**REPUBLIQUE DU SENEGAL**

****

**Un peuple-un but-une foi**

**Ministère de l’Enseignement Supérieur, de la Recherche et de l’Innovation**

**Direction de l’Enseignement Supérieur Privé**

**Institut Supérieur d’Informatique**

**ISI**

**Mémoire de fin de cycle pour l’obtention de la licence professionnelle**

**SPECIALITE** : INFORMATIQUE

**OPTION** : Informatique Industrielle

**CONCEPTION D’UN SYSTEME INTEGRANT l’IA POUR LA COMMANDE DE L’ALIMENTATION DES CHARGES ELECTRO-MENAGER**

**Présentés et soutenus par : Sous la direction de :**

**Prince Gildas MBAMA KOMBILA**

**NGai Hermann**

**Mr Momar SOURANG**

**Année Académique : 2021 -2022**

# A la mémoire de :

Tous ceux qui ont contribué à mon éducation, à ma formation et à notre réussite et qui ne sont plus là malheureusement :

* A ma cher tante SONIA
* A mon grand Père
* A ma tante La Reine mère

# Dédicace

C’est avec une profonde gratitude et sincère mots, que je dédie ce modeste travail de fin d’étude à mes chers parents, qui ont sacrifiés leurs vies pour ma réussite et m’ont éclairé le chemin par leurs conseils judicieux.

J’espère qu’un jour je pourrai leurs rendre un peu de ce qu’ils ont fait pour moi, que Dieu leurs prête bonheur et longue vie.

Je dédie aussi ce travail à toute ma famille de près ou de loin, et mes amis.

Et je dédie ce travail à moi bien sûr, malgré les nuits blanches, je suis fière de ce que j’ai faits.

**Cordialement**

# Remerciements

Avant tout je tiens d’abord à remercier ma mère de m’avoir donné la force, la motivation, le courage, la santé et les moyens afin de pouvoir accomplir ce modeste travail.

Ensuite je tiens à remercier particulièrement mon encadreur, MR Momar SOURANG pour l’orientation, la confiance, la patience qui a constitué un apport considérable sans lequel ce travail n’aurait pas pu être mené.

Mes vifs remerciements vont également aux membres du jury pour l’intérêt qu’ils porteront à mes recherches en acceptant d’examiner mon travail et de l’enrichir par leurs propositions.

Mes sincères remerciements et respects à tous les professeurs qui m’ont enseigné et qui par leurs compétences m’ont soutenu dans la poursuite de mes études.

Enfin je remercie tous ceux qui, de près ou de loin, ont contribués à la réalisation de ce travail.

**Cordialement**

# Avant-propos

L’institut Supérieur d’Informatique (ISI) fut fondé en 1988 sous le nom de JET INFORMATIQUE par des étudiants de l’université Cheikh ANTA DIOP de Dakar en collaboration avec ceux de l’université Laval au Québec Canada. Ainsi en 1994 il devient officiellement Institut Supérieur d’Informatique proposant une formation continue de type académique dans les domaines tels que l’informatique, la gestion, la comptabilité et l’organisation des entreprises offrant ainsi de nombreux diplômes tel que : le BTS, la Licence et le Master professionnel ou éventuellement un Doctorat. Ces formations disposent les connaissances théoriques et pratiques dans le secteur de l’informatique et des télécommunications. L’établissement dispose de plusieurs salles de classes modernes et des salles spécialisées équipées de matériels de pointe en informatique et de télécommunication. Notons aussi que l’institut a la faveur d’être reconnu par le CAMES et l’Anaq’Sup.

Pour l’obtention de la licence en Informatique Industrielle, ISI exige aux étudiants de rédiger un mémoire de fin de cycle sur un sujet dans leur domaine d’étude. C’est dans cette optique que j’ai élaboré ce document qui a pour sujet : **CONCEPTION D’UN SYSTEME INTEGRANT L’IA POUR LA COMMANDE DE L’ALIMENTATION CHARGES ELECTRO-MENAGER**

Ce document constitue notre premier travail de d’investigation académique, c’est pourquoi nous sollicitons de la part du jury, beaucoup d’indulgence pour ce qui concerne son évaluation.

# SOMMAIRE

[A la mémoire de : II](#_Toc118468161)

[Dédicace III](#_Toc118468162)

[Remerciements IV](#_Toc118468163)

[Avant-propos V](#_Toc118468164)

[SOMMAIRE VI](#_Toc118468165)

[Glossaire VII](#_Toc118468166)

[LISTE DES FIGURES IX](#_Toc118468167)

[LISTE DES TABLEAUX X](#_Toc118468168)

[Résumé XI](#_Toc118468169)

[Abstract XII](#_Toc118468170)

[Introduction générale 1](#_Toc118468171)

[Première Partie : Cadre théorique et méthodologique 3](#_Toc118468172)

[I. CADRE THEORIQUE 4](#_Toc118468173)

[II. CADRE METHODOLOGIQUE 6](#_Toc118468174)

[Deuxième partie : le cadre conceptuel 7](#_Toc118468175)

[III. Généralités sur l’intelligence artificielle 8](#_Toc118468176)

[IV. Reconnaissance vocale 13](#_Toc118468177)

[V. Système intelligent 16](#_Toc118468178)

[Troisième partie : Mise en œuvre 21](#_Toc118468179)

[VI. Conception et réalisation 22](#_Toc118468180)

[Conclusion générale 38](#_Toc118468181)

[Webographie 39](#_Toc118468182)

[Annexes 40](#_Toc118468183)

[TABLE DES MATIÈRES 59](#_Toc118468184)

DL

Deep Learning (Apprentissage profond) 8

IA

Intelligence Artificielle 1

ISI

Institut Supérieure d’Informatique V

ML

Machine larning ( Apprentissage Machine) 8

PDF

Portable Document Format 13

PPT

ducument Microsoft PowerPoint 13

XL

Grand forrmat 13

# LISTE DES FIGURES

[Figure 1: l'histoire de l'intelligence artificielle 9](#_Toc124593078)

[Figure 2:Sous domaines de l'IA 13](#_Toc124593079)

[Figure 3: reconnaissance vocale 15](#_Toc124593080)

[Figure 4:Fonctionnement du natural language understanding 16](file:///C:\Users\Administrator\Documents\mémoire\mémoire\Projet%20de%20fin%20d'etudes%20(5).docx#_Toc124593081)

[Figure 5: Technique de la reconnaissance vocale 16](#_Toc124593082)

[Figure 6: Principe de fonctionnement du computer vison 18](#_Toc124593083)

[Figure 7: Comment fonctionne le système 20](file:///C:\Users\Administrator\Documents\mémoire\mémoire\Projet%20de%20fin%20d'etudes%20(5).docx#_Toc124593084)

[Figure 8: Problèmes rencontrés 24](#_Toc124593085)

[Figure 9: liaison entre le système et l'esp32 27](file:///C:\Users\Administrator\Documents\mémoire\mémoire\Projet%20de%20fin%20d'etudes%20(5).docx#_Toc124593086)

[Figure 10: L'esp32 28](#_Toc124593087)

[Figure 11: Node MCU-32S 29](#_Toc124593088)

[Figure 12 : Présentation du serveur Web 30](#_Toc124593089)

[Figure 13: Mode station ou STA 31](#_Toc124593090)

[Figure 14: Relais 32](#_Toc124593091)

[Figure 15: Connexion de tension secteur 32](#_Toc124593092)

[Figure 16: Lampe 34](#_Toc124593093)

[Figure 17: Un ordinateur portable 35](#_Toc124593094)

[Figure 18 : La connexion entre l’esp32 et la lampe 35](#_Toc124593095)

[Figure 19: Plateforme Arduino 36](#_Toc124593096)

[Figure 20: HyperTerminal de l’Arduino (Moniteur Série) 37](#_Toc124593097)

[Figure 21: Structure générale du programme (IDE Arduino 38](#_Toc124593098)

# LISTE DES TABLEAUX

[Tableau 1: Avantages et inconvénients des différents systèmes d’exploitation 20](#_Toc124593062)

[Tableau 2:Avantages et inconvénients du système Lanc 21](#_Toc124593063)

# Résumé

La conception d'un système intégrant l'IA pour la commande de l'alimentation des charges électro-ménagères est un projet visant à développer un assistant virtuel, appelé LANC, qui peut être utilisé pour contrôler l'alimentation des appareils électroniques de la maison. Ce système est similaire à des assistants virtuels tels qu'Alexa, Siri et OK Google, mais est spécifiquement conçu pour gérer l'alimentation des appareils électroniques.

LANC est un logiciel développé sous Windows et doté d'une interface graphique intuitive pour faciliter son utilisation. Il utilise des commandes prédéfinies pour contrôler l'alimentation des appareils connectés, comme allumer ou éteindre les appareils électroniques. Il est également capable de faire des recherches sur Internet en répondant à des questions spécifiques.

Contrairement aux autres assistants virtuels populaires, LANC ne repose pas sur un réseau de neurones, ce qui limite ses capacités en termes de reconnaissance vocale et de compréhension des commandes. Il a été développé à partir de l'inspiration de l'IA nommée JARVIS, créée pour le film Iron Man.

La conception d'un tel système présente de nombreux avantages, notamment la possibilité de gagner du temps et de l'énergie en automatisant la gestion de l'alimentation des appareils électroniques. Il peut également aider à réduire les coûts énergétiques en éteignant les appareils qui ne sont pas utilisés ou en les mettant en veille. Cependant, les utilisateurs doivent être conscients que les fonctionnalités de LANC sont limitées par rapport aux autres assistants virtuels populaires qui reposent sur des réseaux de neurones avancés.

# Abstract

The design of a system incorporating AI for controlling power to household loads is a project to develop a virtual assistant, called LANC, that can be used to control the power to electronic devices in the home. This system is similar to virtual assistants such as Alexa, Siri and OK Google, but is specifically designed to manage power to electronic devices.

LANC is a Windows-based software program with an intuitive graphical interface for ease of use. It uses predefined commands to control the power of connected devices, such as turning on or off electronic devices. It is also capable of searching the Internet by answering specific questions.

Unlike other popular virtual assistants, LANC does not rely on a neural network, which limits its capabilities in terms of speech recognition and command understanding. It was developed from the inspiration of the AI named JARVIS, created for the movie Iron Man.

There are many benefits to designing such a system, including the ability to save time and energy by automating power management for electronic devices. It can also help reduce energy costs by turning off devices that are not in use or putting them on standby. However, users should be aware that LANC's functionality is limited compared to other popular virtual assistants that rely on advanced neural networks.

# Introduction générale

La conception d'un système intégrant l'IA pour la commande des charges électroménagers est un sujet de recherche qui a gagné en importance ces dernières années, notamment avec l'apparition de technologies comme Alexa d'Amazon, qui ont permis de contrôler les appareils électroniques à la maison par la voix.

Cependant, il existe encore des limitations dans l'utilisation de ces systèmes, notamment en ce qui concerne la compréhension de la langue naturelle et l'intégration avec un grand nombre d'appareils différents. La question principale qui se pose est donc de savoir comment concevoir un système intégrant l'IA qui soit capable de comprendre les commandes vocales de manière efficace et de les exécuter sur les appareils électroménagers connectés.

L'objectif principal de cette recherche est de développer un système intégrant l'IA capable de comprendre les commandes vocales de manière efficace et de les exécuter. Les objectifs spécifiques sont de :

Comprendre les différentes techniques d'IA utilisées pour comprendre les commandes vocales.

Évaluer les limites des technologies existantes pour la reconnaissance de la parole et la compréhension de la langue naturelle.

Concevoir et développer un système intégrant l'IA capable de comprendre les commandes vocales de manière efficace et de les exécuter sur les appareils électroménagers connectés.

Le choix de ce sujet est motivé par l'augmentation de la demande pour les technologies intelligentes dans la maison, qui permettent de contrôler les appareils électroménagers par la voix, ou encore faire des recherches sur internet ect.. Ainsi que par les avancées récentes dans les technologies d'IA, qui ont permis de concevoir des systèmes de reconnaissance de la parole plus précis.

Il est possible de concevoir un système intégrant l'IA capable de comprendre les commandes vocales de manière efficace et de les exécuter sur les appareils électroménagers, en utilisant les dernières avancées en matière de reconnaissance de la parole et de compréhension de la langue naturelle.

La recherche sera menée en utilisant une approche méthodologique combinant la revue de la littérature existante, l'évaluation des technologies existantes et le développement d'un prototype de système intégrant l'IA. Il se basera sur une analyse comparative pour les différentes techniques d'IA utilisées pour comprendre les commandes vocales, pour ensuite choisir celles qui sont les plus efficaces et les plus appropriées pour notre système. Le développement du prototype se fera en utilisant des techniques de développement logiciel et d'IA appropriées, telles que le machines Learning, les systèmes de traitement de la langue naturelle et les systèmes de contrôle des appareils électroménagers. Enfin, nous évaluerons les performances du système en utilisant des tests de reconnaissance de la parole et des tests de compréhension de la langue naturelle.

Le plan de la recherche comprendra sera divisé en 3 grandes parties qui comporteront les étapes suivantes :

* Cadres théoriques et méthodologiques
* Cadre Conceptuel
* Mise en Ouvre

: En conclusion, nous allons mettre en place un système intégrant l'IA pour la commande des charges électroménagers, en prenant en compte les avancées technologiques récentes, les problématiques rencontrées dans ce domaine pour enfin proposer une solution efficiente et fonctionnelle pour les utilisateurs.

# Première Partie : Cadre théorique et méthodologique

Dans cette partie nous allons d’abord voir en premier lieu le cadre théorique et en deuxième lieu le cadre méthodologique.

# CADRE THEORIQUE

Dans ce chapitre, nous allons aborder trois sections essentielles à savoir la problématique, les objectifs de notre étude et en fin les hypothèses de recherche.

## Problématique

Rappelons que l’intelligence artificielle permet aux machines de simuler l’intelligence humaine. En fonction des besoins l’intelligence artificiel peux être en atout major, elle permet aux entreprises comme aux particuliers de gagner un temps considérable lors de l’exécution de leur tâche. Prenons par exemple les chaines de production automobile, sur ces chaines nombreuses sont les entreprises qui utilisent des bras mécaniques automatisé ici on parle de COBOT (Association de « robot » et de « collaboratif », le cobot est un robot amélioré, évoluant à des vitesses réduites et accomplissant des tâches simples en collaboration avec l’être humain).

Par ailleurs le sous domaine qui nous intéresse dans l’intelligence artificielle pour ce document est : Le traitement du langage naturel plus précisément la reconnaissance vocale.

La reconnaissance vocale (ou reconnaissance automatique de la parole) consiste pour une application à analyser la voix humaine afin de la transformer en requête informatique. Tout passe par la voix dont les fréquences sonores sont captées par un micro avant d'être traduites sous forme de texte exploitable par la machine (ou speech-to-text).

L’idée ici est de travailler avec ces deux technologies pour créer un système intégrant L’IA et la reconnaissance vocale pour répondre à des questions préprogrammées ou encore exécuter des taches et la question à se poser est de savoir si cela est possible ?

De ce fait nous des objectifs ont été fixés afin d’aboutir à la conception de ce système et le rendre fonctionnel.

## Objectifs de recherche

Après avoir posé la problématique, nous avons jugé nécessaire de disposer d'un objectif général puis déterminer les objectifs spécifiques.

### Objectif général

L’objectif de cette recherche est la conception d’un système intégrant IA et la reconnaissance vocale aux vues de réaliser des tâches préprogrammées et de répondre à des questions préprogrammées.

### Objectifs spécifiques

Ce système permet :

* Faire des recherches sur internet
* Répondre à des questions basiques
* Contrôler des lampes par commande vocale
* Ouverture d’application, document par la reconnaissance vocale

## Hypothèse de recherche

Dans cette partie nous allons évoquer les hypothèses suivantes ?

* Peut-on se passé de la reconnaisse vocale de nos jours ? Pour répondre à cette question prenons par exemple Siri, Alexa ou même Google assistant. Ces systèmes ont une même particularité : ce sont des assistant personnel. Siri par exemple sur l’IPhone il permet de manipuler son téléphone par voix et maintenant imaginer une personne avec des handicap au niveau des membres de son corps cette assistant est d’une grande aide. Et Alexa imaginer pour rentrer dans votre maison et vous dites juste allume la lumière et la magie opère.
* L’intelligence artificielle est-elle dangereuse ? Les résultats produits par l’IA dépendent de la façon dont elle est élaborée et des données qu’elle utilise. Celle-ci peut être dangereuse que si la personne qui l’a programmée veut qu’elle le soit comme toutes technologie elle présente des avantages et des inconvénients.

C’est dans cette logique que nous allons aborder le cadre méthodologique ce sujet.

# CADRE METHODOLOGIQUE

Après avoir présenté la problématique, les objectifs et les hypothèses de notre étude dans le premier chapitre, il s’agira au cours de celui-ci de définir notre méthode de travail et les moyens utilisés pour le mener à bien. D'abord nous délimiterons le sujet, ensuite nous montrerons les techniques d’investigation et enfin nous présenterons les difficultés rencontrées.

## Cadre d’études

Pour la mise en place de ce système, il a été question d’utiliser un ordinateur portable (avec comme système d’exploitation Windows) et des composants électroniques en second.

## Techniques d’investigation

La réalisation et la rédaction d’un bon document nécessite forcément de réunir des informations, nous les avons obtenues grâce à une bonne documentation relative au sujet en parcourant le vaste monde d’INTERNET.

A partir de ces nombreuses techniques de recherche nous allons pouvoir bien cerner notre sujet. Ainsi, ces différentes techniques de recherche ont fait ressortir toutes les difficultés liées à ce sujet.

## Difficultés rencontrées

Pour des difficultés il y en a eu notamment :

* Les informations sur ce sujet étaient majoritairement en anglais
* Les programmes qui fonctionnaient quand il voulait
* Les données qui étaient corrompus sans savoir comment
* La difficulté du programme de comprendre ce que tu dis juste à un mauvais micro ou mauvaise connexion.

# 

# Deuxième partie : le cadre conceptuel

# Généralités sur l’intelligence artificielle

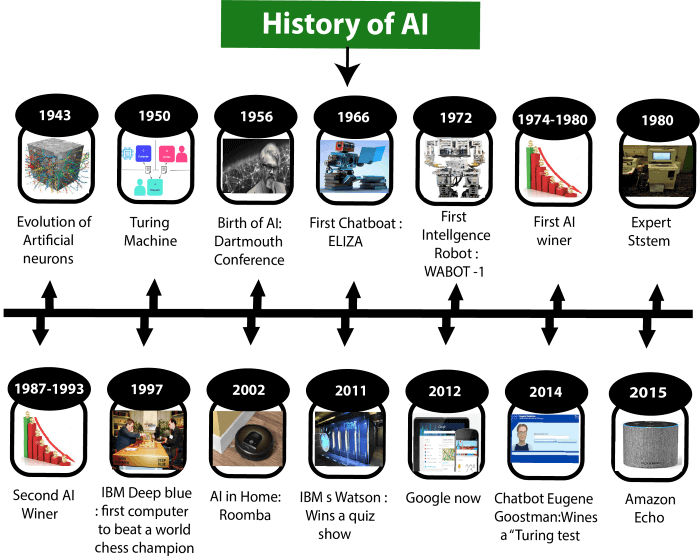
Dans cette partie nous parlerons de l’intelligence artificielle

## Intelligence artificielle

L'intelligence artificielle (IA) fait référence à la simulation de l'intelligence humaine dans des machines qui sont programmées pour penser comme des humains et imiter leurs actions. Le terme peut également être appliqué à toute machine qui présente des traits associés à un esprit humain tels que l'apprentissage et la résolution de problèmes.

