

Un robot nommé Toby

Petit récapitulatif de qui est Toby :

Toby est né de la volonté de révolutionner la domotique. Les assistants vocaux sont omniprésents de nos jours et capables de réaliser une multitude de tâches mais restent toujours immobiles. Nous avons alors voulu combler ce manque de déplacement en le dotant de la possibilité de se déplacer. C'est de là qu'est venue l'idée de Toby, l'assistant vocal sur chenille capable de se déplacer pour aider son propriétaire.

Cahier des charges théorique de Toby :

Fonctions de service	Criteres	Niveaux	Flexibilité
Repondre a certaines questions	Meteo/wikipedia/etc....	N/A	5
Se deplacer de maniere autonome	Venir a nous	N/A	3
Simuler une reaction	Yeux avec ecran OLED	10 visages	4
Autonomie	Batteries	30min en utilisation intensifs et 2h en utilisation classique	3
Vitesse		3-4 km/h	3
Deplacement sur plusieurs terrains	plat/terre		1

Cahier des charges final :

- Répondre à des questions ✓
- Se déplacer de manière autonome ✗
- Déplacer Toby à distance ✓
- Simuler une réaction ✓
- Autonomie / batterie ✗
- Vitesse ✓
- Déplacement sur plusieurs terrains ✓

Conclusion cahier des charges :

Le projet final ne respecte pas entièrement le cahier des charges initial. En effet, certaines fonctions n'ont pas pu être respectées. Certaines parties du robot ont été bien plus longues à réaliser que prévues et ainsi le retard encaissé pour finir toutes ces fonctions nous a contraint de ne pas pouvoir respecter le cahier des charges en entier. Pour le déplacement autonome, nous avons privilégié finir le déplacement à distance par l'utilisateur plutôt que d'avoir 2 fonctionnalités peut-être convaincantes. Enfin pour l'autonomie, notre robot devait être branché directement par secteur, la batterie nous a posé des problèmes notamment de par sa taille et son poids. Cependant, les fonctions les plus importantes telles que le robot puisse répondre à une question, pouvoir le déplacer ou lui permettre des expressions faciales sont présentes.

Planning théorique :

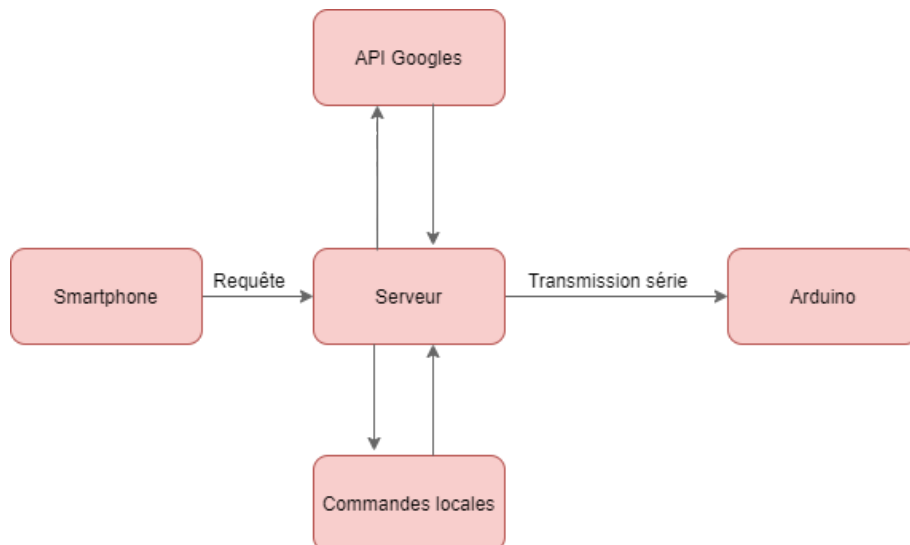
	Antoine	Thomas
Semaine 1	Connection entre arduino et téléphone par wifi et commandes faciles à utiliser avec Google Assistant. Connection au cloud avec l'arduino	Dimensionnement du robot
Semaine 2		Création des expressions faciales du robot sur écran OLED
Semaine 3	Réponses aux questions basiques telles que la météo, l'heure, etc...	Déplacement autonome du robot (sans arrivée précise)
Semaine 4		Déplacer le robot avec la voix
Semaine 5	Modélisation 3D du robot (roues)	Modélisation 3D du robot (corps)
Semaine 6		
Semaine 7	Réalisation premier prototype	Réalisation premier prototype
Semaine 8	Corrections des bugs	Correction des bugs

Planning réel et problèmes rencontrés :

