



Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [Préparer votre participation](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à info@trophees-nsi.fr.

NOM DU PROJET : MIKU

> PRÉSENTATION GÉNÉRALE :

Idée : MIKU est une IA à commandes et réponses vocales nomade qui utilise ChatGPT (GPT-3).

L'Objectif du projet « MIKU » est de réaliser une IA conversationnelle en utilisant ChatGPT.

L'originalité de ce projet c'est d'avoir créé une interface homme-machine (IHM) complètement audio avec l'Intelligence Artificielle. On interroge l'IA en parlant et elle devra nous répondre en parlant également.

*On désire aussi que MIKU soit nomade et on développera ce projet sur un nano-ordinateur de type **Raspberry Pi**. La faible consommation de ce nano-ordinateur doit rendre possible cette version nomade du projet.*

On ajoutera trois LEDS, une verte, une jaune et une rouge pour compléter l'IHM dans la version nomade.

Origines et intérêts :

Depuis quelques mois on entend tous parler de ChatGPT et plus généralement des IA.

Réaliser un projet en utilisant la version gratuite de ChatGPT et en utilisant uniquement des logiciels libres permettra à chacun de pouvoir s'approprier ce projet.

Le coût en sera minimisé et la mise en œuvre devra être la plus simple possible. Le système, les applications, librairies, programmes, et matériels (Pi) utilisés dans ce projet sont tous libres de droits et très peu énergivores (nomade).

Seule la réponse de l'IA ChatGPT ne peut être contrôlée. Ce projet doit nous permettre d'apprendre à utiliser l'IA de manière intelligente, apprendre à poser les bonnes questions, être plus efficaces dans notre rapport à l'IA. C'est incontournable, il y en aura de plus en plus. En réalisant ce projet ludique à moins de 100€ et nomade nous pourrons tous appréhender le monde de l'IA.

> ORGANISATION DU TRAVAIL :

L'équipe est composée de 4 membres :

Prénom Nom	Genre	Partie	Tâches
Léa DELATTRE	F	1	OS, VOSK, Traitement texte 1
Ninon GODART	F	1	OS, VOSK, Traitement texte 1
Julien BUSEYNE	M	2	ChatGPT, Traitement texte 2
Louis JANKOWSKI	M	3	Espeak, Mbrola.

Répartition des tâches :

Partie 1 :

Léa et Ninon s'occupent dans un premier temps de réaliser le système d'exploitation, on utilise une carte Raspberry Pi4 (8Go) et le système : Raspberry Pi OS(64-bits) avec bureau.

<https://www.raspberrypi.com/software/operating-systems/>

Dans un deuxième temps elles ont réalisé l'analyse vocale VOSK, elle ont utilisé le programme : **test_microphone.py** (dossier Sources) comme programme de base qu'elles ont ensuite modifié pour répondre aux exigences de cette première partie.

Elles ont écrit la fonction : **Traitement_texte_1**

La fonction "**Traitement de texte 1**"

sert à enlever les caractères qui structurent la réponse de VOSK afin d'envoyer à ChatGPT une question textuellement bien structurée.

Exemple:

Réponse de VOSK:

```
{  
  "text" : "Raconte moi une histoire de lapin"  
}
```

Après traitement: Les caractères rouges ont été supprimés.

Raconte moi une histoire de lapin

Partie 2:

Julien s'occupe de:

- 1: l'utilisation de ChatGPT, écriture d'une fonction en python qui permette d'envoyer une question et de récupérer la réponse dans une variable de type texte.
- 2: L'installation de la librairie openai.
- 3: Création d'une clé API en créant un compte chez openai.

Lors des tests audios, les élèves ont remarqué des erreurs de prononciation.

Exemple: si dans la réponse il y a une valeur de volume de **3.5m3**, la synthèse vocale prononçait **3.5** «m» «3» au lieu de dire «**mètre cube**».

Ils ont donc identifié quelques incohérences

4: Julien a réalisé une fonction:

Traitement de texte 2 qui permet de remplacer dans les réponses de ChatGPT les incohérences audios.

Partie 3:

Louis s'occupe de:

- 1: Installer la synthèse vocale Espeak et MBrola
- 2: Faire les réglages de la synthèse vocale, voix, vitesse.
- 3: Montage vidéo.