La caractéristique idéale de l'intelligence artificielle est sa capacité à rationaliser et à prendre des mesures qui ont les meilleures chances d'atteindre un objectif précis. Un sous-ensemble de l'intelligence artificielle est [l'apprentissage automatique](https://www.investopedia.com/terms/m/machine-learning.asp) (ML), qui fait référence au concept selon lequel les programmes informatiques peuvent automatiquement apprendre et s'adapter à de nouvelles données sans être assistés par des humains. [Les techniques d'apprentissage en profondeur](https://www.investopedia.com/terms/m/machine-learning.asp)(DL) permettent cet apprentissage automatique grâce à l'absorption d'énormes quantités de données non structurées telles que du texte, des images ou des vidéos.

## Histoire de l’intelligence artificielle



**Figure 1: l'histoire de l'intelligence artificielle**

* **Année 1943 :** Les premiers travaux qui sont maintenant reconnus comme IA ont été réalisés par Warren McCulloch et Walter Pits en 1943. Ils ont proposé un modèle de **neurones artificiels**.
* **Année 1949 :** Donald Hebb a démontré une règle de mise à jour pour modifier la force de connexion entre les neurones. Sa règle s'appelle maintenant **l'apprentissage hebbien**.
* **Année 1950 :** L'Alan Turing qui était un mathématicien anglais et pionnier de l'apprentissage automatique en 1950. Alan Turing publie **"Computing Machinery and Intelligence"** dans lequel il propose un test. Le test peut vérifier la capacité de la machine à présenter un comportement intelligent équivalent à l'intelligence humaine, appelé **test de Turing**.
* **Année 1955 :** Un Allen Newell et Herbert A. Simon ont créé le "premier programme d'intelligence artificielle" qui a été nommé **"Logic Theorist ».** Ce programme avait prouvé 38 des 52 théorèmes mathématiques et trouvé de nouvelles preuves plus élégantes pour certains théorèmes.
* **Année 1956 :** Le mot "Intelligence Artificielle" est adopté pour la première fois par l'informaticien américain John McCarthy lors de la Conférence de Dartmouth. Pour la première fois, l'IA est devenue un domaine universitaire.
* **Année 1966 :** Les chercheurs ont mis l'accent sur le développement d'algorithmes capables de résoudre des problèmes mathématiques. Joseph Weizenbaum a créé le premier chatbot en 1966, nommé ELIZA.
* **Année 1972 :** Le premier robot humanoïde intelligent a été construit au Japon sous le nom de WABOT-1.
* **Année 1980 :** Après la durée hivernale de l'IA, l'IA revient avec "Expert System". Des systèmes experts ont été programmés pour imiter la capacité de prise de décision d'un expert humain.
* En 1980, la première conférence nationale de l'American Association of Artificial Intelligence **s'est tenue à l'Université de Stanford**.
* **Année 1997 :** En 1997, IBM Deep Blue bat le champion du monde d'échecs, Gary Kasparov, et devient le premier ordinateur à battre un champion du monde d'échecs.
* **Année 2002 :** pour la première fois, l'IA entre dans la maison sous la forme de Roomba, un aspirateur.
* **Année 2006 :** L'IA est arrivée dans le monde des affaires jusqu'en 2006. Des entreprises comme Facebook, Twitter et Netflix ont également commencé à utiliser l'IA.
* **Année 2011 :** Jeopardy Watson avait prouvé qu'il pouvait comprendre le langage naturel et résoudre rapidement des questions délicates.
* **Année 2012 :** Google a lancé une fonctionnalité d'application Android "Google now", qui était en mesure de fournir des informations à l'utilisateur sous forme de prédiction.
* **Année 2014 :** En 2014, Chatbot "Eugene Goostman" a remporté un concours dans le tristement célèbre "test de Turing".
* **Année 2018 :** Le "Project Debater" d'IBM a débattu de sujets complexes avec deux maîtres débatteurs et a également très bien performé.

[Les robots intelligents](https://builtin.com/artificial-intelligence/robotics-ai-companies) et les êtres artificiels sont apparus pour la [première fois](https://news.stanford.edu/2019/02/28/ancient-myths-reveal-early-fantasies-artificial-life/) dans les mythes grecs anciens. Et le développement du syllogisme par Aristote et son utilisation du raisonnement déductif ont été un moment clé dans la quête de l'humanité pour comprendre sa propre intelligence. Bien que les racines soient longues et profondes, l'histoire de l'IA telle que nous la concevons aujourd'hui s'étend sur moins d'un siècle. Ce qui suit est un bref aperçu de certains des événements les plus importants de l'IA.

### Domaine d’application de l’IA

L’IA est quasi visible partout et plus ou moins dans tous les domaines, si on devait citez les différents domaines il y a des centaines pour faire simple quelques exemples seraient cités ici :

L’un exemple que l’on utilise au quotidien sont les réseaux sociaux, l’IA traite et analyse les données pour vous donner les meilleures résultats possible grâce à des algorithmes poussés.

Dans le domaine de l’automobile les voitures sont désormais capables de conduire toutes seul, les robots sont capables de communiquer avec les humains, les chabots de Facebook et j’en passe, l’objectif ici n’est de pas rentrer en détail mais juste de citer quelques exemples.

### Différents types d’intelligence artificielle

L'intelligence artificielle peut être divisée en deux catégories différentes : **faible et forte.**

* [**L'intelligence artificielle faible**](https://www.investopedia.com/terms/w/weak-ai.asp)incarne un système conçu pour effectuer un travail particulier. Les systèmes d'IA faibles incluent les jeux vidéo tels que l'exemple d'échecs ci-dessus et les assistants personnels tels qu'Alexa d'Amazon et Siri d'Apple. Vous posez une question à l'assistant, et il y répond pour vous.
* [**Les systèmes d'intelligence artificielle forte**](https://www.investopedia.com/financial-technology-and-automated-investing-4689759) sont des systèmes qui effectuent des tâches considérées comme humaines. Ceux-ci ont tendance à être des systèmes plus complexes et compliqués. Ils sont programmés pour gérer des situations dans lesquelles ils peuvent être amenés à résoudre des problèmes sans qu'une personne n'intervienne. Ces types de systèmes peuvent être trouvés dans des applications telles que les voitures autonomes ou dans les salles d'opération des hôpitaux.

### Sous-domaine de l’intelligence artificielle

* **Apprentissage automatique (Machine learning)**

L'apprentissage automatique est une fonctionnalité de l'intelligence artificielle qui donne à l'ordinateur la capacité de collecter automatiquement des données et d'apprendre de l'expérience des problèmes ou des cas rencontrés plutôt que d'être spécialement programmé pour effectuer la tâche ou le travail donné.

* **Apprentissage profond (Deep learning)**

C'est le processus d'apprentissage en traitant et en analysant les données d'entrée par plusieurs méthodes jusqu'à ce que la machine découvre la sortie unique souhaitable. Il est également connu sous le nom d'auto-apprentissage des machines.

* **Réseaux et neurones (Neural network)**

Les réseaux de neurones sont le cerveau de l'intelligence artificielle. Ce sont les systèmes informatiques qui sont la réplique des connexions neuronales dans le cerveau humain. **Les neurones artificiels correspondants du cerveau sont connus sous le nom de perceptron.**

* **Calcul cognitif (cognitive computing)**

Le but de cette composante de l'intelligence artificielle est d'initier et d'accélérer l'interaction pour l'achèvement de tâches complexes et la résolution de problèmes entre les humains et les machines. Tout en travaillant sur divers types de tâches avec des humains, les machines apprennent et comprennent le comportement humain, les sentiments dans diverses conditions distinctives et recréent le processus de pensée des humains dans un modèle informatique.

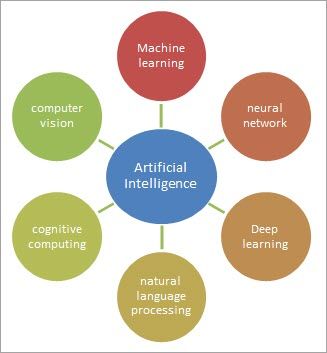
* **Traitement du langage naturel (NLP Naturel Language Processing)**

Grâce à cette fonctionnalité de l'intelligence artificielle, les ordinateurs peuvent interpréter, identifier, localiser et traiter le langage et la parole humains. Le concept derrière l'introduction de ce composant est de rendre l'interaction entre les machines et le langage humain transparente et les ordinateurs deviendront capables de fournir des réponses logiques à la parole ou à la requête humaine.

* **Vision par ordinateur (computer vison)**

La vision par ordinateur est une partie très vitale de l'intelligence artificielle car elle permet à l'ordinateur de reconnaître, d'analyser et d'interpréter automatiquement les données visuelles des images et des visuels du monde réel en les capturant et en les interceptant.

Il intègre les compétences d'apprentissage en profondeur et de reconnaissance de formes pour extraire le contenu des images de toutes les données fournies, y compris des images ou des fichiers vidéo dans un document PDF, un document Word, un document PPT, un fichier XL, des graphiques et des images, etc.



**Figure 2:Sous domaines de l'IA**

# Reconnaissance vocale

Dans cette partie, nous allons présenter la reconnaissance vocale, les différentes technologies qu’elle utilise, ses domaines d’application.

## Présentation de la reconnaissance vocale

La reconnaissance vocale (ou reconnaissance automatique de la parole) consiste pour une application à analyser la voix humaine afin de la transformer en requête informatique. Tout passe par la voix dont les fréquences sonores sont captées par un micro avant d'être traduites sous forme de texte exploitable par la machine (ou speech-to-text). Vient ensuite l'analyse de ces fichiers sonores par les technologies d'[intelligence artificielle](https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1493139-intelligence-artificielle/), et notamment de deep learning. Une seconde phase qui correspond au natural language understanding (NLU).

L'utilisation consécutive du speech-to-text et du natural language understanding permet d'obtenir une traduction optimale de la voix en données interprétables par la machine, et à partir desquelles cette dernière pourra ensuite apporter une réponse adéquate.

IBM a joué un rôle de premier plan dans la reconnaissance vocale depuis sa création, en lançant ["Shoebox"](https://www.ibm.com/ibm/history/ibm100/us/en/icons/speechreco/transform) en 1962. Cette machine avait la capacité de reconnaître 16 mots différents, faisant progresser les travaux initiaux des Bell Labs des années 1950. Cependant, IBM ne s'est pas arrêté là, mais a continué à innover au fil des ans, en lançant l'application [VoiceType Simply Speaking](https://www.ibm.com/ibm/history/ibm100/us/en/icons/speechreco/breakthroughs" \t "_blank) en 1996. Ce logiciel de reconnaissance vocale avait un vocabulaire de 42 000 mots, prenait en charge l'anglais et l'espagnol et comprenait un dictionnaire d'orthographe de 100 000 mots. Alors que la technologie vocale avait un vocabulaire limité à ses débuts, elle est aujourd'hui utilisée dans un grand nombre d'industries, telles que l'automobile, la technologie et la santé. Son adoption n'a fait que s'accélérer ces dernières années en raison des progrès de l'apprentissage en profondeur et du big data

## Domaine d’application

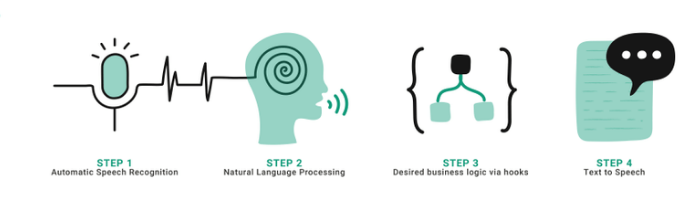
La technologie vocale a été déployée dans les assistants personnels numériques, les haut-parleurs intelligents, les maisons intelligentes et une large gamme d'autres produits. La technologie nous permet d'effectuer une variété de tâches activées par la voix.

Siri d'Apple et Alexa de Google utilisent la reconnaissance vocale basée sur l'IA pour fournir une prise en charge vocale ou textuelle, tandis que les applications voix-texte telles que Google Dictate transcrit vos mots dictés en texte. Taper avec votre voix vous permet de créer des e-mails et des documents en appuyant sur l'option microphone du clavier de votre appareil.

## Fonctionnement la reconnaissance vocale

La reconnaissance de la parole se décline en trois étapes :

* L'analyse acoustique qui permet de découper le message vocal en vecteurs acoustiques ingérables par la machine,
* Le [machine learning](https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501881-machine-learning/) qui associe ensuite les fréquences sonores à des mots.
* L'analyse de la parole qui combine trois modèles (un modèle de langage, un modèle de prononciation, un modèle accoustico-phonétique) en vue d'identifier les suites de mots les plus probablement prononcés par le locuteur.



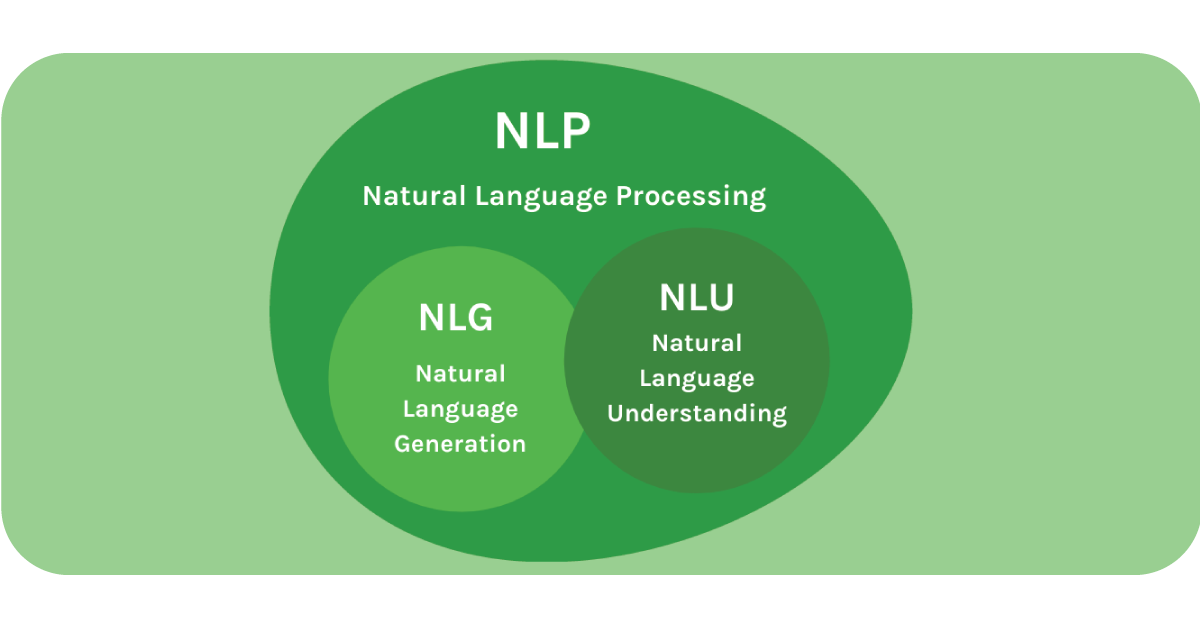
**Figure 3: reconnaissance vocale**

## Fonctionnement du natural language understanding ?

Le natural language understanding (NLU), ou [compréhension du langage naturel](https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501889-natural-language-understanding-nlu/), est un sous-domaine du traitement automatique du langage naturel (ou natural language processing ou NLP) qui s'adosse à des modèles de [deep learning](https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501333-deep-learning-definition-et-principes-de-l-apprentissage-profond/" \o "Apprentissage profond) pour permettre à la machine de saisir le sens d'un texte.

Le Natural Language Understanding (NLU) est une technique d'intelligence artificielle qui permet à une machine de comprendre le sens des phrases écrites en langage naturel. Il utilise des techniques de traitement automatique du langage naturel pour analyser le texte et en extraire des informations utiles, comme les concepts, les sentiments et les intensions.

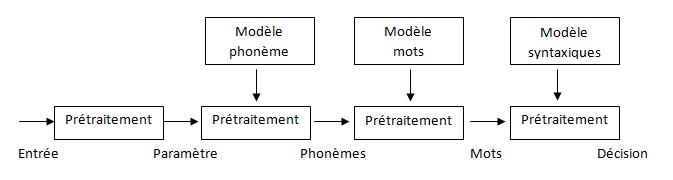
Cela peut inclure des tâches telles que l'analyse de la sémantique, la reconnaissance de l'entité, la reconnaissance de la relation et la détection de l'émotion. Les données de sortie de NLU sont généralement utilisées comme entrées pour d'autres systèmes tels que les assistants virtuels, les chatbots et les systèmes de réponse automatique.



**Figure 4:Fonctionnement du natural language understanding**

## Trois principales techniques de reconnaissance vocale ?

La reconnaissance vocale combine principalement trois modèles (un modèle de langage, un modèle de prononciation, un modèle acoustico-phonétique). Leur combinaison permet de calculer la probabilité la plus élevée d'apparition d'une suite de mots au sein d'un signal sonore. Leur entraînement nécessite un gros [data set](https://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501339-dataset/) ou base de données d'exemples vocaux étiquetés.



**Figure 5: Technique de la reconnaissance vocale**

# Le Computer vison

## Définition

La vision informatique (Computer Vision en anglais) est un domaine de l'intelligence artificielle qui se consacre à la compréhension de l'image par une machine. Il utilise des algorithmes pour analyser, comprendre et agir sur des images numériques et des vidéos. Il vise à simuler la capacité humaine de vision pour donner aux machines la capacité de "voir" et de comprendre le monde qui les entoure.

## Principe de fonctionnement du computer vison

Le principe de fonctionnement de la vision informatique repose sur l'utilisation d'algorithmes pour analyser et comprendre les images numériques. Les algorithmes peuvent être basés sur des modèles mathématiques ou sur des réseaux de neurones.

Les algorithmes de vision informatique peuvent être utilisés pour effectuer différentes tâches, telles que la détection de formes, la reconnaissance d'objets, la segmentation d'images et la reconstruction 3D.

