connaissances particulièrement conséquente.

Algorithme par apprentissage (facultatif) : L'algorithme possède une base de connaissances limitée à la base, et l'utilisateur, au fur et à mesure des discussions, l'aidera à apprendre de nouvelles choses rien qu'en discutant avec lui, l'ordinateur posant des questions pour associer les phrases entre elles, tentant de faire des liaisons avec des phrases qu'il connaît déjà. Cependant, nous ne savons pas encore si nous avons les capacités de réaliser un tel algorithme, aussi cette partie du projet nécessite de plus amples recherches, et ne sera peut être pas réalisée.

Une fois la phrase réponse constituée, il l'affichera dans la zone de conversation, et l'utilisateur pourra alors entrer une nouvelle phrase.

## 2.4 Critères d'appréciation

Le projet devra donc fonctionner sous les principaux Operating System des utilisateurs, à savoir les versions les plus récentes de windows (à partir de windows vista), ainsi que la version de linux utilisée sur les ordinateurs de l'EISTI.

La fonction principale de ce projet reste la discussion avec un utilisateur, à travers l'interface clavier.

Le logiciel produit doit être simple d'utilisation pour un utilisateur lambda. Bien que ce soit une vision assez subjective, ce critère sera jugé sur un jury externe au projet.

La discussion de l'ordinateur peut être limitée, mais la machine doit tout de même répondre systématiquement à chaque phrase entrée par l'utilisateur.

## Chapitre 3

# Ressources

### 3.1 Ressources humaines

Ce projet a nécessité la participation de deux étudiants de l'EISTI en cycle préparatoire d'école d'ingénieurs, INGOUFF Christian et PEDREAU Sébastien, ainsi que du soutien de leur professeur référent, Yannick le NIR, enseignant de l'EISTI. Au cours de nos travaux, nous avons bénéficié d'une interview avec Fabrice MALFRERE, le directeur en recherche d'une entreprise se spécialisant dans la synthèse vocale.

### 3.2 Ressources financières

La programmation de ce logiciel, ainsi que les recherches s'y rattachant ne nous ont demandés aucune contribution financière.

### 3.3 Ressources informatiques

Les logiciels nous ayant permis de mener à bien notre projet nous ont été fournis par l'établissement. Le cahier des charges ainsi que le poster de présentation ont été écrits sous le logiciel de traitement de texte GEDIT.

## Chapitre 4

# Conclusion

Cette première approche dans la conception et la réalisation nécessaire à notre TIPE nous a permis de mettre en place un certain nombre de fonction, plus ou moins importantes pour le fonctionnement. La rédaction de ce cahier des charges nous a permis de poser des limites claires, des normes et des principes qu'il nous faudra respecter lors de la prochaine réalisation.