

## Problématique

Beaucoup d'applications modernes auraient besoin de **voir** : un jeu qui réagit aux obstacles, une app d'accessibilité pour non-voyants, etc.

Or, installer un modèle de détection dans un service **fiable** en production reste un projet à part entière, hors du cœur de métier d'un développeur de jeu.

L'industrie n'a pas de service de vision **autonome** et prêt à l'emploi, conçu pour être branché rapidement à n'importe quelle application. Seules les APIs de vendeurs externes sont disponibles.

## Objectifs

Ce projet vise à développer un **système de vision intelligente** qui reçoit une image par une simple requête web et renvoie son identification, sous forme d'un service modulaire et réutilisable sans sacrifier l'**indépendance numérique** du projet.

## CodeProject.AI-Server

C'est un logiciel open source sur Github qui facilite l'utilisation de l'intelligence artificielle sur son propre serveur.

Il permet d'utiliser les possibilités de l'IA pour la détection, l'analyse, la modification de toutes sortes d'images et plus encore. Les avantages d'utiliser ce service seraient la confidentialité des données, la flexibilité de l'offre et l'autonomie des ressources informatiques.

## Matériel et Logiciel

Pour répéter l'expérience, vous aurez besoin des éléments suivants :

- **Serveur** : Ubuntu 24.04 LTS, 4 Go RAM
- **Sécurité** : passerelle Nginx + HTPasswd
- **IA** : CodeProject.AI 2.9.5 + YOLOv5 6.2

## Méthodologie

- 1) Infrastructure:**  
Déploiement d'un VPS linux sécurisé avec pare-feu et fail2ban. Mise en place d'un serveur web en reverse proxy authentifié pour exposer un point d'entrée unique sécurisé.
- 2) Intelligence artificielle :**  
Installation de serveur CodeProject.AI et sélection du module YOLOv5. Désactivation des autres modules. Optimisation d'inférence et validation de la pipeline avec des images de test utilisées dans une interface web.

## Mini jeu d'utilisation de la technologie

**3) Application réelle :**  
Afin de démontrer l'utilisation dans une application, développement d'un mini-jeu Unity qui capture le point de vue du personnage et lit la réponse JSON. Préparation du mode temps réel de flux vidéo continu.

## Résultats & Futur

Le service **industrialisé\*** permet d'obtenir pour n'importe quelle image ses informations signalétiques : objet identifié, position, confiance.

*\*Le service fournit un contrat d'API stable encapsulant des modèles versionnés. La latence est contrôlée et le système est résistant aux attaques.*

**Flux pour un client (exemple du mini-jeu) :**

1. Le client capture une image
2. Requête multipart 'POST /v1/vision/detection'
3. Nginx authentifie et transmet
4. YOLOv5 analyse et renvoie '{ label, confiance, box }'
5. Le client applique sa propre logique selon l'information

précision sensibilité 100%

Les améliorations futures sont de quatre ordres :

- Application aux caméras de surveillance
- Isolation automatique des images
- Automatisation du déploiement clé-en-main
- Portail libre-service d'inscription pour recevoir une clé d'API et visualiser un tableau de bord