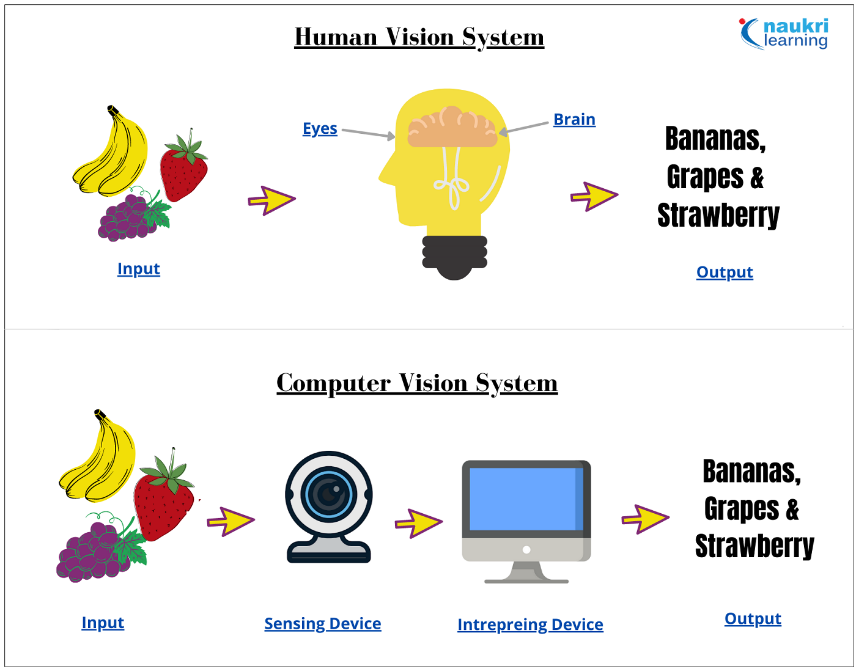
Pour la détection de formes, les algorithmes peuvent utiliser des techniques de traitement d'images pour détecter des caractéristiques spécifiques dans une image, telles que les bords, les contours et les coins.

La reconnaissance d'objets utilise des algorithmes de reconnaissance de formes pour identifier les objets dans une image en comparant les caractéristiques de l'image à une base de données d'images étiquetées.

La segmentation d'images consiste à séparer une image en différentes régions ou segments, chacun correspondant à un objet ou une région spécifique.

La reconstruction 3D utilise des algorithmes pour créer une représentation en 3D d'un objet à partir de plusieurs images prises à des angles différents.

En somme, la vision informatique utilise des techniques d'analyse d'image pour extraire des informations utiles à partir d'images numériques, et permet aux machines de "voir" et de comprendre le monde qui les entoure, en simulant la capacité humaine de vision.



**Figure 6: Principe de fonctionnement du computer vison**

# Système intelligent (LANC)

Dans cette partie, nous allons présenter la domotique, ainsi que ses objectifs, les différentes technologies qu’elle utilise, ses domaines d’application.

## Fonctionnement Le système

LANC ne signifie rien particulièrement, c’est juste un nom prit au Hazard.

Ce système peut être aussi appeler assistant personnel « intelligent » ou presque si l’on ajoute le terme « intelligent » cela signifierait qu’il est capable de réfléchir par lui-même pour donner les meilleurs résultats possibles or ici les réponses et les questions sont préprogrammées. Il est similaire à google assistant, Sir, Alexa leur rôles : vous assister dans votre quotidien. Mais alors pourquoi ajouté « Intelligent » ? Tout cela, car il arrive à reconnaitre le langage Humain et exécute des tâches.  Notre système est avant tout un [agent conversationnel](https://fr.wikipedia.org/wiki/Agent_conversationnel), dont la fonction de base est donc d'associer la [reconnaissance vocale](https://fr.wikipedia.org/wiki/Reconnaissance_automatique_de_la_parole), le [traitement de la langue naturelle](https://fr.wikipedia.org/wiki/Traitement_automatique_du_langage_naturel) et la synthèse vocale afin de dialoguer vocalement avec l'utilisateur. Pour notre part il fonctionne seulement sur le système d’exploitation Windows. Pourquoi ? Nous y reviendrons.

Dans un premier temps l’utilisateur pose une question par exemple « quelle heure fait-il ? »

La demande est envoyée par le biais du micro au programme ici on parle de « Speech Recognition ou reconnaissance vocale » le programme va convertir cette voix en texte « Speech to Text » le texte va être analysé et traité pour être compréhensible par la machine et au final la machine va convertir le texte en voix « Text to Speech » pour ressortir une réponse en se basant sur ceux que le programmeur aura écrit comme question et réponse.

Le programme reconnait toutes les voix car il n’a pas été entrainé pour une seule voix.

Si le programme ne reconnait aucun mot ou si le mot prononcé n’est pas dans sa base de données il ne dira rien pour notre part ou si voulu il dira « veuillez répéter » et à la longue ça en devient problématique c’est pourquoi sur des assistant comme, Siri, Alexa, Google assistant il y à des boutons sur lesquels il faut appuyer ou des mots spécifiques pour interagir avec l’assistant Ok Google, Dis SIRI, ALEXA, le nôtre il est toujours en mode écoute pour l’instant.

Notre système peut fonctionner hors ligne comme en ligne, lors du démarrage s’il détecte qu’il n’y a pas de connexion internet il basculera automatiquement en mode hors ligne en ouvrant le selon programme qui ne nécessite pas une connexion internet et vis-versa.

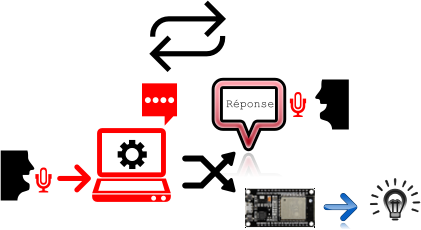
Noté que ce système ne bénéficie pas de base de données donc est très limiter en fonctionnalités pour le moment, et n’intègre pas de mot spécifique pour démarré il est toujours en mode écoute.

Réponse

Traitement/analyse

Question

Reconnaissance



**Figure 7: Comment fonctionne le système**

Pourquoi Windows ?

Pour répondre à cette question nous ferons un tableau pour noter les avantages et inconvénients que nous avons eu vis-à-vis des autres systèmes. L’idée de base était de développé ce système sur Raspberry mais cela ne s’ait pas très bien passé.

Tableau 1: Avantages et inconvénients des différents systèmes d’exploitation

et leur plateformes

|  |  |  |
| --- | --- | --- |
| Système d’exploitation | PC (Windows) | RASPBERRY (Raspbian) |
| Portabilité | Oui (un peu gros mais cela dépend de votre machine) | Oui (plus petit et plus facile à transporter) |
| Micro | oui | non |
| Baffle | oui | non |
| Batterie | oui | non |
| Clavier + souris + écran | oui | non |
| Prix | Le plus petit prix 35000 et varient selon les caractéristiques | 35000 en fonction du modèle noté qu’une carte mémoire compatible est nécessaire pour installer le système d’exploitation |
| Prise en main et installation des modules nécessaires | Facile | Assez complexe car tout les modules ne passeront pas directement |
| Modelé de voix | Mieux entrainés se rapproche plus de l’humain avec les paramètres par défauts | Les voix ressemblent beaucoup à ceux d’un robot assez désagréable à écouter avec le paramètre par défaut |
| Pins | Pas de Pins | Pins GPIO |

Alors pourquoi pas MacOs, Linus, ou même Android ? Tout simplement parce que je n’avais aucun de ces systèmes avec moi, comme j’ai préféré faire avec ceux que j’ai ?

## Avantages et inconvénients de LANC

Nous le rappelons cette version est une version dite MVP (produit minimal viable), donc l’objectif ici est d’obtenir des retours et ensuite amélioré le produit en fonction des retours ci-dessous une liste des avantages et inconvénients noté :

Tableau 2:Avantages et inconvénients du système Lanc

|  |  |
| --- | --- |
| Avantages | Inconvénients |
| Très réactif comparé aux autres systèmes faits maison | Nécessite un réseau wifi pour faire fonctionne les lampes |
| Fonctionne sans ou avec une connexion internet | Parle seul par moment |
| Il est doté d’une interface graphique | Il ne dispose pas de base de données |
|  | N’a pas de réseau de neurone (cerveau) |
| Reconnait toutes les voix | Fonctionne parfaitement dans un environnement fermé |
| Possibilité de contrôlé les lampe avec ses mains | Nécessite d’avoir son ordinateur toujours allumé |
| Possibilité de mettre son nom pour simuler une interaction | Peu configurable |
|  | Ne dispose pas de mot clé pour etre |

## Idée du projet

L'idée de projet proposée a pour thème la conception d'un système intégrant l'IA pour la commande des appareils électroménagers. Le système se veut similaire à Alexa,Siri, google home en permettant aux utilisateurs de contrôler leurs appareils connectés à leur réseau domestique à l'aide de commandes vocales, de poser des questions et de faire des recherches sur internet simplement en utilisant leur voix.

Pour réaliser ce projet, il est nécessaire de développer un modèle d'IA capable de comprendre les commandes de l'utilisateur en langage naturel, et de les traduire en actions pour les appareils connectés. Ce modèle doit être entraîné avec des exemples de commandes vocales pour qu'il puisse les reconnaître avec précision. Il peut utiliser des techniques de reconnaissance vocale et de traitement du langage naturel pour comprendre les commandes de l'utilisateur.

Une fois le modèle d'IA développé, il sera intégré dans un logiciel sous système d’exploitation Windows qui sera utilisée par les utilisateurs pour contrôler leurs appareils connectés. Ce logiciel doit être dotée d'une interface utilisateur intuitive pour que les utilisateurs puissent facilement naviguer et contrôler leurs appareils connectés.

En plus de cela, le logiciel pourrait inclure un système de reconnaissance vocale pour poser des questions et faire des recherches sur internet, en utilisant des techniques similaires à celles utilisées pour comprendre les commandes de l'utilisateur pour les appareils connectés.

Enfin, le logiciel doit inclure des fonctions de d’enregistrement de nom, pour donner à l’utilisateur cette impression de communiquer avec une autre personne0 et la possibilité de contrôlé les lampes en utilisant juste ça main.

En somme, cette idée de projet vise à créer un logiciel qui permet aux utilisateurs de contrôler leurs appareils électroménagers connectés à leur réseau domestique à l'aide de commandes vocales, de poser des questions et de faire des recherches sur internet simplement en utilisant leur voix, tout en ayant une interface utilisateur intuitive.

## Domaines d’applications et Fonctionnalités

### Fonctionnalités

Imaginez-vous rentrez tard du travail et vous êtes fatigué. En entrant dans la maison il vous suffit juste de dire « Allume la lumière » et comme par magies votre lumière s’allume d’autres fonctionnalités seront disponible notamment :

* Contrôle des lumières
* Recherche YouTube
* Recherche Google
* Recherche Wikipédia
* Donner l’heure
* Donner la date
* Donner la température
* Contrôle d’appareils électroménagers
* Jouer de la musique
* Et bien d’autre
* Possibilité de contrôler les lampes avec les mains

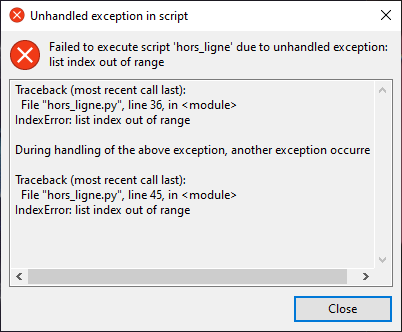
### Domaines d’applications :

* La santé
* Les Banques
* Militaire
* L’aérospatial
* Les jeux vidéo
* Le Cinéma
* Robotique
* La finance

Les assistants virtuels tels que Siri, Alexa, et Ok Google sont importants dans les domaines mentionnés car ils permettent de rendre les services plus accessibles et plus pratiques pour les utilisateurs. Dans le domaine de la santé, par exemple, les assistants virtuels peuvent aider les patients à trouver des informations sur des conditions médicales, des médicaments, ou des professionnels de santé, et peuvent même être utilisés pour suivre les données de santé. Dans les banques, les assistants virtuels peuvent aider les clients à gérer leurs comptes, effectuer des transactions, et obtenir des informations financières. Dans le domaine militaire, les assistants virtuels peuvent aider les soldats à accéder à des informations tactiques et à communiquer avec leur équipe. Dans l'aérospatial, les assistants virtuels peuvent aider les pilotes à naviguer et à effectuer des tâches complexes. Dans les jeux vidéo, les assistants virtuels peuvent fournir des informations sur les personnages, les armes, et les niveaux. Dans le cinéma, les assistants virtuels peuvent aider les cinéastes à filmer des scènes complexes. Dans la robotique, les assistants virtuels peuvent aider les utilisateurs à commander des robots et à les programmer. Dans la finance, les assistants virtuels peuvent aider les investisseurs à suivre leurs portefeuilles et à effectuer des transactions.

## Problèmes rencontrés

Dans un premier temps lorsqu’on exécutait le programme dans une autre machine le problème principal était le suivant :



**Figure 8: Problèmes rencontrés**

Ce problème survient lorsque la voix choisit est en dehors de la liste qui se trouve dans Windows. Pour résoudre ce problème, il a fallu revoir le code et ajouter certaines exceptions.

Autre problème rencontré est celui du micro, notre micro étant de mauvaise qualité il se peut que le système parle tout seul ou qu’il n’écoute pas très bien ceux que vous dites.

# Troisième partie : Mise en œuvre

# Conception et réalisation

Dans cette section, nous verrons les éléments nécessaires pour mettre en place notre projet.

## Choix des Matériaux

Dans ce chapitre, nous allons nous intéresser à l’aspect Hardware, aux composants électroniques utilisés :

* Cartes ESP32
* Relais
* Ampoule
* Câble USB, jumpers.
* Un ordinateur

Nous avons choisi l'ESP32 comme matériau principal pour contrôler nos lampes via WiFi. Les câbles et les relais sont nécessaires pour connecter l'ESP32 aux lampes, tandis que l'ordinateur est utilisé pour envoyer des commandes et programmer l'ESP32. Cette solution vous permet de contrôler vos lampes à distance, offrant ainsi une flexibilité et une commodité accrues.

L'ESP32 est un microcontrôleur basé sur l’architecture Xtensa Dual-Core 32-bit LX6. Il possède une mémoire flash de 4 Mo et une mémoire RAM de 520 Ko. Il intègre également une connectivité WiFi et Bluetooth, ce qui permet de contrôler les lampes via un réseau sans fil. Il est également doté d'un grand nombre d'entrées/sorties (GPIO), de ports série, de ports I2C et de ports SPI, ce qui facilite la connexion à d'autres périphériques.

Un arduino UNO est un autre type de microcontrôleur populaire basé sur un microcontrôleur AVR. Il possède une mémoire flash de 32 Ko et une mémoire RAM de 2 Ko. Il n'inclut pas de connectivité WiFi ou Bluetooth, mais peut être connecté à des modules de communication externes pour ces fonctionnalités. Il possède également une entrée/sortie numérique moins nombreuse et une fréquence de fonctionnement moins élevée par rapport à l'ESP32.

En résumé, l'ESP32 est un microcontrôleur plus performant et plus polyvalent que l'Arduino UNO, grâce à sa connectivité WiFi et Bluetooth intégrée, sa plus grande mémoire et ses entrées/sorties plus nombreuses. Il peut être utilisé pour des applications nécessitant une connectivité sans fil et un traitement de données plus importantes, tandis que l'Arduino UNO est plus adapté à des applications simples qui n'ont pas besoin de connectivité sans fil.

### Carte ESP32

Le ESP32 va servir de lien entre le système et les différents appareils.

«

Appareils connectés

Esp32

Système de contrôle

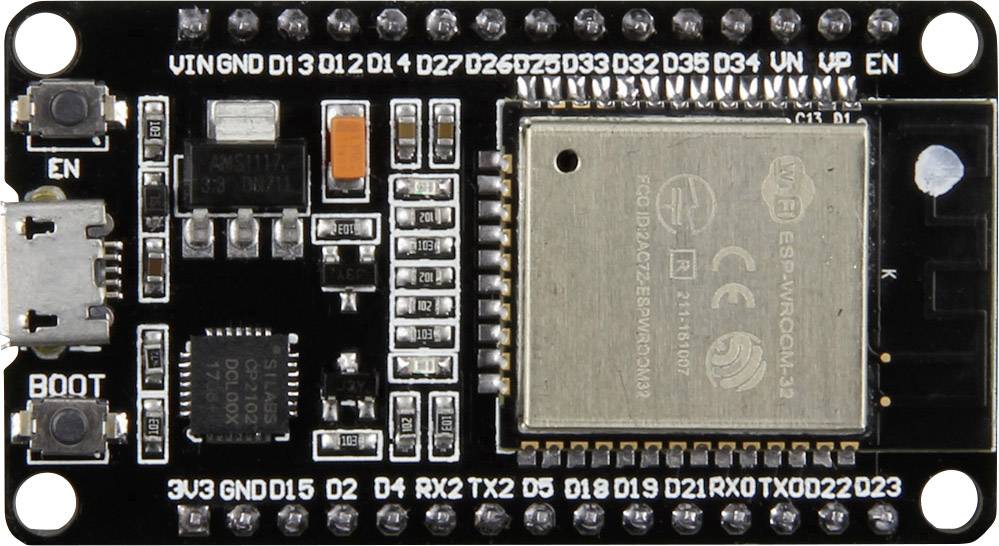
**Figure 9: liaison entre le système et l'esp32**

NodeMCU ESP32 est un microcontrôleur avec des modules Wifi et Bluetooth intégrés. Très simple d’utilisation il est léger et possède une capacité de mémoire et de calcul supérieure aux Arduino. Ce qui en fait une carte idéale pour l’apprentissage de la programmation, le développement d’objets connectés ou de serveur.

Le microcontrôleur NodeMCU ESP32 utilise le microprocesseur ESP-WROOM-32 (Tensilica Xtensa LX6). Ce processeur fonctionne à une fréquence d’horloge de 240 MHz. Il possède une mémoire RAM de 520 kB, EEPROM de 448 kB et aussi une mémoire Flash de 4000 kB (pour la programmation et l’enregistrement de données).

Le microcontrôleur possède une puce Wifi permettant de se connecter au réseau local, de créer un serveur ou de créer son propre réseau afin que d’autre appareils s’y connectent. Le microcontrôleur possède une puce Bluetooth qui lui permet d’interagir avec d’autres appareils.