Pour les deux premières semaines, le planning a été respecté. Nous avons d'abord commencé le développement de l'application grâce au framework Flutter pour communiquer les informations au serveur que nous avons également développé grâce au framework NestJS dès la première semaine. Nous avons aussi configuré le SDK (System Development Kit) fourni par Google sur le serveur pour pouvoir communiquer avec les API de Google. Enfin nous avons dimensionné le robot (moteur, largeur, hauteur, etc...) et codé une grande partie des expressions faciales. Cependant, dès la troisième semaine, les problèmes se sont suivis et il était presque impossible de suivre le planning. En effet, le premier problème était de transmettre et stocker les informations du serveur à la carte Arduino pour ensuite les lire. La prise en main du module et trouver les bugs lors de la connexion a été difficile. On a eu beaucoup de problèmes de transfert, de stockage de l'information, etc... La détection vocale sur l'application fut également un problème difficile à résoudre de par la complexité du plugin utilisé. Ainsi, toutes les fins de planning furent très peu respectées.

Les différentes fonctions de Toby :

Répondre aux demandes vocales de l'utilisateur:

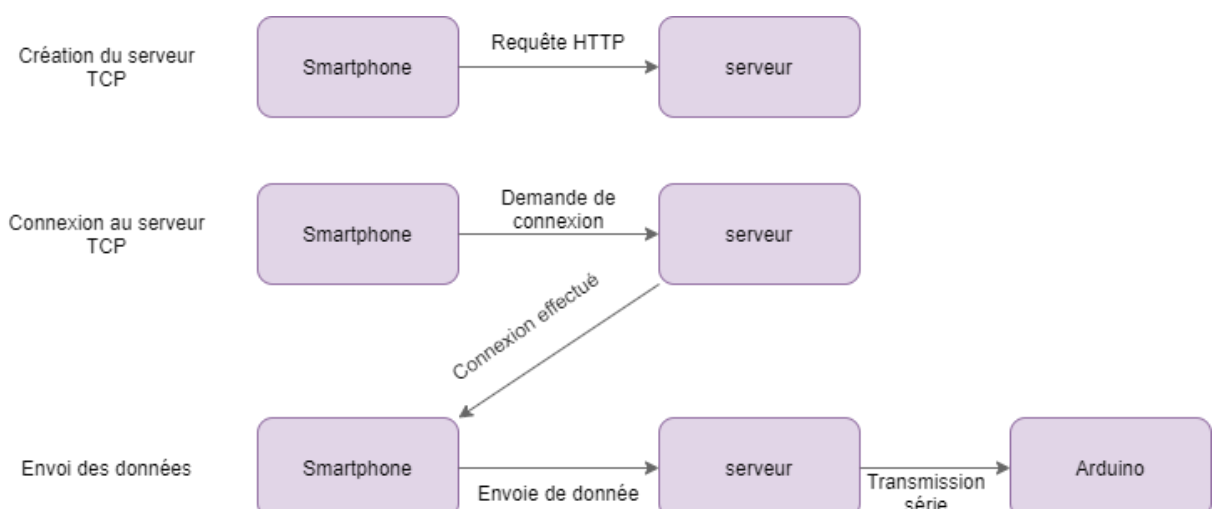


Pour que Toby puisse répondre à nos questions, la logique suit un algorithme précis. Tout d'abord, l'application sur le smartphone retranscrit la voix de l'utilisateur sous forme de texte. L'application envoie alors le texte via une requête HTTP à notre serveur. Nous avons alors 2 cas possibles :

- Si le texte correspond à une demande préconfiguré, alors on récupère la réponse en stocké dans un fichier sur le serveur en local et on le transforme en fichier mp3
- Sinon, on envoie une requête aux API de Google et on récupère la réponse mp3 fourni

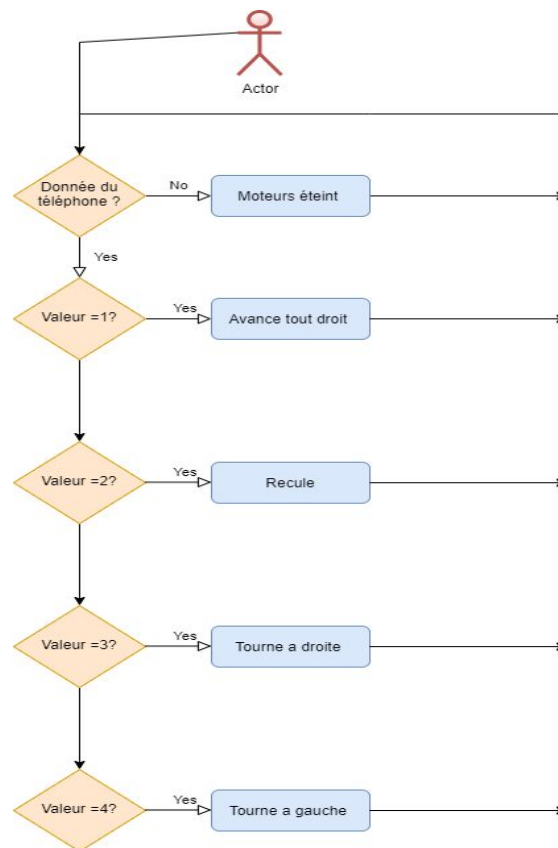
Une fois la réponse récupérer, on l'envoie via liaison série depuis le serveur à l'Arduino et on stocke les informations dans un fichier mp3 que l'on lit une fois toutes les informations recuperer.