Organisation temporelle:

On a utilisé les heures d'AP du samedi matin, et pendant l'année on a construit, pas à pas, pratiquement tous les pré-requis pour mener à bien ce projet.

Tp serveur LAP, Tp commande vocale,...

Les deux dernières semaines de cours ont été consacrées à la mise au point des 3 parties et à l'intégration générale.

Outils/logiciels utilisés pour la communication et le partage du code:

Les programmes ont été développés avec l'application Thonny.
Sur un clavier ordinateur de type Pi400 (doc).

Chaque élève possède une Pi400 avec sa carte Micro-SD.

Nous avons un serveur de fichiers en réseaux qui permet l'échange de programmes.

Les 15 derniers jours nous avons travaillé à distance puisque nous étions en vacances scolaires le vendredi 15/04.

Nous avons donc utilisé notre ENT pour la messagerie, la Web conférence de l'ENT pour un entretien audio.

Notre site maths-code.fr met également à disposition tous nos cours et un forum spécial projet «MIKU» avec un chat instantané.

> LES ÉTAPES DU PROJET :

Etape 1:

Les trois parties du projet ont été développées simultanément.
Elles sont indépendantes et peuvent être testées séparément.

Etape 2:

Intégration de chat GPT avec la partie 1.

Etape 3:

Intégration de la synthèse vocale dans le programme principal.

Etape 4:

Intégration de l'IHM et des trois LEDS.

Etape 5:

Modification du programme principal pour qu'il ne réponde qu'à l'appel de son nom «Miku»

Etape 6:

Rendre le programme exécutable au démarrage pour la version nomade.

Etape 7:

Intégration des composants dans la boîte «Miku»

Etape 8:

Tests, mesure de consommation, améliorations,...

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

*Nous avons obtenu une version finalisée qui répond parfaitement à notre cahier des charges.
La totalité du projet «Miku» est opérationnelle.*

La principale difficulté a été de gérer le fonctionnement de l'analyse vocale VOSK.

Elle mémorise constamment le son, en fait elle déclenche l'enregistrement dès qu'elle capte du son avec le micro.

Elle arrête l'enregistrement si le son s'arrête pendant au moins 1 seconde. Si vous laissez le micro actif, le son des conversations locales sera capté par votre micro et VOSK passera son temps à convertir du son qui ne lui est pas destiné. Cela a un impact sur le temps de réponse de VOSK lorsque l'on parle... si

dans la queue il reste du son indésirable qui a été enregistré précédemment, ce que vous venez de dire mettra un certain temps avant que VOSK vous renvoie sa conversion en mots, voir il disparaîtra!

Pour corriger ce dysfonctionnement, on a acheté un micro avec interrupteur, ce qui permet de couper le son lorsqu'on n'utilise pas «Miku», la réponse de VOSK est maintenant suffisamment rapide pour rendre ce projet utilisable.

2 petits inconvénients:

- ChatGPT utilisé dans sa version gratuite n'est pas prioritaire sur les accès à ce service, le temps de réponse de l'IA peut varier de façon significative.

- Lorsqu'on allume le micro, il faut attendre 1 à 2 seconde(s) pour qu'il soit bien identifié et intégré au système.

Si non vous risquez de perdre le début de votre question!

> OUVERTURE :

Idées d'améliorations:

MIKU n'est pas seulement une IA conversationnelle, elle a un peu d'IHM avec la commande des LEDS.

On pourra développer la partie domotique dans une utilisation à la maison ou à l'école!

Accéder à des services en ligne pour récupérer des informations (ex: la météo locale,...)

Des versions en langues étrangères : anglaise,.. VOSK possède une vingtaine de langues différentes.

Pour toucher un large publique on compte sur les réseaux qui vont promouvoir notre projet.

Pour le rendre accessible à tous, il faut héberger une image du système complet, la procédure pour réaliser ce projet ne prendrait pas plus d'une heure pour le rendre opérationnel.

Une version minimaliste qui tient dans la poche en implémentant ce projet sur une Raspberry Pi zéro 2W.

Analyse critique:

Le projet Miku n'est pas terminé, il sera en constante amélioration en intégrant les nouvelles incohérences audios détectées dans l'utilisation de Miku.

DOCUMENTATION

Programme Principal (doc):