* Prix faible ~5€ pour le module ~8€ pour une petite carte de developpement
* Intègre du Wifi 802.11 b/g/**n**/e/i (WPA)/WPA2/WPA2-Enterprise/Wi-Fi Protected Setup (WPS)
* Intègre Bluetooth **4.2** - BLE Bluetooth low Energy
* Compatible avec l'environnement de développement **ARDUINO**
* **Compatible avec le langage de programmation python**
* Intègre un microcontrôleur 32 bits performants et de nombreux périphériques (ADC **12bit,** DAC, 3xUART, PWM, I2C, SPI, etc …)
* S'alimente directement en USB

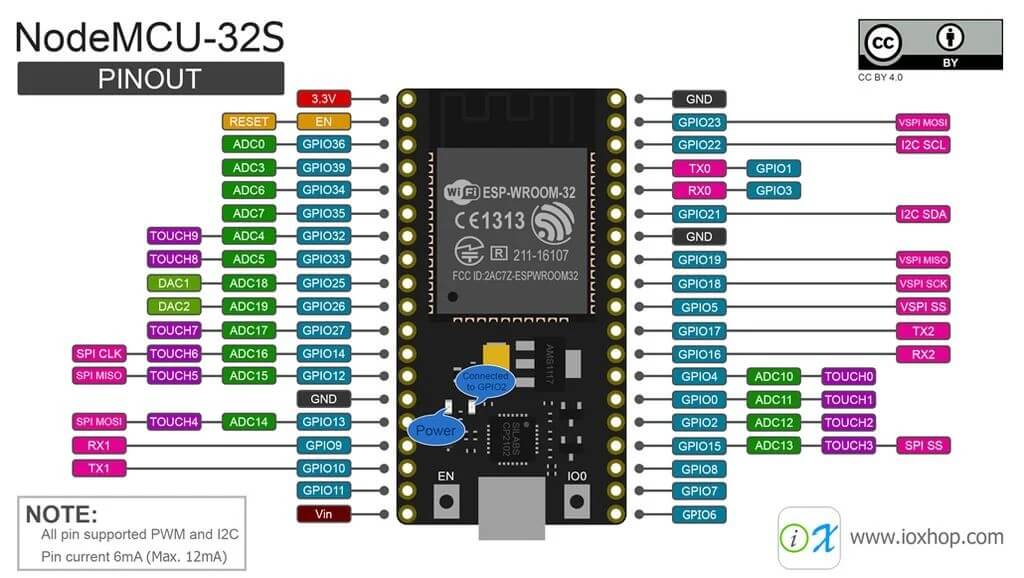


**Figure 10: L'esp32**

Le microcontrôleur NodeMCU ESP32 fonctionne sur une plage de tension de 7-12V grâce à son régulateur de tension embarqué.Le microprocesseur, quant à lui, fonctionne avec une tension de 3.3V. Le microcontrôleur consomme, en fonctionnement normal, jusqu’à 45mA (s’il n’alimente rien) et peut accepter sur chacune des broches IO un courant maximum de 40mA.

**PINOUT**

* Analog I/O : 15 (2, 4, 12, 13, 14, 15, 25, 26, 27, 32, 33, 34, 35, 36, 39)
* Digital I/O : 4 (all except 6 to 11)
* Broches PWM : 16 (2, 4, 5, 6, 12, 13, 14, 15, 18, 19, 21, 25, 26, 27, 32, 33)
* Communication Serial: 9 (1, 2, 3, 4, 5, 12, 13, 14, 15)
* Communication I2C : 1 ((’21’, ’22’))
* Communication SPI : 1 ((‘5′, ’18’, ’19’, ’23’))
* Communication I2S: 1 ((’26’, ’25’, ’33’))
* Touch : 9 (2, 4, 12, 13, 14, 15, 27, 32, 33)
* Interrupt : 26 (0, 1, 2, 4, 5, 7, 12, 13, 14, 15, 16, 17, 18, 19, 21, 22, 23, 25, 26, 27, 32, 33, 34, 35, 36, 39)

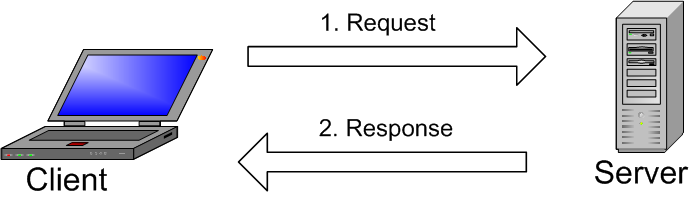


**Figure 11: Node MCU-32S**

#### **Présentation du serveur Web**

Lorsque nous parlons de serveur Web dans le contexte de l'Internet des objets, le serveur Web n'est rien d'autre que l'emplacement où les fichiers de notre site Web sont stockés et le serveur Web traite ces fichiers pour les clients en fonction de la demande d'un client. Les fichiers de site Web sont des pages Web de n'importe quel site. Comme vous utilisez ce site Web pour lire un article sur la création d'un serveur Web EPS32, cette page Web est stockée sur un serveur de notre société d'hébergement. Ainsi lorsque vous cliquez sur le lien pour ouvrir cet article, vous demandez un client à un serveur de notre hébergeur.

Après avoir reçu cette demande de page particulière, le serveur de notre société d'hébergement traite cette demande pour afficher cette page Web. De même, nous pouvons utiliser ESP32 comme hôte pour stocker des pages Web, et lorsque quelqu'un demande une page Web via un réseau local, esp32 servira cette page Web. L'image ci-dessous fournit une vue de la communication client-serveur.

[](https://microcontrollerslab.com/wp-content/uploads/2019/03/client-and-server.png)

**Figure 12 : Présentation du serveur Web**

#### **Service Web et communication client**

Il s'agit d'une idée de base de la communication basée sur le serveur et le client sur Internet. Maintenant, la question est de savoir comment établir la communication entre le client et le serveur Web ? HTTP, également connu sous le nom de contrôle de transfert hypertexte, est un protocole utilisé pour transférer des données entre le client Web et le serveur Web. Le serveur Web de ce didacticiel sera ESP32 et le client Web sera n'importe quel navigateur Web ou application Android. Chaque fois qu'un client Web a besoin d'accéder à des fichiers Web, il lance une requête HTTP GET au serveur Web.

#### **Comment un serveur Web répond-il à une requête client ?**

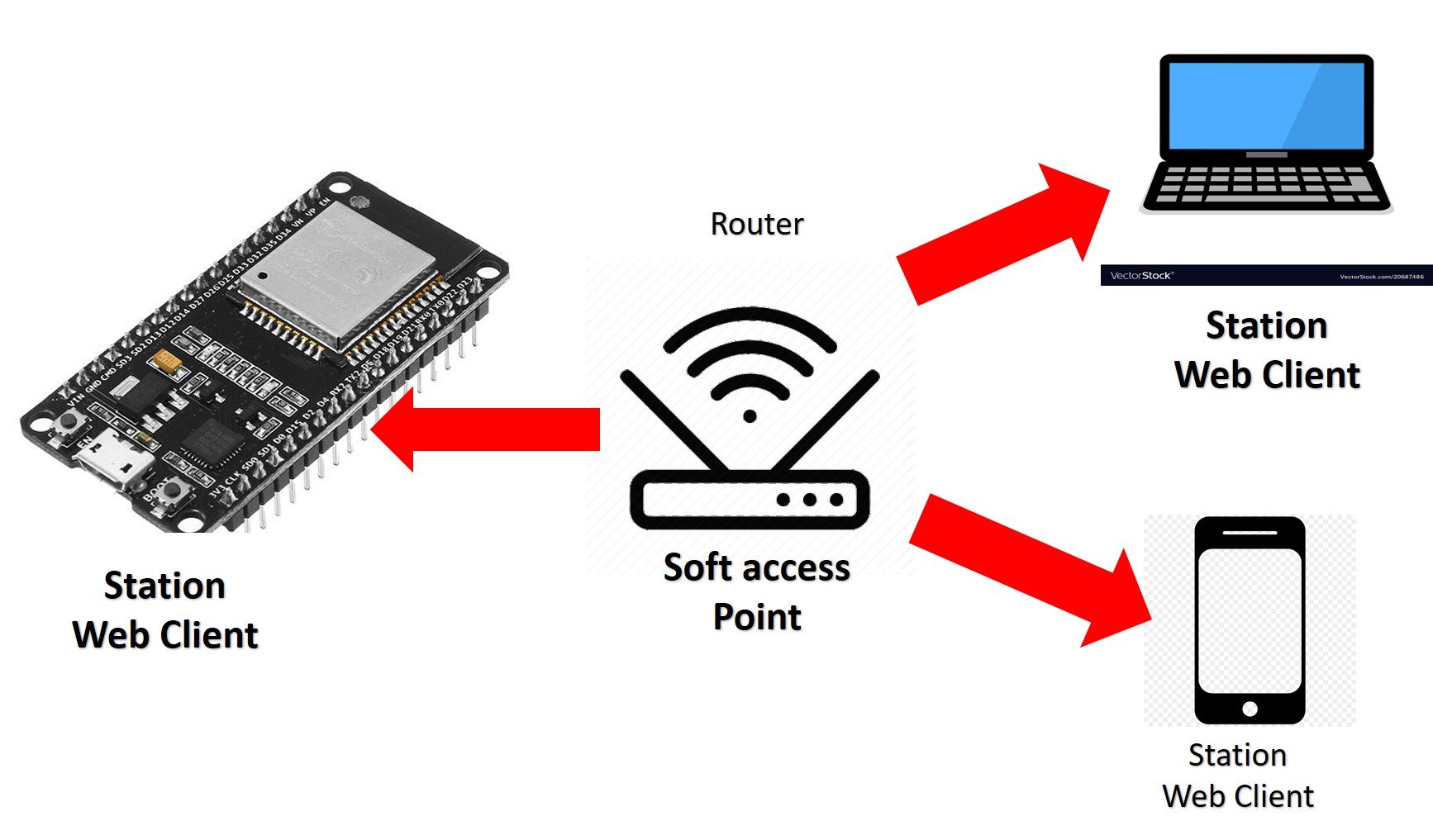
Après avoir reçu la requête HTTP GET, le serveur répond avec des pages Web. Le serveur Web répond également via le protocole HTTP. Si les fichiers Web ne sont pas disponibles sur le serveur ou s'il est en panne pour des raisons techniques, vous verrez un message d'erreur 404 sur le navigateur Web. Dans la figure ci-dessus, il n'y a qu'un seul client et un seul serveur. Mais un serveur peut se connecter à plusieurs clients Web et demandes de service via le protocole HTTP. Le serveur Web est également connu sous le nom d'hôte de serveur qui écoute les autres clients Web et traite les demandes.

#### **Mode de fonctionnement de l’ESP32**

Beaucoup de gens pensent qu'il ne peut être utilisé en mode station que s'il peut se connecter à un réseau existant. Mais ce module WiFi a des fonctionnalités très importantes pour utiliser un mode point d'accès logiciel, un mode station et les deux modes en même temps. Dans ce tutoriel, je parlerai uniquement du mode station. Dans les prochains tutoriels, j'écrirai également sur la façon d'utiliser cette carte en mode d'accès souple. Voyons maintenant la différence entre le mode station et le mode accès souple.

#### **Mode station ou STA**

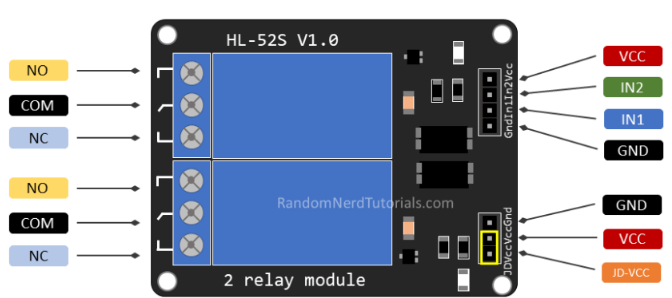
Dans ce mode, la carte ESP32 se connecte à votre réseau WiFi via un routeur. Ainsi, le moyen de communication entre le client Web et ESP32 est le routeur. Il obtient l'adresse IP du routeur Wifi. Avec cette adresse IP, les clients Web peuvent accéder au serveur Web via un réseau local existant. Et c’est le mode que nous allons utilisés pour notre mémoire.



**Figure 13: Mode station ou STA**

### Module de Relais 2 canaux

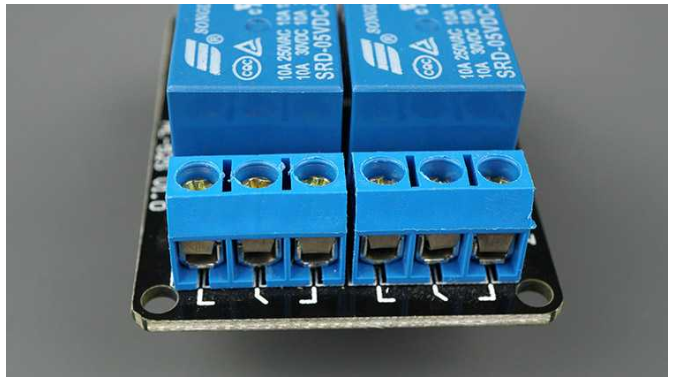
Le relais est un interrupteur pilotable qui permet d'isolé la partie commande de la partie puissance. C'est un composant très utilisé lorsqu'on veut faire le lien entre électronique et électricité. Il permet d'ouvrir ou fermer un contacteur sur un circuit de puissance en fonction d'un signal entre 0 et 5V. Il peut être contrôlé avec des basses tensions, comme le 3,3V fourni par les GPIO ESP32/ESP8266 et nous permet de contrôler des hautes tensions comme le 12V, 24V ou la tension secteur (230V en Europe et 120V aux États-Unis).



**Figure 14: Relais**

Sur le côté gauche, il y a deux ensembles de trois prises pour connecter les hautes tensions, et les broches sur le côté droit (basse tension) se connectent aux GPIO ESP.

#### Connexion de tension secteur



**Figure 15: Connexion de tension secteur**

Le module relais montré sur la photo précédente possède deux connecteurs, chacun avec trois prises : commun COM, Normalement fermé NC et normalement ouvert NON.

* **COM**: connectez le courant que vous souhaitez contrôler (tension secteur)
* **NC** (Normalement Fermé) : la configuration normalement fermée est utilisée lorsque l’on souhaite que le relais soit fermé par défaut. Les broches NC sont COM sont connectées, ce qui signifie que le courant circule à moins que vous n’envoyiez un signal de l’ESP au module de relais pour ouvrir le circuit et arrêter le flux de courant.
* **NON** ‘Normalement ouvert) : la configuration normalement ouverte fonctionne dans l’autre sens : il n’y a pas de connexion entre les broches NO et COM, donc le circuit est interrompu à moins que vous n’envoyiez un signal de l’ESP pour fermer le circuit.

Un module de relais a un total de 6 broches. VCC, GND, Signal, Normalement ouvert (NO), Normalement fermé (NC) et Contact commun (CC). Ici, nous avons utilisés un module à 2 relais, donc les broches VCC et GND sont les mêmes pour les deux relais. Les broches In1 et In2 sont respectivement les broches de signal du premier et du deuxième relais.

Les broches In1 et In2 sont respectivement connectées aux broches GPIO12 et GPIO13. Les numéros de broches des cartes ESP32 se trouvent sur la carte elle-même. La broche Common Connect doit être connectée à une alimentation électrique. Pour mes appareils électroménagers, j'utilise ici une alimentation secteur. Une ligne de l'alimentation secteur est directement connectée à l'appareil. L'autre ligne va dans la broche Common Connect. Maintenant, s'il faut amener une ligne de NO ou NC pour se connecter à l'appareil, cela dépend si vous voulez utiliser le relais comme "actif haut" ou "actif bas".

#### **Actif bas**

« Actif bas » signifie que le signal exécutera sa fonction lorsque son niveau logique est 0.

#### **Actif élevé**

« Actif bas » signifie que le signal exécutera sa fonction lorsque son niveau logique est 1.

Le port normalement ouvert (NO) est normalement ouvert lorsque l'entrée de la broche Signal est basse ou 0. Cela signifie que, si nous n'envoyons pas de signal positif à la broche Signal, l'appareil restera allumé s'il est connecté au Broche normalement ouverte, c'est-à-dire que le relais agit comme un interrupteur qui est activé. Mais dans le même cas, la broche normalement fermée reste fermée. Il agit comme un interrupteur qui est éteint. Maintenant, si nous envoyons un signal positif ou 1 dans la broche Signal, la broche Normalement Fermée agit comme un interrupteur qui est activé, la broche Normalement Ouverte agit comme un interrupteur qui est éteint.

#### **Utilisation de l'actif bas**

Maintenant, pour utiliser actif bas, nous devons connecter la broche normalement ouverte à l'appareil, car elle agit comme un interrupteur actif (un interrupteur qui est allumé) lorsque l'entrée sur la broche de signal est basse.

Pour garder l'interrupteur fermé, nous devons garder l'entrée dans la broche Signal aussi élevée (nous l'avons fait via le code).

#### **Point d'accès Wi-Fi**

On peut créer un point d'accès WiFi avec l'ESP32. Ensuite, nous pouvons connecter notre téléphone à ce point d'accès comme nous le connectons à n'importe quel WiFi. Nous pouvons ensuite héberger une page Web avec des commutateurs et une logique pour que ces commutateurs contrôlent les sorties de l'ESP32 qui entrent en tant qu'entrées dans les broches de signal du module de relais. Nous utiliserons la sortie de la broche GPIO12 de l'ESP32 comme entrée de la broche In1 du module de relais (broche de signal du premier relais) et la broche GPIO13 de l'ESP32 comme entrée de la broche In2 du module de relais (broche de signal du deuxième relais).

### Ampoule électrique

Une ampoule à incandescence aussi appelée lampe électrique est un dispositif complexe qui permet de produire de la [lumière](https://fr.vikidia.org/wiki/Lumi%C3%A8re) (énergie lumineuse) à partir d'une [énergie électrique](https://fr.vikidia.org/wiki/%C3%89nergie_%C3%A9lectrique).



**Figure 16: Lampe**

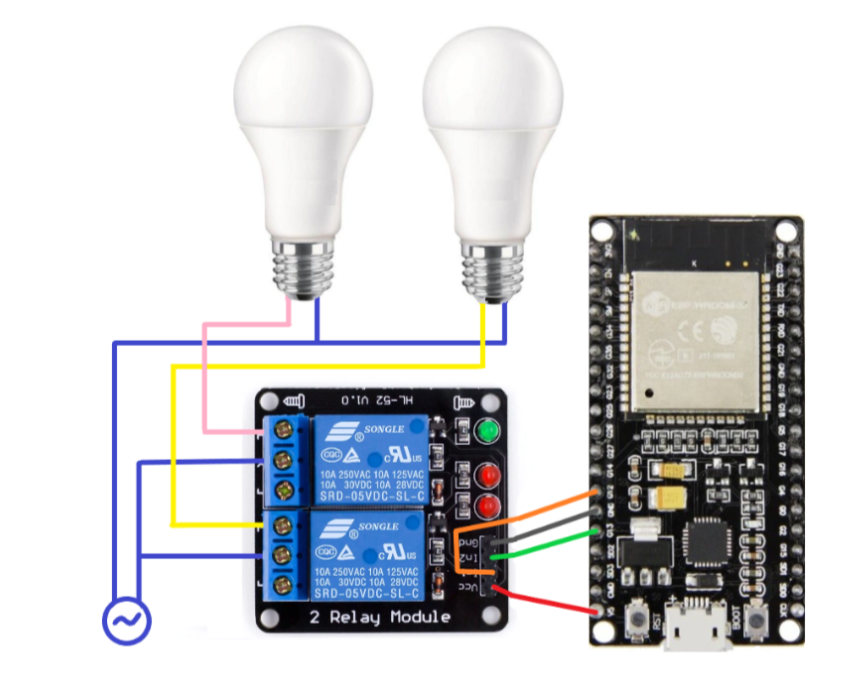
### Ordinateur portable (PC)

Machine automatique de traitement de l'information, obéissant à des programmes formés par des suites d'opérations arithmétiques et logiques.