Pouvoir être contrôlé depuis le téléphone :



Pour pouvoir contrôler Bobby à distance, il existe un onglet sur notre application. Lorsque l'utilisateur veut prendre le contrôle alors une requête HTTP est envoyé au serveur. Le serveur va alors créer un serveur TCP secondaire. Une fois le serveur TCP créer, le smartphone envoie une demande de connection que le serveur accepte et le transfert fluide des données peut alors commencer.

Diriger Bobby:



Pour contrôler le robot l'utilisateur verra apparaître sur son écran des flèches haut, bas, gauche, droite. Si l'utilisateur n'appui sur aucune flêhe alors Bobby ne fait rien mais dès que l'utilisateur appuie sur une des quatres flèches, le smartphone envoie alors une donnée en fonction de la flèche correspondante et en fonction de cela le code arduino saura qu'elle fonction lancé pour déplacer Bobby.

Quelles ont été les limites de notre projets :

Nous nous sommes vite rendu compte que la création d'un assistant vocal allez être bien plus complexe que ce que nous pensions. Cela a eu un fort impact sur le design et le temps apporté au visuel de Bobby auquel on a donné très peu de temps. Ensuite, le temps entre lorsque l'utilisateur parle au smartphone et le temps que Bobby reponde fut bien plus long que prévu. En effet, la vitesse d'écriture de la carte SD n'est pas assez rapide pour assurer un temps de transfert rapide et on dû bloquer la vitesse de la transmission série à 9600 bauds. Si la vitesse était trop élevé, des pertes de données apparaissait ce qui rendait la réponse incompréhensible voir qui rendait le fichier illisible. De plus, si la question était trop longue, la détection de la voix pouvait mal s'effectuer mal et plusieurs requêtes était alors envoyé et qui bloquait alors le serveur. Enfin, la programmation du serveur avec le SDK Google assistant car il y a avait un mélange de programmation fonctionnel et événementiel assez difficile à prendre en main et qui

Ce que nous à apporter ce projet:

Point de vue de Thomas :

Pour moi ce projet m'a apporté deux grande chose. La première, une bien meilleure connaissance de tout ce qui touche à la programmation de type événementielle , chose que je ne connaissais pas. Heureusement, Antoine était là et m'a permis de vite comprendre et pouvoir travailler dessus. La deuxième chose que m'a apporté ce projet est la capacité à travailler de manière autonome car même si j'étais avec Antoine chacun avait ces objectifs et je devais donc être autonome pour ne pas ralentir Antoine.

Point de vue d'Antoine :

Ce projet à réaliser en groupe a été très bénéfique pour moi. D'une part, j'ai toujours eu l'habitude de travailler seul sur des projet et ainsi de toujours faire tout, ce projet en groupe m'a alors permis de faire confiance à Thomas pour les tâches qu'il devait réaliser de son côté. On devait ensuite regrouper et joindre nos travaux pour que tout fonctionne et nous a alors appris à nous organiser, planifier les tâches, communiquer pour que tout se passe bien. Enfin, le développement du serveur avec le SDK qui nécessite de la programmation événementielle m'a permis de découvrir ce type de programmation qui est très utilisé, de pouvoir également approfondir mes connaissances dans la reconnaissances vocales, la structure de projet. Pour conclure, ce projet ne fut que benefique au point de vue sociale et de connaissances acquise.

Apprendre des ses difficultés :

Si nous devions refaire ce projet en connaissant les difficultés précise. On aurait une approche différentes du déroulement du projet. Tout d'abord, on passera beaucoup moins de temps sur le design de l'application, du robot, des expressions faciales et de même pour les performances sur serveur et de l'application.

Rendu final du projet :

Finalement, Toby remplit une majeure partie de ses fonctions. Il peut être déplacé à distance depuis l'application sur plusieurs types de terrain grâce aux chenilles. Il répond également à des questions comme "Quelle est la météo de Paris" mais également aux commandes personnalisées comme "Bonjour Toby". Enfin, il peut également s'exprimer avec ses yeux et sa bouche.

