Développement d'une Application d'Assistance aux Personnes Agées pour le Robot Humanoïde NAO

Mohamed Adibe Chemaou, Xhersika Kenga,Sebastien Roach, Marie Legrand

Mai 2017

Sommaire

1 Introduction

2 Le contexte et problème

3 La solution

3.1 Définition des scénarios

3.2 Architecture global du système

3.3 Réalisation scénario d'interaction

3.3.1 Appel météo

Ce scénario était un des premiers scénarios réalisés. Le principe de ce fonctionnalite est très simple : Chaque fois que la personne âgée demande à NAO de savoir les conditions métrologiques courantes, NAO lui donne toutes les informations nécessaires. En plus il donne des conseils pour la manière dont la personne doit s'habiller, par rapport au météo.

L'aspect technique de ce scénario est le suivant :
Dans un box Python, on lit un fichier json qui contient toutes les donnes de météo courantes. Ce fichier est lu depuis un service météo en ligne , l'OpenWeatherAPI. En premier lieu, Avant d'avoir commencé réaliser notre code, on a fait une inscription sur le site du notre fournisseur des donnes. Après cette inscription on récupère un user id qui est indispensable pour les prochaines étapes de réalisation.

Dans un premier temps a cree une fonction $url_builder(cityid)$. Dans cette fonction on fait la génération d'une URL valide pour accéder nos donnes météo qui se trouve dans le site du OpenWeatherAPI. Notre lien complet pour accéder au fichier json est constitué dans un premier temps de l'adresse api, ou on ajoutele mode de récupération des donnes(dans notre ça jSon) et notre user id.

Apres avoir généré notre URL, il était nécessaire de récupérer notre fichier jSon .Pour faire cela a cree une fonction $data_fetch(URL)$, qui prenant en paramètre l'URL de notre fichier, retourne toutes les donnes que ce fichier contient. Les modules python nécessaire pour réaliser cette tâche étant : json et urllib2.
On utilise une fonction du module json appele json.loads qui servent à decoder notre fichier json.

La prochaine étape de réalisation de notre scénario estait d'organiser toutes les donnes brut recupere de notre fichier json, pour ensuite être capable à les transmettre à notre robot, de manière qu'il puisse les prononcer. La fonction qui sert à faire l'organisation des données et $data_organizer$ qui prends en paramètre les donnes retournées par notre fonction précédent $data_fetch$. Il s'occupe de la création d'un dictionnaire python, qui a comme clés la ville, température maximale, minimale etc. Il renvoyait à la fin ce dictionnaire.

L'étape prochaine, qu'on peut qualifier comme étant la plus essentielle, consiste à appeler une fonctionnalite de NAO et Coreographe qui ete très importante et frequentment utilise dans notre projet, le module ALTextToSpeech. Il renvoyait du texte au moteur d'articulation de NAO. Pour utiliser ce module on utilise une ligne très simple : (exemple) tts = ALProxy("ALTextToSpeech", "127.0.0.1", 49905). On definit cette variable après nos fonctions de classe. Elle est appele après dans notre fonction.
Notre fonction $data_output$ prend alors en entree notre dictionnaire cree precedamment. On utilise la variable tes qu'ont appele avec tts.say, la fonction du module ALTextToSpeech qui prend en paramètrer un texte et traduit ce dernier en discors de notre robot.

Dans notre fonction de classe $onInput_onStart$ on appele :(data organizer(data fetch(url builder()))). Cette ligne permet de lier toute la fonction cree dans notre petit box python. Et après voilà, NAO et capable de dire clairement la météo quand la personne lui demande !

4 La gestion du projet

4.1 Planning

4.2 Organisation

4.3 Difficultés et solutions

5 Conclusion