**Figure 17: Un ordinateur portable**

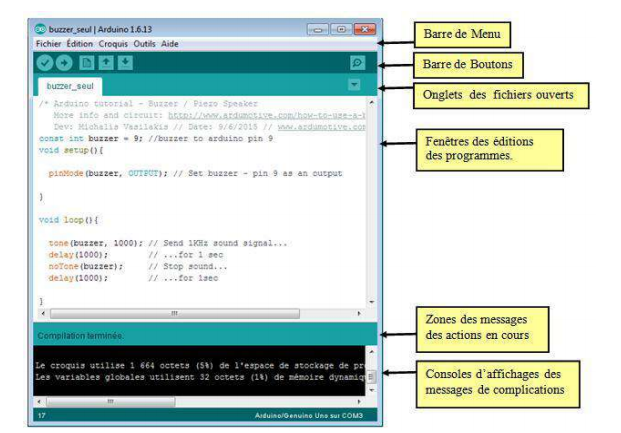
### Connexion entre l’esp32 et la lampe



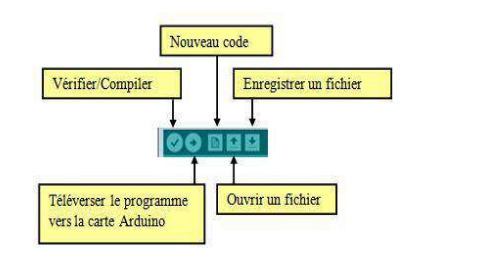
**Figure 18 : La connexion entre l’esp32 et la lampe**

## Plateforme de programmation Arduino

La façon dont le microcontrôleur gère ses entrées / sorties est fixée par un programme, contenu dans le microcontrôleur. Ce programme doit être écrit par l’utilisateur. En pratique, l’utilisateur écrit le programme en langage C, en utilisant un environnement de développement spécialisé (IDE) installé sur un ordinateur. Ce programme est ensuite compilé et téléversé dans le microcontrôleur par liaison série (USB). Microcontrôleur Arduino : Guide de Base 11 Nous utiliserons l’IDE standard Arduino (arduino.exe). Il suffit de taper le code dans la fenêtre dédiée, de compiler et de téléverser le programme sur la carte Arduino. La carte doit être reliée à l’ordinateur par un câble USB. La modèle de la carte Arduino (il y a plusieurs types de carte) ainsi que le port série sur lequel elle est branchée doivent être déclarés dans le menu de l’IDE Outils/type de carte et Outils/port série.



**Figure 19: Plateforme Arduino**



Une fois téléversée dans le microcontrôleur, le programme s’exécute.

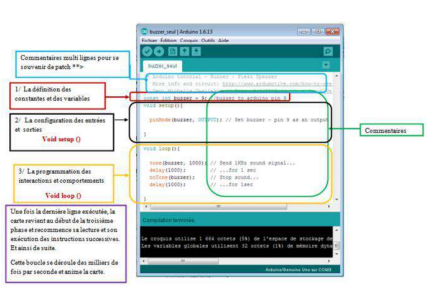
La fonction « Setup » contiendra toutes les opérations nécessaires à la configuration de la carte (directions des entrées sorties, débits de communications série, etc.). La fonction « Loop », elle est exécutée en boucle après l’exécution de la fonction setup. Elle continuera de boucler tant que la carte n’est pas mise hors tension, redémarrée (par le bouton reset). Cette boucle est absolument nécessaire sur les microcontrôleurs étant donné qu’ils n’ont pas de système d’exploitation. En effet, si l’on omettait cette boucle, à la fin du code produit, il serait impossible de reprendre la main sur la carte Arduino qui exécuterait alors du code aléatoire.

Le logiciel comprend aussi un moniteur série (Equivalent à HyperTerminal) qui permet d'afficher des messages textes émis par la carte Arduino et d'envoyer des caractères vers la carte Arduino (en phase de fonctionnement)



**Figure 20: HyperTerminal de l’Arduino (Moniteur Série)**

Un programme utilisateur Arduino est une suite d’instructions élémentaires sous forme textuelle, ligne par ligne. La carte lit puis effectue les instructions les unes après les autres, dans l’ordre défini par les lignes de code. La structure d’écriture d’un programme sous Arduino est de la forme suivante :

: ****

**Figure 21: Structure générale du programme (IDE Arduino**

### Langage Arduino

Le langage Arduino est inspiré de plusieurs langages. On retrouve notamment des similarités avec le C, le C++, le Java et le Processing. Le langage impose une structure particulière typique de l’informatique embarquée. La fonction setup contiendra toutes les opérations nécessaires à la configuration de la carte (directions des entrées sorties, débits de communications série, etc.).

## Le langage Programmation PYTHON

Python est le langage de programmation le plus utilisé dans le domaine du Machine Learning, du Big Data et de la Data Science.

Créé en 1991, le langage de programmation Python apparu à l’époque comme une façon d’automatiser les éléments les plus ennuyeux de l’écriture de scripts ou de réaliser rapidement des prototypes d’applications.

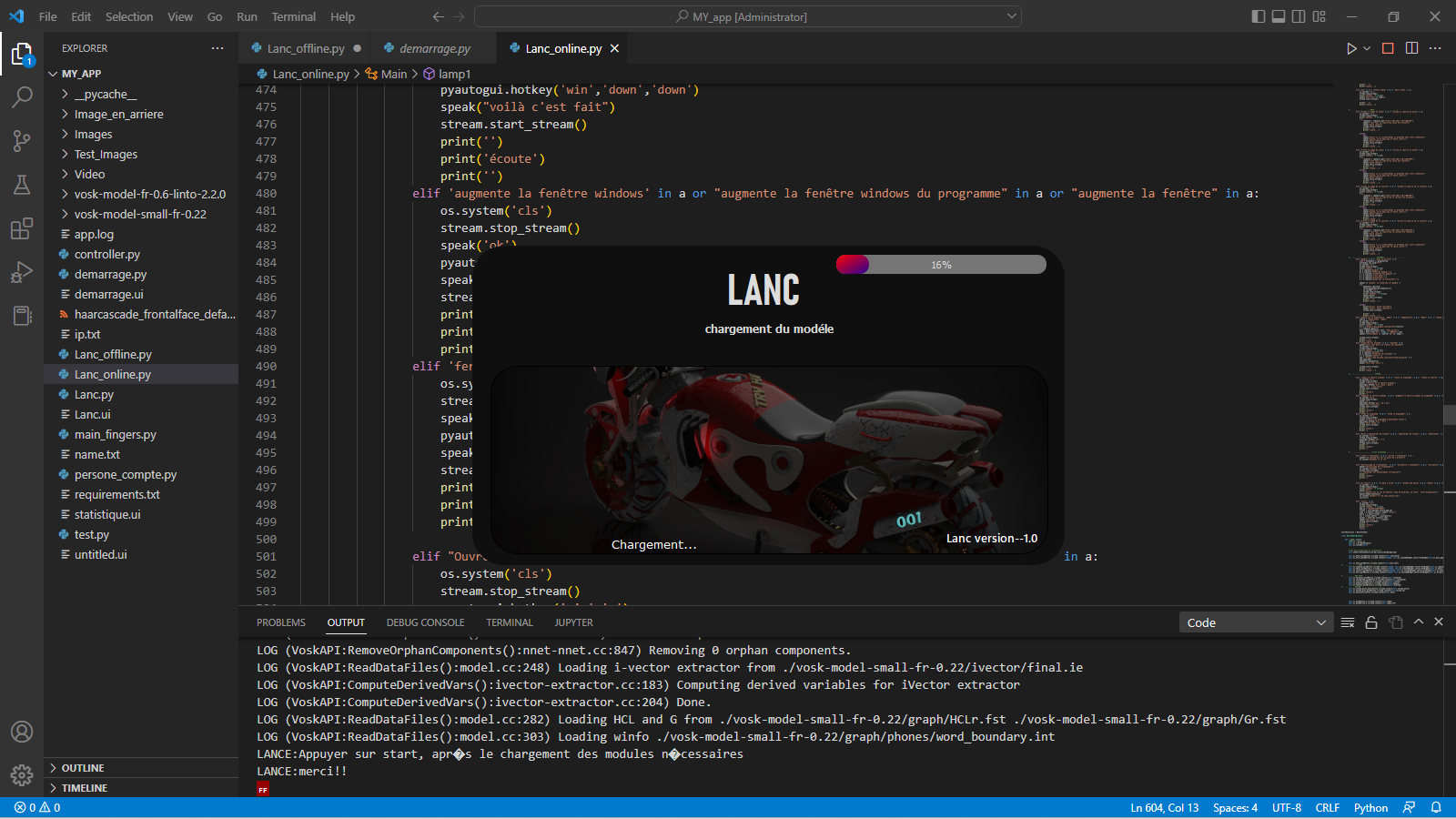
Python est un langage de programmation interprété, orienté objet et de haut niveau avec une sémantique dynamique. Les structures de données de haut niveau intégrées combinées au typage et à la liaison dynamiques le rendent extrêmement utile non seulement pour le développement rapide d'applications, mais également pour une utilisation en tant que langage de script ou de liaison pour lier des composants existants attrayants. La syntaxe simple et facile à apprendre de Python met l'accent sur la lisibilité, réduisant ainsi les coûts de maintenance du programme. Python prend en charge les modules et les packages qui favorisent la modularité du programme et la réutilisation du code. L'interpréteur Python et la vaste bibliothèque standard sont disponibles gratuitement sous forme source ou binaire pour toutes les principales plates-formes et peuvent être librement distribués. Les programmeurs tombent souvent amoureux de Python en raison de l'augmentation de productivité qu'il offre. Comme il n'y a pas d'étape de compilation, le cycle édition-test-débogage est très rapide. Le débogage des programmes Python est facile. Une erreur ou une mauvaise entrée ne causera pas d'erreur de segmentation. Au lieu de cela, une exception est levée si l'interpréteur rencontre une erreur. Si votre programme n'attrape pas d'exception, l'interpréteur imprime une trace de pile. Le débogueur au niveau de la source vous permet d'inspecter les variables locales et globales, d'évaluer des expressions arbitraires, de définir des points d'arrêt, de parcourir le code ligne par ligne, etc. Le débogueur est écrit en Python lui-même, un témoignage de l'introspection de Python. D'un autre côté, le moyen le plus rapide de déboguer un programme consiste souvent à ajouter des instructions d'impression à la source.

## Tests et analyses

Dans cette partie nous parlerons de l’aspect analyse, simulation, conception

### L’interface Graphique

Lorsque l’application démarre certains modules doivent êtres charger en arrière-plan notamment, les model entrainés et les différentes bibliothèques d’où la nécessité de cet écran de chargement. Alors pour rappel cette interface a été développé sous Python PYQT5

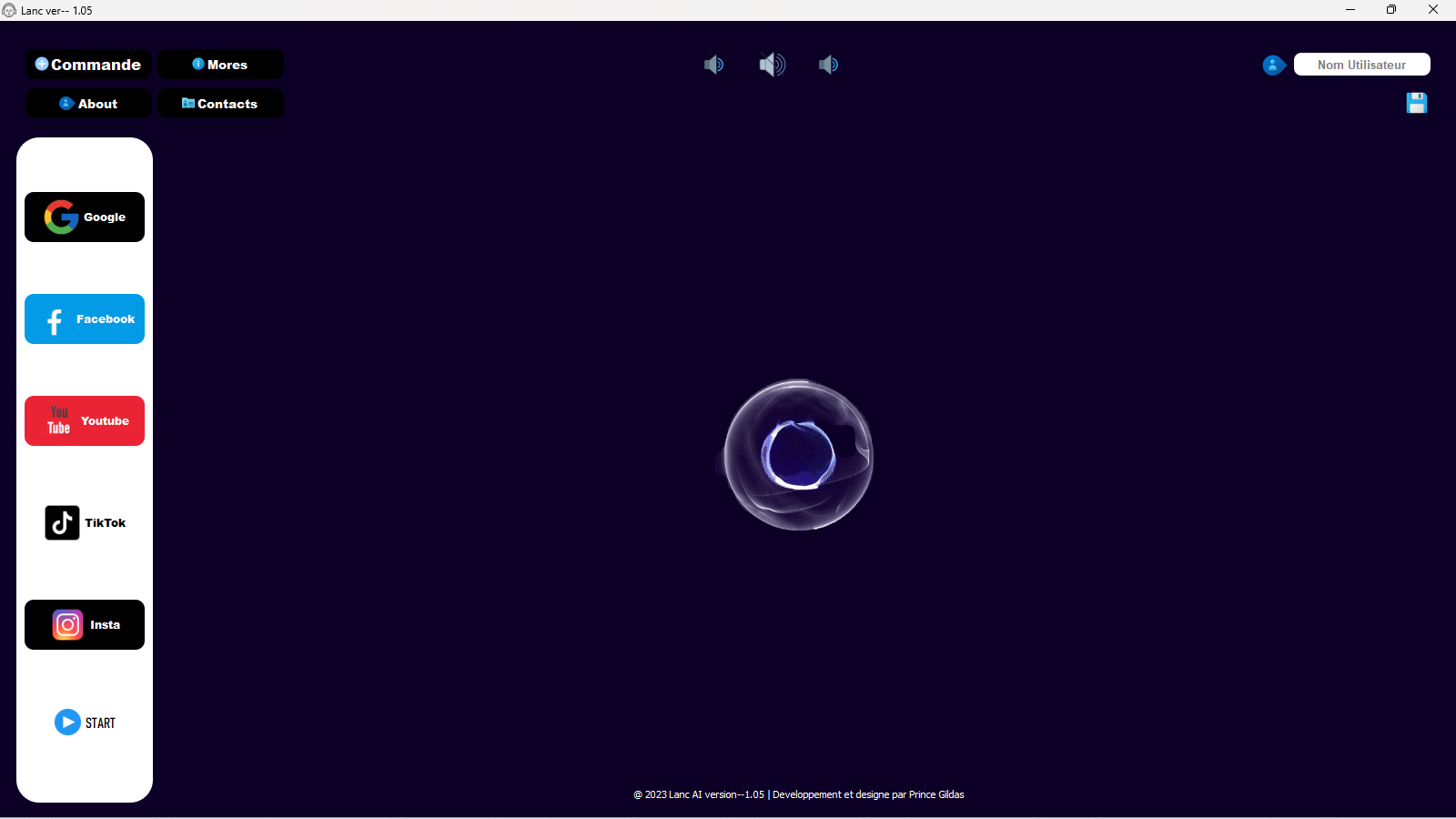


Ensuite vous devez apparaitre devant cette page, pour commencer avec LANC il suffit d’appuyer sur le Bouton « START ».

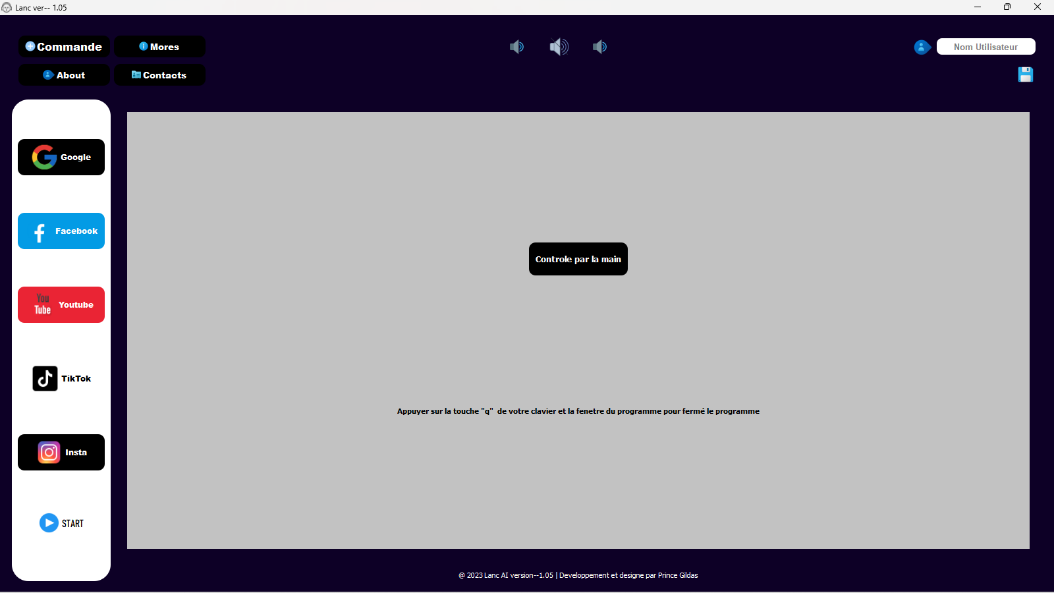
En bleu vous avez les boutons volume plus à droite, mute au milieu et volume moins à gauche.

En rouge vous avez des boutons qui vont vous rediriger vers vos réseaux sociaux préférés.

En blanc vous avez 4 boutons, commande, mores, about, contacts.



La page affichée ci-dessous est celle appelé « Mores ». Ici vous avez la possibilité de contrôlé les lampes en utilisant vos doigts ce domaine s’appelle La vision par ordinateur : nous en avons parlé au paravent.



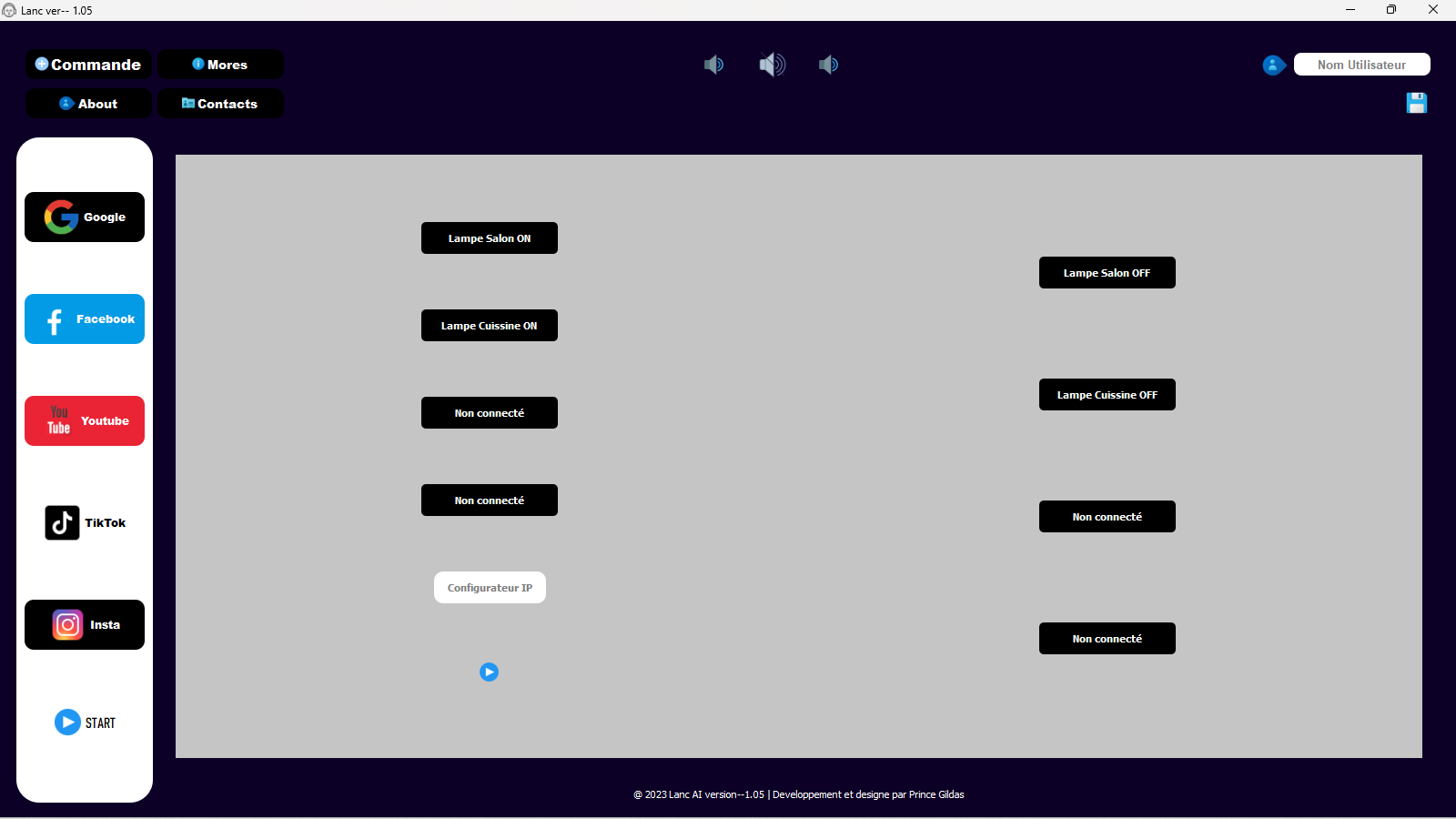
Voici à quoi le programme de contrôle par les mains l’erreur affiché, l’erreur affiché intervient lorsque vous essayer d’ouvrir le programme de contrôle par les mains mais le programme n’arrive pas à accéder à la page web correspondante.



Ensuite vous avez la page Commande.

En rouge seul ces 2 boutons fonctionnent pour le moment, l’utilisateur à la possibilité de contrôler les lampes avec des boutons.

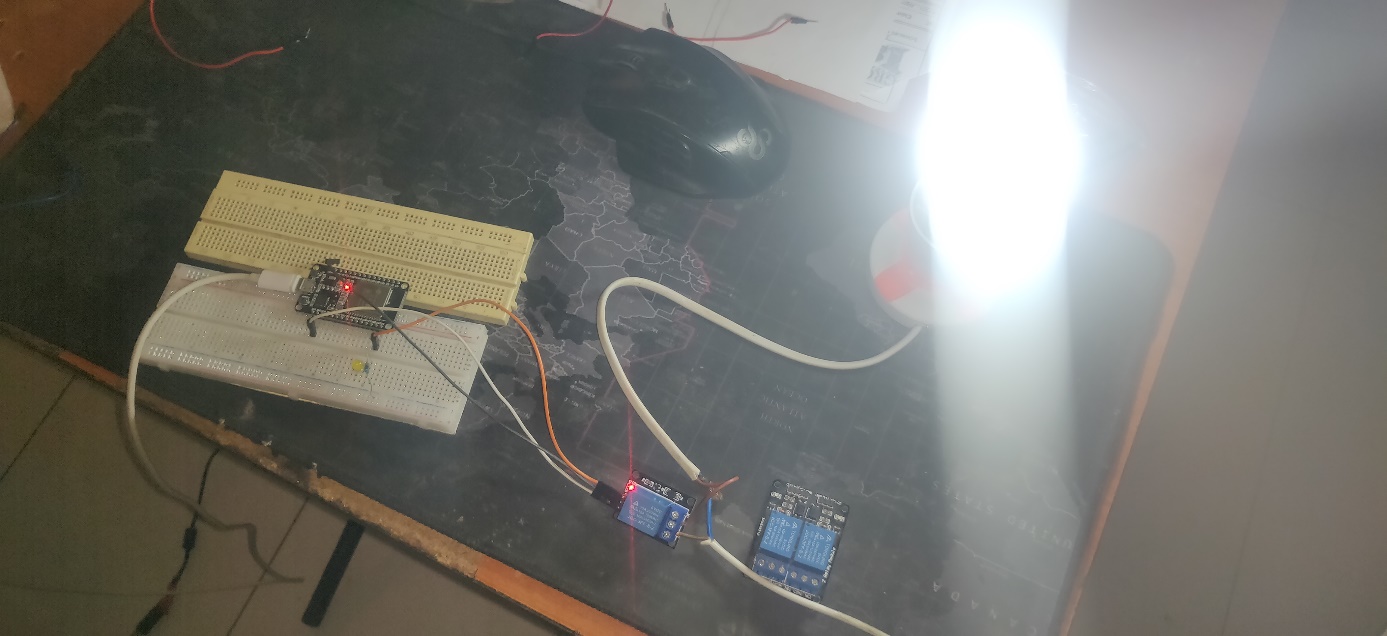
En jaune, cette option a été ajouté au cas ou il faille modifier l’adresse de connexion il suffit juste de rentrer l’adresse IP de connexion fourni lors du téléversement du code dans la carte programmable ex : 192.168.1.13 et ensuite appuyer sur le bouton en bleu NB : pas besoin de redémarré le programme.



### Pratique

Ci-dessous quelques tests notés que le système final est susceptible de changer merci.

En bleu nous avons m’ESP32 et en route le module relay.



# Conclusion générale

En définitive, ce présent mémoire rentre dans le cadre de la conception d’un système intégrant l’IA pour la commande de l’alimentation des charges électro-ménager. Ainsi dans notre travail, notre problématique se résumait à savoir comment mettre en place ce système ? C’est dans cette optique que nous nous sommes fixés comme objectif général la conception d’un système intégrant IA et la reconnaissance vocale aux vues de réaliser des tâches préprogrammées et de répondre à des questions préprogrammées.

Nous pouvons aussi définir ce système comme étant un assistant Virtual tout comme ALEXA, SIRI, GOOGLE Assistant, CORTANA il permet d’effectuer des tâches prédéfinis par l’homme. L’avantage est que ce genre de système permettent de gagner un temps considéra sur les taches du quotidien, il fonctionne avec et sans connexion, les délais de réponses sont considérablement réduits, l’inconvénient est qu’ils ne sont pas très réactifs par moment du aux bruit ambiant et nécessite beaucoup l’intervention humaine. Pour notre système ceci n’est juste qu’un prototype d’autres fonctionnalités seront ajoutés. Grace à ce projet nous avons pu toucher à plusieurs technologies dont nous n’avions pas connaissance.

Néanmoins, il a pu voir le jour grâce à notre persévérance, beaucoup de nuits blanches et beaucoup de recherche sur internet et en ce qui concerne le domaine de l’intelligence artificiel la majorité de la documentation est en anglais ceux qui peut faire abandonner certain.

La prochaine étape est de miniaturisé ce système pour le rendre plus compact et l’intégrée par exemples sur des robot et pouvoir les contrôlés seulement avec la voix.

# Bibliographies

"Conception et mise en œuvre d'un système de commande domotique intelligent utilisant le traitement automatique de la langue naturelle" par M. L. Bouzid et al. (2017)

"Interface de langage naturel pour la commande de la maison intelligente" par S. B. Ndiaye et al. (2015)

"Développement d'un système domotique vocal avec Raspberry Pi et Google Assistant" par A. K. Diallo et al. (2018)

"Système d'automatisation de la maison intelligente utilisant le traitement automatique de la langue naturelle" par N. D. Ngom et al. (2019)

"Conception et mise en œuvre d'un système d'automatisation de la maison intelligente utilisant la reconnaissance vocale" par A. Traoré et al. (2018)

"Design and Implementation of an Intelligent Home Control System Using Natural Language Processing" par Hongda Su et al. (2017)

"A Natural Language Interface for Smart Home Control" par Jie Liu et al. (2016)

"Development of a Voice-Controlled Smart Home System Using Raspberry Pi and Google Assistant" par Hong-Seok Kim et al. (2019)

"An Intelligent Home Automation System Using Natural Language Processing" par R. S. Sindhuja et al. (2018)

"Design and Implementation of an Intelligent Home Automation System Using Voice Recognition" par Md. Hasanuzzaman et al. (2016)

"Building a Jarvis-Like AI Using Python" par Rishabh Jain (2018)

"Creating a Jarvis-like AI Assistant Using Python" par Shubham Sarda (2018)

"Building your own AI assistant like Jarvis in Python" par Anirudh Jayaraman (2017)

"Creating a Jarvis-like AI in Python" par Aakash N S (2018)

"Designing a Jarvis-inspired AI assistant using Python" par Rajat Sharma (2019)

"Creating a Jarvis-like AI using Python" par Rajat Gupta (2018)

"Designing a Personal AI Assistant like Jarvis using Python" par Pritesh Patel (2019)

"Building a Jarvis-inspired AI using Python" par Shashank Sharma (2019)

"Creating a Virtual Assistant like Jarvis using Python" par Neha Sharma (2020)

"Designing a Personal AI Assistant like Jarvis in Python" par Ankit Jain (2020)

"Creating a Jarvis-like AI in Python" par Shubham Patel (2020)

"Building a Jarvis-inspired Virtual Assistant using Python" par Rajat Gupta (2020)

"Designing a Personal AI Assistant similar to Jarvis using Python" par Ankit Sharma (2021)

"Creating a Jarvis-like AI using Python" par Shubham Jain (2021)

"Building a Jarvis-inspired AI using Python" par Pritesh Patel (2021)

# Webographie

[**https://randomnerdtutorials.com/esp32-servo-motor-web-server-arduino-ide/**](https://randomnerdtutorials.com/esp32-servo-motor-web-server-arduino-ide/)22/06/2022

[**https://randomnerdtutorials.com/esp32-esp8266-micropython-web-server/**](https://randomnerdtutorials.com/esp32-esp8266-micropython-web-server/)09/09/2022

[**https://github.com/BolisettySujith/J.A.R.V.I.S/blob/main/JARVIS.py**](https://github.com/BolisettySujith/J.A.R.V.I.S/blob/main/JARVIS.py)21/08/2022

[**https://github.com/ThierrySilva/Flutter-Arduino-Bluetooth**](https://github.com/ThierrySilva/Flutter-Arduino-Bluetooth)21/09/2022

[**https://blog.logrocket.com/adding-speech-to-text-text-to-speech-support-flutter-app/**](https://blog.logrocket.com/adding-speech-to-text-text-to-speech-support-flutter-app/)01/07/2022

[**https://www.codewithharry.com/videos/python-tutorials-for-absolute-beginners-120/**](https://www.codewithharry.com/videos/python-tutorials-for-absolute-beginners-120/)21/09/2022

[**https://github.com/nateshmbhat/pyttsx3**](https://github.com/nateshmbhat/pyttsx3)21/09/2022

[**https://towardsdatascience.com/wikipedia-api-for-python-241cfae09f1c**](https://towardsdatascience.com/wikipedia-api-for-python-241cfae09f1c)21/09/2022

[**https://www.raspberrypi.com/documentation/computers/configuration.html**](https://www.raspberrypi.com/documentation/computers/configuration.html)21/07/2022

[**https://roboticsbackend.com/control-arduino-with-python-and-pyfirmata-from-raspberry-pi/**](https://roboticsbackend.com/control-arduino-with-python-and-pyfirmata-from-raspberry-pi/)

[**How to Access the Raspberry Pi Desktop with a Remote Desktop Connection**](https://www.circuitbasics.com/access-raspberry-pi-desktop-remote-connection/) 21/07/2022

[**speech\_recognition/library-reference.rst at master · Uberi/speech\_recognition · GitHub**](https://github.com/Uberi/speech_recognition/blob/master/reference/library-reference.rst) 21/09/2022

[**An Introduction to pyttsx3: A Text-To-Speech Converter for Python | by Ng Wai Foong | Better Programming**](https://betterprogramming.pub/an-introduction-to-pyttsx3-a-text-to-speech-converter-for-python-4a7e1ce825c3)22/08/2022

[**https://github.com/dreymond1/a-simple-python-virtual-assistant**](https://github.com/dreymond1/a-simple-python-virtual-assistant)05/09/2022

[**https://realpython.com/python-speech-recognition/**](https://realpython.com/python-speech-recognition/)30/07/2022

[**https://ireadblog.com/posts/141/how-to-build-your-personal-ai-assistant-using-python**](https://ireadblog.com/posts/141/how-to-build-your-personal-ai-assistant-using-python)30/09/2022

# Annexes

**Le code :**

from vosk import Model, KaldiRecognizer

import pyttsx3

import pyaudio

import msvcrt as m

import random

import os

import json

def wait():

    m.getch()

engine = pyttsx3.init("sapi5")

voices = engine.getProperty("voices")

rate = engine.setProperty("rate",195)

try:

    engine.setProperty('voice', voices[4].id)

except ValueError:

    engine.setProperty("voice", voices[3].id)

except TypeError:

    engine.setProperty("voice", voices[2].id)

except :

    engine.setProperty("voice", voices[0].id)

def speak(audio):

    print ("LANCE:" + str(audio)) #pour prototype 1

    engine.say(audio)

    engine.runAndWait()

model = Model("vosk-model-small-fr-0.22")

rec = KaldiRecognizer(model, 16000)

cap = pyaudio.PyAudio()

stream=cap.open(format=pyaudio.paInt16, channels=1, rate=16000, input=True, frames\_per\_buffer=8192)

stream.start\_stream()

while True:

    data = stream.read(4000, exception\_on\_overflow=False)

    if len(data) == 0:

        break

    if rec.AcceptWaveform(data):

        result = rec.Result()

        result = json.loads(result)

        print ("speaker: "+ result["text"])

        a = result["text"]

        if "comment tu vas" in a or "comment vas-tu" in a or "ça va toi" in a:

            os.system('cls')

            stream.stop\_stream()

            print('speaker: '+ str(a))

            a = ["merci, je vais bien","pas trop mal merci","ça peux aller","je n'ai pas vraiment d'émotions, mais ça peux aller"]

            speak(random.choice(a))

            stream.start\_stream()

            print("...")

            print('écoute...')

        elif 'bonjour' in a or 'salut' in a or 'hello'in a or 'coucou' in a :

            reply\_bonjour = ['bonjour bonjour!', 'hello, comment vous allez?',"bonjour à vous!", 'bonjour,comment vous allez?']

            os.system('cls')

            stream.stop\_stream()

            print('speaker: '+ str(a))

            speak(random.choice(reply\_bonjour))

            stream.start\_stream()

            print("...")

            print('écoute...')

        elif "je t'admire" in a or "tu es tellement" in a:

            reply\_aime = ["Je sais", "j'apprécie votre sentiment", "merci c'est gentil", "ohh merci beaucoup", "vous de meme"]

            os.system('cls')

            stream.stop\_stream()

            print('speaker: '+ str(a))

            speak(random.choice(reply\_aime))

            stream.start\_stream()

            print("...")

            print('écoute...')

        elif "je vais bien" in a or "je vais bien " in a or "ça va" in a or 'ça peut aller' in a:

            reply = ["tant mieux", "heureux de l'entendre", "cool alors", "super, et sinon comment je peux t'aider?", "ravis de l'entendre, et sinon comment puis-je t'aider?"]

            os.system('cls')

            stream.stop\_stream()

            print('speaker: '+ str(a))

            speak(random.choice(reply))

            stream.start\_stream()

            print("...")

            print('écoute...')

        elif "je ne vais pas bien" in a  or "bof" in a or "ça ne va pas bien" in a :

            reply = ["pourquoi?", "désolé pour vous",]

            os.system('cls')

            stream.stop\_stream()

            print('speaker: '+ str(a))

            speak(random.choice(reply))

            speak("je vous suggére d'allez prendre un peu d'air")

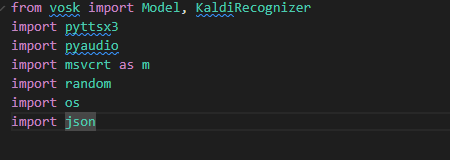
            stream.start\_stream()

            print("...")

            print('écoute...')

Noté que ce code n’est pas définitif, des changements peuvent être effectués.

**Explication du code :**

****

Dans un premier temps, il faut importer les différentes bibliothèques qui vont nous aider dans ce projet :

Au préalable il faut bien sur installer les différentes bibliothèques, python et un éditeur de code pour ma part c’est Visual Studio Code

**Vosk** : pour son utilisation il faut importer un model au préalable qu’il faut télécharger sur le site de Vosk <https://alphacephei.com/vosk/models>. Permet la conversion de parole en texte

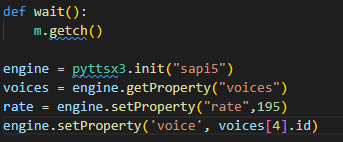
**Pyttsx3** il va servie de conversion en texte-parole

**Pyaudio** il va permettre d’utiliser les différents ports audios

***Msvrt***: **pip install getch** Le module getch effectue une entrée à caractère unique (ceci est donné par la documentation officiel de vosk)

**Random** : **pip install random** ce module permet d’avoir des résultats aléatoires

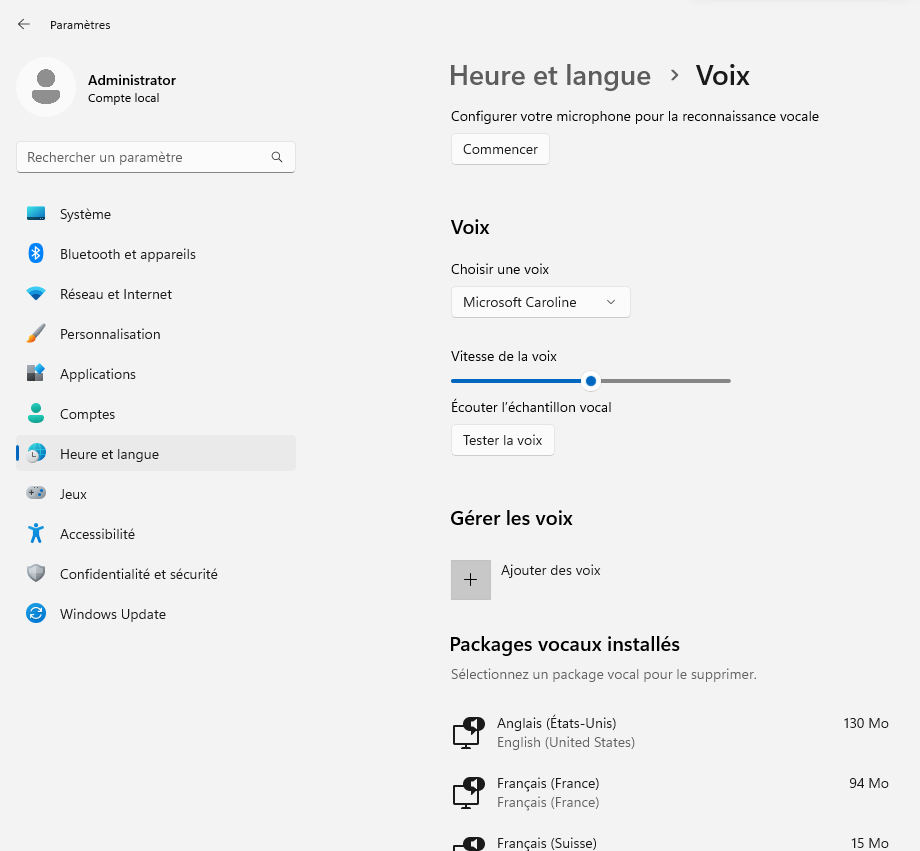
**OS** : **pip install os** il permet d’interagir avec les fichiers du système

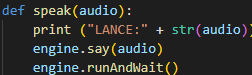


On initialise le moteur en spécifiant sur quel système nous nous trouvons, pour nos cas cas il s’agit de SAPI5. Si nécessaire vous pouvez changer d’autres paramètres en vous basant sur la documentation officielle <https://pyttsx3.readthedocs.io/en/latest/engine.html#the-engine-factory>.

Engine.setProperty(‘voice’,voices[4].id le 4 permet de spécifié quel genre de voix on veut les numéro peuvent changer en fonction du par de voix installé sur votre système d’exploitation par défaut vous pouvez choisir entre 0 et 1 0 pour la femme et 1 pour l’homme .

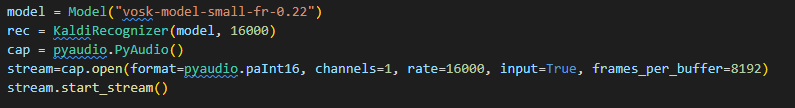
Les packs de voix peuvent être installés depuis vos paramètres Windows. Pour ma part sur Windows 11 paramètres/Heure et langue/Voix/Packages vocaux installés.



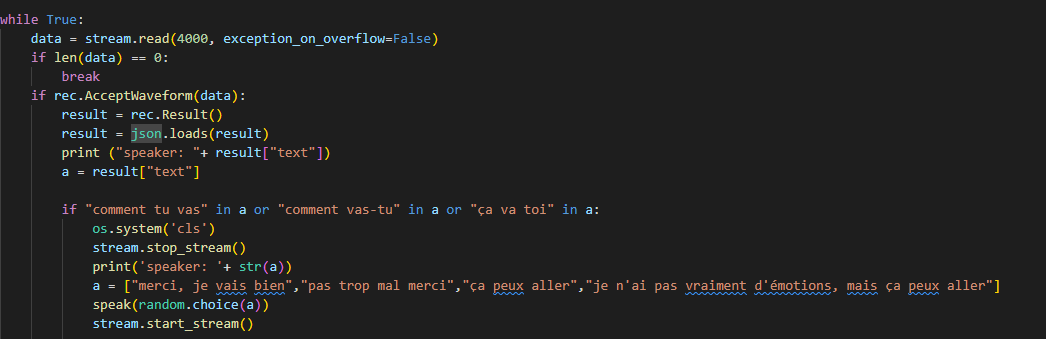
****

Cette ligne de code permet au système de répondre à vos questions par la parole.

Le Print (‘LANCE…………..) est un ajout personnel.

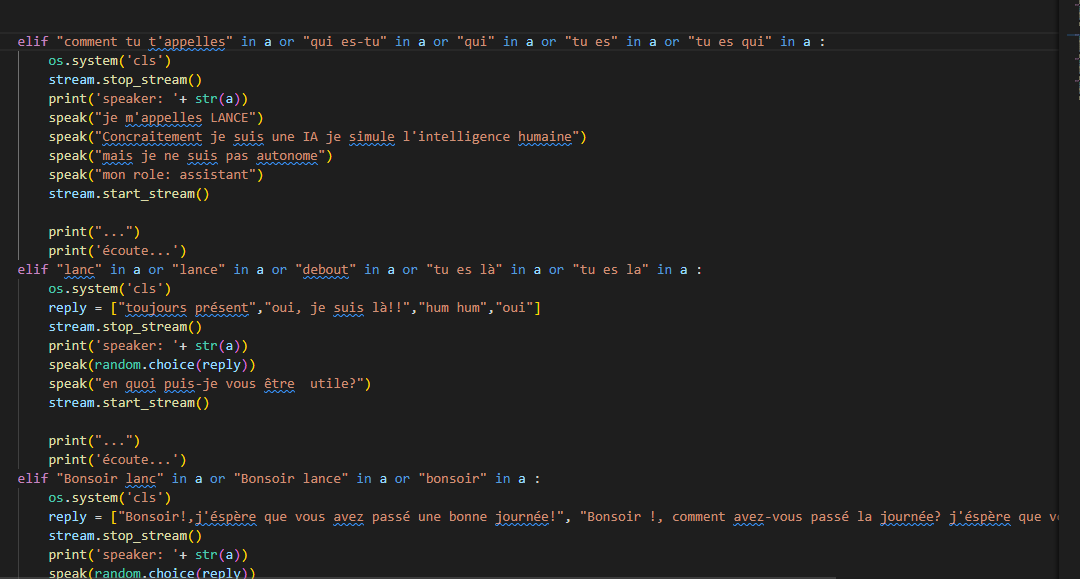


Ici il s’agit de spécifier l’emplacement du model au préalable télécharger.



Tout le reste du code va se trouver dans une boucle pour que le système fonctionne sans interruption (ce référé à la documentation officielle)

Os.system(‘cls ‘) cette ligne n’est pas nécessaire juste un ajout personnel qui permettre au système d’un réactive.



elif "allume la lampe du salon" in a or "allume la lumière du salon" in a:

            os.system('cls')

            stream.stop\_stream()

            print('speaker: '+ str(a))

            #url = 'http://192.168.1.10/humidity'

            #webbrowser.open(url)

            speak("c'est fait la lumière du salon est allumée")

            #os.system("taskkill /im chrome.exe /f")

            speak("autre chose?")

            stream.start\_stream()

            print("...")

            print('écoute...')

Voici un simple exemple de code pour contrôler nos ampoules

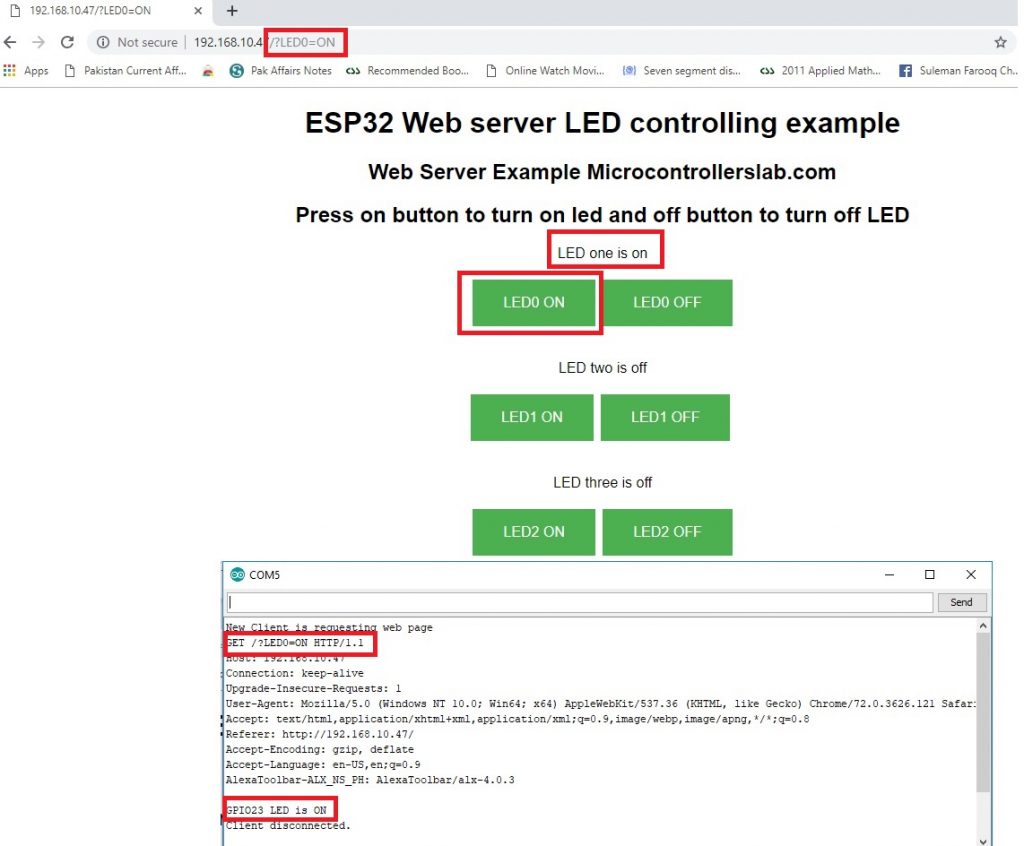
Prix des composants :

|  |  |  |
| --- | --- | --- |
|  | Prix | Quantité |
| Relais | 3000 | 2 |
| Esp32 | 6000 | 1 |
| Douille | 500 | 2 |
| Lampe | 1000 | 2 |
| Cable électrique | - | - |

Le code pour contrôler à lampe à distance :

**Contrôle des LED à partir de la page Web**

Lorsque vous appuyez sur un bouton LED0, vous verrez que l'adresse IP contiendra désormais un texte supplémentaire "/?LED0=ON" et ce message sera envoyé à la carte ESP32. vous verrez également le message dans la chaîne "GET /?LED0=ON HTTP/1.1″ . Ainsi, ESP32 recevra cette demande du client Web et allumera le LED0 que nous avons spécifié dans le code ci-dessus. J'expliquerai cette partie dans la section « Comment fonctionne le code ». vous remarquez également que l'état de la LED sera affiché sur la page Web. De même, vous pouvez tester d'autres boutons et vérifier la même réponse.

[](https://microcontrollerslab.com/wp-content/uploads/2019/03/Webserver-LED0-ON.jpg)

Vous devez donc suivre ces points lors du test du serveur Web.

* Tout d'abord, vous devez appuyer sur chaque bouton et vérifier si vous voyez le même message dans l'URL et également sur le moniteur série de l'IDE Arduino.
* Vérifiez également si l'état du message LED change en fonction du bouton sur lequel vous avez appuyé sur la page Web.
* Voir également si la LED respective bascule en fonction du bouton enfoncé.

**Comment fonctionne le code du serveur Web ESP32 ?**

Nous allons maintenant voir comment fonctionne ce code et je vais vous fournir des détails sur la fonction et chaque ligne du code.

WiFi.h est une bibliothèque de modules WiFi. Nous inclurons cette bibliothèque car nous utiliserons la fonction WiFi pour se connecter à un réseau, envoyer des données au client et recevoir des données du client. Cette bibliothèque est pré-installée dans Arduino IDE. Donc, nous allons simplement l'inclure avec la directive #include.

#include <WiFi.h>

**Connectez-vous au Wi-Fi**

Ces deux variables servent à stocker le nom et le mot de passe du réseau auquel vous souhaitez connecter votre carte ESP32. Vous devez remplacer « PTCL-BBB » par le nom de votre réseau de féodalité sans fil et également mettre le mot de passe dans la variable WIFI\_password.

const char\* WIFI\_NAME= "PTCL-BB";

const char\* WIFI\_PASSWORD = "5387c614";

Chaque fois que nous définissons le serveur Web, lorsqu'il est nécessaire de définir son nom de port, cette ligne définira le WiFiServer () sur le port 80.

WiFiServer server(80);

Ce nom de variable de chaîne est utilisé pour stocker toutes les données reçues par la carte de développement ESP32 via des requêtes HTTP.

String header;

**Définir les LED**

Comme je l'ai mentionné, nous basculons trois LED dans ce tutoriel, donc ces trois variables sont utilisées pour stocker l'état de ces trois diodes électroluminescentes. Initialement, nous définissons l'état des LED sur l'état éteint. Mais ces états changeront en fonction de l'état des LED.

String LED\_ONE\_STATE = "off";

String LED\_TWO\_STATE = "off";

String LED\_THREE\_STATE = "off";

Les broches GPIO 22, 23 et 15 sont utilisées avec les LED. Ainsi, ces trois déclarations donneront le nom à ces broches GPIO. Chaque fois que nous voulons faire référence à ces broches dans notre code, nous les utiliserons par ces noms au lieu du numéro de broche.

const int GPIO\_PIN\_NUMBER\_22 = 22;

const int GPIO\_PIN\_NUMBER\_23 = 23;

const int GPIO\_PIN\_NUMBER\_15 = 15;

Dans la fonction setup() d'abord, nous avons initialisé le débit en bauds de la fonction 'serial.begin' à la vitesse de 115200 bits par seconde.

Serial.begin(115200);

Maintenant, nous définissons ces broches GPIO comme une sortie numérique à l'aide de la fonction pinMode () et définissons également ces broches de sortie d'entrée à usage général sur numériquement bas à l'aide de la fonction digitalWrite ().

pinMode(GPIO\_PIN\_NUMBER\_22, OUTPUT);

pinMode(GPIO\_PIN\_NUMBER\_23, OUTPUT);

pinMode(GPIO\_PIN\_NUMBER\_15, OUTPUT);

digitalWrite(GPIO\_PIN\_NUMBER\_22, LOW);

digitalWrite(GPIO\_PIN\_NUMBER\_23, LOW);

digitalWrite(GPIO\_PIN\_NUMBER\_15, LOW);

Maintenant, nous allons discuter de certaines des fonctions de la bibliothèque WiFi qui sont utilisées dans ce cours. Pour que vous puissiez comprendre le code plus clairement.

**Connexion au Wi-Fi**

Tout d'abord, il imprimera un message "connexion à" avec le nom WiFi. Après cela, il se connectera à un réseau. S'il est correctement connecté à un réseau wifi. vous verrez une adresse IP et réussirez à vous connecter à un message de réseau Wifi sur le moniteur série. Pendant qu'il essaie de se connecter à Internet, il affichera "essayer de se connecter au réseau wifi". La fonction Server.begin () démarre le serveur et maintenant n'importe quel client Web peut accéder à la page Web avec cette adresse IP.

Serial.print("Connecting to ");

Serial.println(WIFI\_NAME);

WiFi.begin(WIFI\_NAME, WIFI\_PASSWORD);

while (WiFi.status() != WL\_CONNECTED)

{

delay(1000);

Serial.print("Trying to connect to Wifi Network");

}

Serial.println("");

Serial.println("Successfully connected to WiFi network");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

server.begin();

La boucle principale sert la page Web au client et reçoit également des données via HTTP get request sur l'état des broches GPIO. À l'intérieur de la fonction de boucle, cette ligne est utilisée pour recevoir une demande de nouveaux clients Web. Si la demande du client est disponible, la fonction server.avaialble() stocke une valeur logique dans la variable client.

WiFiClient client = server.available();

Maintenant, si la demande du client est disponible, ces instructions commenceront à recevoir des données du client et stockeront les données dans la chaîne d'en-tête et continueront à recevoir des données jusqu'à ce que '\ n' soit introuvable, ce qui signifie que le client s'est déconnecté.

if (client)

{

Serial.println("New Client is requesting web page");

String current\_data\_line = "";

while (client.connected()) {

if (client.available()) {

char new\_byte = client.read();

Serial.write(new\_byte);

header += new\_byte;

if (new\_byte == '\n') {

if (current\_data\_line.length() == 0)

{

client.println("HTTP/1.1 200 OK");

client.println("Content-type:text/html");

client.println("Connection: close");

client.println();

**Commande des LED Code**

Maintenant, sur la base des données reçues du client que nous avons stockées dans une chaîne "en-tête", nous allons activer et désactiver la LED respective. Ces conditions vérifieront si le bouton est enfoncé ou non. Si le bouton LED0 est enfoncé, comme nous l'avons vu précédemment, nous recevrons un message '?LED0=0N' du client et nous enregistrerons ce message dans la chaîne d'en-tête. Ces lignes utilisent la fonction header.indexof () qui vérifie si une chaîne spécifique est disponible dans l'en-tête ou non, si disponible, elle s'allumera et s'éteindra la broche numérique respective et changera également l'état de la LED qui sera mise à jour sur le serveur Web. Nous avons trois boutons et chaque bouton a deux états. Par conséquent, nous avons ici six conditions pour vérifier ce qui est reçu dans la chaîne d'en-tête.

if (header.indexOf("LED0=ON") != -1)

{

Serial.println("GPIO23 LED is ON");

LED\_ONE\_STATE = "on";

digitalWrite(GPIO\_PIN\_NUMBER\_22, HIGH);

}

if (header.indexOf("LED0=OFF") != -1)

{

Serial.println("GPIO23 LED is OFF");

LED\_ONE\_STATE = "off";

digitalWrite(GPIO\_PIN\_NUMBER\_22, LOW);

}

if (header.indexOf("LED1=ON") != -1)

{

Serial.println("GPIO23 LED is ON");

LED\_TWO\_STATE = "on";

digitalWrite(GPIO\_PIN\_NUMBER\_23, HIGH);

}

if (header.indexOf("LED1=OFF") != -1)

{

Serial.println("GPIO23 LED is OFF");

LED\_TWO\_STATE = "off";

digitalWrite(GPIO\_PIN\_NUMBER\_23, LOW);

}

if (header.indexOf("LED2=ON") != -1)

{

Serial.println("GPIO15 LED is ON");

LED\_THREE\_STATE = "on";

digitalWrite(GPIO\_PIN\_NUMBER\_15, HIGH);

}

if(header.indexOf("LED2=OFF") != -1) {

Serial.println("GPIO15 LED is OFF");

LED\_THREE\_STATE = "off";

digitalWrite(GPIO\_PIN\_NUMBER\_15, LOW);

}

**Affichage du HTML sur le serveur Web**

La prochaine partie du programme consiste à afficher le code HTML et CSS sur le navigateur Web. Cette fonction client.println () est utilisée pour envoyer des commandes HTML et CSS au client qui accède à un serveur Web via une adresse IP. Ce fichier HTML que nous voulons envoyer à un client. Vous pouvez trouver plus de détails sur HTML et CSS sur ce [lien](https://www.w3schools.com/) .

<!DOCTYPE html>

<html>

<head>

<style>

.button {

background-color: #4CAF50;

border: 2px solid #4CAF50;;

color: white;

padding: 15px 32px;

text-align: center;

text-decoration: none;

display: inline-block;

font-size: 16px;

margin: 4px 2px;

cursor: pointer;

}

</style>

</head>

<body>

<center><h1 style="color:blue;">ESP32 Web server LED controlling example</h1></center>

<center><h2 style="color:black;">Web Server Example Microcontrollerslab.com</h2></center>

<center><h2 style="color:Green;">Press "ON" button to turn on led and "OFF" button to turn off LED</h3></center>

<form>

<center>

<button class="button" name="LED0" value="ON" type="submit">LED0 ON</button>

<button class="button" name="LED0" value="OFF" type="submit">LED0 OFF</button><br><br>

<button class="button" name="LED1" value="ON" type="submit">LED1 ON</button>

<button class="button" name="LED1" value="OFF" type="submit">LED1 OFF</button> <br><br>

<button class="button" name="LED2" value="ON" type="submit">LED2 ON</button>

<button class="button" name="LED2" value="OFF" type="submit">LED2 OFF</button>

</center>

</form>

</body>

</html>

* Nous servirons ce fichier HTML et CSS à un client Web via ESP32. Chaque fois qu'un client Web essaie d'accéder à la page Web via une adresse IP, nous envoyons cette page HTML.
* Les boutons sont utilisés pour contrôler les LED. Chaque fois qu'un client Web appuie sur le bouton, nous recevons une requête HTTP. Sur la base de la requête HTTP, nous prenons des mesures et contrôlons les LED.

**Affichage de la page Web avec ESP32**

Mais nous pouvons l'envoyer via la fonction client.println() uniquement. Voyons maintenant comment configurer une page Web via ESP32. Cette ligne montrera que nous voulons envoyer le fichier HTML.

client.println("<!DOCTYPE html><html>");

Cette ligne est utilisée parce que nous voulons rendre notre page Web accessible sur tous les appareils comme un ordinateur, un ordinateur portable, un mobile et une tablette et cela rendra la page Web réactive.

client.println("<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1\">");

Ces lignes sont utilisées pour donner du style et des couleurs aux textes et aux boutons utilisés sur la page Web. Ils améliorent l'apparence de la page Web. Pour cette ligne, nous utilisons la famille de polices Helvetica et définissons la couleur d'une page Web par défaut que nous pouvons modifier ultérieurement et également définir le texte au centre de la page.

client.println("<style>html { font-family: Helvetica; display: inline-block; margin: 0px auto; text-align: center;}");

Pour donner du style aux boutons et aussi de la couleur, cette ligne définira la couleur de fond en vert et la couleur de la bordure en noir et la couleur du texte en blanc. Il définit également le rembourrage, l'alignement du texte et la taille de la police du texte utilisé à l'intérieur des boutons.

client.println(".button { background-color: #4CAF50; border: 2px solid #4CAF50;; color: white; padding: 15px 32px; text-align: center; text-decoration: none; display: inline-block; font-size: 16px; margin: 4px 2px; cursor: pointer; }");

client.println("text-decoration: none; font-size: 30px; margin: 2px; cursor: pointer;}");

Ces instructions de code impriment les parties d'en-tête du code et affichent un message texte sur le serveur.

client.println("</style></head>");

client.println("<body><center><h1>ESP32 Web server LED controlling example</h1></center>");

client.println("<center><h2>Web Server Example Microcontrollerslab.com</h2></center>" );

client.println("<center><h2>Press on button to turn on led and off button to turn off LED</h3></center>");

client.println("<form><center>");

**Code HTML pour afficher l'état des boutons et des voyants**

Cette partie est utilisée pour afficher les boutons ainsi que l'état des voyants sur une page Web.

client.println("<p> LED one is " + LED\_ONE\_STATE + "</p>");

// If the PIN\_NUMBER\_22State is off, it displays the ON button

client.println("<center> <button class=\"button\" name=\"LED0\" value=\"ON\" type=\"submit\">LED0 ON</button>") ;

client.println("<button class=\"button\" name=\"LED0\" value=\"OFF\" type=\"submit\">LED0 OFF</button><br><br>");

client.println("<p>LED two is " + LED\_TWO\_STATE + "</p>");

client.println("<button class=\"button\" name=\"LED1\" value=\"ON\" type=\"submit\">LED1 ON</button>");

client.println("<button class=\"button\" name=\"LED1\" value=\"OFF\" type=\"submit\">LED1 OFF</button> <br><br>");

client.println("<p>LED three is " + LED\_THREE\_STATE + "</p>");

client.println ("<button class=\"button\" name=\"LED2\" value=\"ON\" type=\"submit\">LED2 ON</button>");

client.println ("<button class=\"button\" name=\"LED2\" value=\"OFF\" type=\"submit\">LED2 OFF</button></center>");

client.println("</center></form></body></html>");

client.println();

Il s'agit donc de ce premier tutoriel sur la création d'un serveur Web à l'aide d'ESP32 dans l'IDE Arduino.

**Fonctions Wi-Fi Arduino**

Les détails des fonctions utilisées dans ce code sont donnés ci-dessous :

* WifiSever() : cette procédure permet de créer un serveur utilisé pour écouter les requêtes HTTP entrantes du client Web.
* WiFi.begin(ssid, password) : cette fonction initialise la bibliothèque WiFi avec deux arguments en entrée de cette fonction. Le premier argument est le nom du réseau et le deuxième argument est le mot de passe du réseau sans fil.
* WiFi.connected() : cette fonction vérifie si ESP32 est connecté au réseau ou non. Il renverra le statut TRUE ou FALSE selon la connexion.
* WiFilocalIP() : Si ESP32 est connecté avec succès à un réseau, nous pouvons utiliser cette fonction pour obtenir l'adresse IP qui est attribuée à la carte.
* server.begin() : il vous dira de démarrer la connexion au client Web et de commencer à recevoir les demandes des clients Web.

**Le code entier :**

#include <WiFi.h>

const char\* WIFI\_NAME= "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

const char\* WIFI\_PASSWORD = "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*";

WiFiServer server(80);

String header;

String LED\_ONE\_STATE = "off";

String LED\_TWO\_STATE = "off";

String LED\_THREE\_STATE = "off";

const int GPIO\_PIN\_NUMBER\_22 = 22;

const int GPIO\_PIN\_NUMBER\_23 = 23;

const int GPIO\_PIN\_NUMBER\_15 = 15;

void setup() {

Serial.begin(115200);

pinMode(GPIO\_PIN\_NUMBER\_22, OUTPUT);

pinMode(GPIO\_PIN\_NUMBER\_23, OUTPUT);

pinMode(GPIO\_PIN\_NUMBER\_15, OUTPUT);

digitalWrite(GPIO\_PIN\_NUMBER\_22, LOW);

digitalWrite(GPIO\_PIN\_NUMBER\_23, LOW);

digitalWrite(GPIO\_PIN\_NUMBER\_15, LOW);

Serial.print("Connecting to ");

Serial.println(WIFI\_NAME);

WiFi.begin(WIFI\_NAME, WIFI\_PASSWORD);

while (WiFi.status() != WL\_CONNECTED) {

delay(1000);

Serial.print("Trying to connect to Wifi Network");

}

Serial.println("");

Serial.println("Successfully connected to WiFi network");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

server.begin();

}

void loop(){

WiFiClient client = server.available();

if (client) {

Serial.println("New Client is requesting web page");

String current\_data\_line = "";

while (client.connected()) {

if (client.available()) {

char new\_byte = client.read();

Serial.write(new\_byte);

header += new\_byte;

if (new\_byte == '\n') {

if (current\_data\_line.length() == 0)

{

client.println("HTTP/1.1 200 OK");

client.println("Content-type:text/html");

client.println("Connection: close");

client.println();

if (header.indexOf("LED0=ON") != -1)

{

Serial.println("GPIO23 LED is ON");

LED\_ONE\_STATE = "on";

digitalWrite(GPIO\_PIN\_NUMBER\_22, HIGH);

}

if (header.indexOf("LED0=OFF") != -1)

{

Serial.println("GPIO23 LED is OFF");

LED\_ONE\_STATE = "off";

digitalWrite(GPIO\_PIN\_NUMBER\_22, LOW);

}

if (header.indexOf("LED1=ON") != -1)

{

Serial.println("GPIO23 LED is ON");

LED\_TWO\_STATE = "on";

digitalWrite(GPIO\_PIN\_NUMBER\_23, HIGH);

}

if (header.indexOf("LED1=OFF") != -1)

{

Serial.println("GPIO23 LED is OFF");

LED\_TWO\_STATE = "off";

digitalWrite(GPIO\_PIN\_NUMBER\_23, LOW);

}

if (header.indexOf("LED2=ON") != -1)

{

Serial.println("GPIO15 LED is ON");

LED\_THREE\_STATE = "on";

digitalWrite(GPIO\_PIN\_NUMBER\_15, HIGH);

}

if(header.indexOf("LED2=OFF") != -1) {

Serial.println("GPIO15 LED is OFF");

LED\_THREE\_STATE = "off";

digitalWrite(GPIO\_PIN\_NUMBER\_15, LOW);

}

client.println("<!DOCTYPE html><html>");

client.println("<head><meta name=\"viewport\" content=\"width=device-width, initial-scale=1\">");

client.println("<link rel=\"icon\" href=\"data:,\">");

client.println("<style>html { font-family: Helvetica; display: inline-block; margin: 0px auto; text-align: center;}");

client.println(".button { background-color: #32cfbf; border: 2px solid #32cfbf;; color: black; padding: 15px 32px; text-align: center; text-decoration: none; display: inline-block; font-size: 16px; margin: 4px 2px; cursor: pointer; }");

client.println("text-decoration: none; font-size: 30px; margin: 2px; cursor: pointer;}");

// Web Page Heading

client.println("</style></head>");

client.println("<body><center><h1>ESP32 Controle de LED </h1></center>");;

client.println("<form><center>");

client.println("<p> LED 1 est " + LED\_ONE\_STATE + " <p> GPIO 22" + "</p>" );

// If the PIN\_NUMBER\_22State is off, it displays the ON button

client.println("<center> <button class=\"button\" name=\"LED0\" value=\"ON\" type=\"submit\">LED0 ON</button>") ;

client.println("<button class=\"button\" name=\"LED0\" value=\"OFF\" type=\"submit\">LED0 OFF</button><br><br>");

client.println("<p>LED 2 est " + LED\_TWO\_STATE + "<p> GPIO 23" + "</p>");

client.println("<button class=\"button\" name=\"LED1\" value=\"ON\" type=\"submit\">LED1 ON</button>");

client.println("<button class=\"button\" name=\"LED1\" value=\"OFF\" type=\"submit\">LED1 OFF</button> <br><br>");

client.println("<p>LED 3 est " + LED\_THREE\_STATE + "<p> GPIO 15" + "</p>");

client.println ("<button class=\"button\" name=\"LED2\" value=\"ON\" type=\"submit\">LED2 ON</button>");

client.println ("<button class=\"button\" name=\"LED2\" value=\"OFF\" type=\"submit\">LED2 OFF</button></center>");

client.println("</center></form></body></html>");

client.println();

break;

}

else

{

current\_data\_line = "";

}

}

else if (new\_byte != '\r')

{

current\_data\_line += new\_byte;

}

}

}

// Clear the header variable

header = "";

// Close the connection

client.stop();

Serial.println("Client disconnected.");

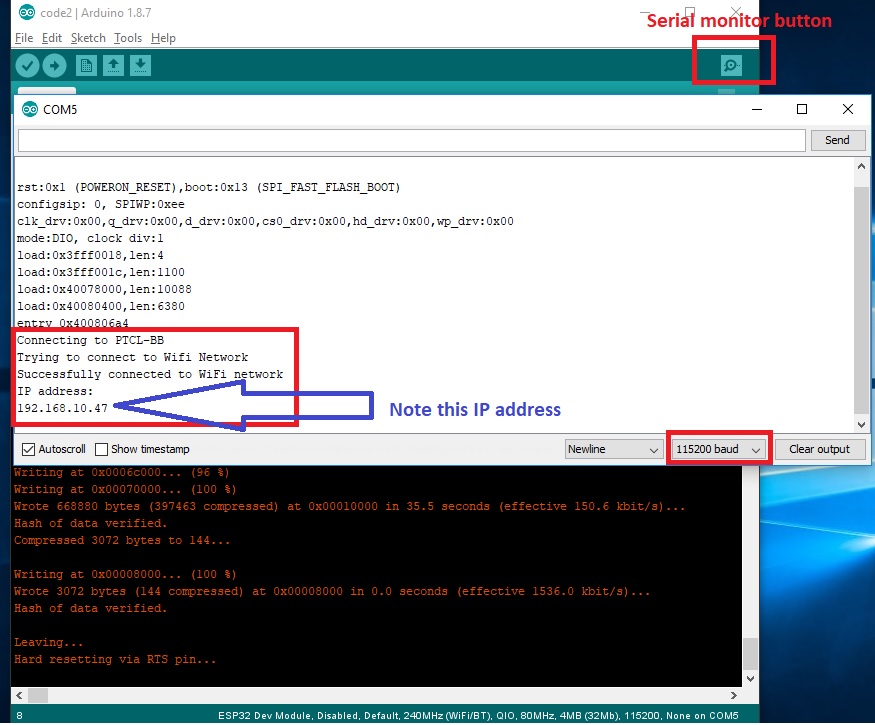
Serial.println("");

}

}

**Obtenir l'adresse IP**

Apres avoir téléversé votre code dans votre carte esp32**,** ouvrez le moniteur série et cliquez sur le bouton de réinitialisation de la carte du kit de développement ESP32. Vous verrez le message "essayer de se connecter à un réseau WiFi" et après un certain temps, vous verrez le message "connecté avec succès au réseau WiFi" ainsi que l'adresse IP comme indiqué dans la figure ci-dessous :



# TABLE DES MATIÈRES

[A la mémoire de : II](#_Toc124585265)

[Dédicace III](#_Toc124585266)

[Remerciements IV](#_Toc124585267)

[Avant-propos V](#_Toc124585268)

[SOMMAIRE VI](#_Toc124585269)

[Glossaire VII](#_Toc124585270)

[LISTE DES FIGURES IX](#_Toc124585271)

[LISTE DES TABLEAUX X](#_Toc124585272)

[Résumé XI](#_Toc124585273)

[Abstract XII](#_Toc124585274)

[Introduction générale 1](#_Toc124585275)

[Première Partie : Cadre théorique et méthodologique 3](#_Toc124585276)

[I. CADRE THEORIQUE 4](#_Toc124585277)

[I.1 Problématique 4](#_Toc124585278)

[I.2 Objectifs de recherche 5](#_Toc124585279)

[I.2.1 Objectif général 5](#_Toc124585280)

[I.2.2 Objectifs spécifiques 5](#_Toc124585281)

[I.3 Hypothèse de recherche 5](#_Toc124585282)

[II. CADRE METHODOLOGIQUE 6](#_Toc124585283)

[II.1 Cadre d’études 6](#_Toc124585284)

[II.2 Techniques d’investigation 6](#_Toc124585285)

[II.3 Difficultés rencontrées 6](#_Toc124585286)

[7](#_Toc124585287)

[Deuxième partie : le cadre conceptuel 7](#_Toc124585288)

[III. Généralités sur l’intelligence artificielle 8](#_Toc124585289)

[III.1 Intelligence artificielle 8](#_Toc124585290)

[III.2 Histoire de l’intelligence artificielle 9](#_Toc124585291)

[III.2.1 Domaine d’application de l’IA 11](#_Toc124585292)

[III.2.2 Différents types d’intelligence artificielle 11](#_Toc124585293)

[III.2.3 Sous-domaine de l’intelligence artificielle 11](#_Toc124585294)

[IV. Reconnaissance vocale 13](#_Toc124585295)

[IV.1 Présentation de la reconnaissance vocale 13](#_Toc124585296)

[IV.2 Domaine d’application 14](#_Toc124585297)

[IV.3 Fonctionnement la reconnaissance vocale 14](#_Toc124585298)

[IV.4 Fonctionnement du natural language understanding ? 15](#_Toc124585299)

[IV.5 Trois principales techniques de reconnaissance vocale ? 15](#_Toc124585300)

[V. Le Computer vison 16](#_Toc124585301)

[V.1 Définition 16](#_Toc124585302)

[V.2 Principe de fonctionnement du computer vison 16](#_Toc124585303)

[VI. Système intelligent (LANC) 17](#_Toc124585304)

[VI.1 Fonctionnement Le système 17](#_Toc124585305)

[VI.2 Avantages et inconvénients de LANC 20](#_Toc124585306)

[VI.3 Idée du projet 21](#_Toc124585307)

[VI.4 Domaines d’applications et Fonctionnalités 21](#_Toc124585308)

[VI.4.1 Fonctionnalités 21](#_Toc124585309)

[VI.4.2 Domaines d’applications : 22](#_Toc124585310)

[VI.5 Problèmes rencontrés 23](#_Toc124585311)

[Troisième partie : Mise en œuvre 24](#_Toc124585312)

[VII. Conception et réalisation 25](#_Toc124585313)

[VII.1 Choix des Matériaux 25](#_Toc124585314)

[VII.1.1 Carte ESP32 26](#_Toc124585315)

[VII.1.1.1 Présentation du serveur Web 28](#_Toc124585316)

[VII.1.1.2 Service Web et communication client 29](#_Toc124585317)

[VII.1.1.3 Comment un serveur Web répond-il à une requête client ? 29](#_Toc124585318)

[VII.1.1.4 Mode de fonctionnement de l’ESP32 29](#_Toc124585319)

[VII.1.1.5 Mode station ou STA 30](#_Toc124585320)

[VII.1.2 Module de Relais 2 canaux 30](#_Toc124585321)

[VII.1.2.1 Connexion de tension secteur 31](#_Toc124585322)

[VII.1.2.2 Actif bas 32](#_Toc124585323)

[VII.1.2.3 Actif élevé 32](#_Toc124585324)

[VII.1.2.4 Utilisation de l'actif bas 33](#_Toc124585325)

[VII.1.2.5 Point d'accès Wi-Fi 33](#_Toc124585326)

[VII.1.3 Ampoule électrique 33](#_Toc124585327)

[VII.1.4 Ordinateur portable (PC) 34](#_Toc124585328)

[VII.1.5 Connexion entre l’esp32 et la lampe 34](#_Toc124585329)

[VII.2 Plateforme de programmation Arduino 35](#_Toc124585330)

[VII.2.1 Langage Arduino 37](#_Toc124585331)

[VII.3 Le langage Programmation PYTHON 37](#_Toc124585332)

[VII.4 Tests et analyses 38](#_Toc124585333)

[VII.4.1 L’interface Graphique 38](#_Toc124585334)

[VII.4.2 Tests 42](#_Toc124585335)

[Conclusion générale 43](#_Toc124585336)

[Bibliographies 44](#_Toc124585337)

[Webographie 46](#_Toc124585338)

[Annexes 47](#_Toc124585339)

[TABLE DES MATIÈRES 66](#_Toc124585340)