

2024-2025

Cahier de test unitaire

Canne Intelligente pour Aveugle

Encadrant : Gilles VENTURINI

Client : Gilles VENTURINI

Etudiant : Irfan AKTURK

1. Introduction

Dans ce document, je décris les tests unitaires réalisés pour valider le bon fonctionnement de chaque module de ma canne intelligente. L'objectif principal est de vérifier que chaque composant fonctionne correctement individuellement avant de les intégrer dans le système complet.

Les tests portent sur les modules suivants :

- Bouton : Détection des pressions effectuées par l'utilisateur.
- Caméra : Capture d'images et détection d'objets avec reconnaissance.
- Synthèse vocale : Lecture de messages pour guider l'utilisateur.
- Capteur ultrason : Mesure de la distance avec les obstacles.
- Vibreur : Génération de retours haptiques pour signaler les obstacles.

Ces tests m'ont permis de m'assurer que les fonctionnalités de chaque module répondent aux attentes définies dans la conception détaillée.

2. Organisation des Tests

Méthodologie :

Pour chaque module, j'ai écrit des tests qui vérifient les fonctions principales. J'ai noté les étapes pour reproduire chaque test, ainsi que le résultat attendu. Si le résultat correspond aux attentes, le test est validé.

Critères de validation :

- Le module doit fonctionner correctement, sans erreur, de manière autonome.
- Toutes les fonctionnalités principales doivent être testées.
- Les ressources utilisées (comme les GPIO, caméra, etc.) doivent être libérées correctement après l'exécution.




Outils utilisés :

- Jetson Nano pour exécuter le projet.
- Python 3.6 comme langage de programmation.
- Bibliothèques utilisées : Jetson.GPIO, jetson.inference, serial pour les capteurs et la caméra.
- Périphériques : bouton poussoir, caméra, capteur ultrason, et vibreur.

3. Cas de Tests




1. Module : Bouton

Objectif : Vérifier que les interactions avec le bouton sont correctement détectées.

ID	Test	Procédure	Résultat attendu	Statut
BTN_001	Détection d'un appui	<ul style="list-style-type: none">- Connecter le bouton sur le GPIO 11.- Exécuter le fichier bouton.py.- Appuyer sur le bouton.	Le message "Bouton appuyé !" s'affiche dans le terminal à chaque pression sur le bouton.	 Validé
BTN_002	Détection multiple	<ul style="list-style-type: none">- Maintenir le bouton enfoncé pendant plusieurs secondes.- Observer si plusieurs détections se produisent.	Une seule détection est enregistrée pour chaque pression.	 Validé
BTN_003	Nettoyage des GPIO	<ul style="list-style-type: none">- Arrêter le script avec Ctrl + C.	Les GPIO sont libérés proprement sans générer d'erreur.	 Validé



2. Module : Caméra

Objectif : Tester les fonctionnalités de capture d'images et de détection.

ID	Test	Procédure	Résultat attendu	Statut
CAM_001	Capture d'image	<ul style="list-style-type: none">- Connecter la caméra à la Jetson Nano.- Exécuter le fichier camera.py.	Aucune erreur de connexion ne doit être affichée.	 Validé
CAM_002	Détection d'objets	<ul style="list-style-type: none">- Positionner des objets devant la caméra.- Exécuter le fichier camera.py.- Observer les objets détectés.	Les objets détectés s'affichent dans le terminal avec leurs niveaux de confiance.	 Validé
CAM_003	Gestion des ressources	<ul style="list-style-type: none">- Lancer le script, puis l'arrêter avec Ctrl + C.	Les ressources liées à la caméra sont correctement libérées.	 Validé




3. Module : Synthèse Vocale

Objectif : Valider la conversion de texte en parole.

ID	Test	Procédure	Résultat attendu	Statut
SND_001	Annonce d'un message	- Exécuter le fichier sound.py. - Entrer un message à lire dans le script.	Le texte est lu correctement via le haut-parleur.	 Validé
SND_002	Gestion des erreurs	- Fournir un chemin incorrect pour le script text_to_speech.sh.	Un message d'erreur est affiché dans le terminal.	 Validé




4. Module : Capteur Ultrason

Objectif : Tester les mesures de distance et leur précision.

ID	Test	Procédure	Résultat attendu	Statut
USN_001	Lecture de la distance	- Connecter le capteur à la Jetson Nano. - Exécuter le fichier ultrasonic.py. - Positionner un objet à 50 cm.	La distance mesurée s'affiche dans le terminal avec une précision de +/- 5 %.	 Validé
USN_002	Obstacle hors portée	- Positionner un objet à moins de 3 cm ou au-delà de 5 m. - Observer le message affiché dans le terminal.	Un message "En-dessous de la limite inférieure" ou "Erreur lors de la mesure" est affiché.	 Validé
USN_003	Gestion des erreurs	- Débrancher le capteur, puis exécuter le script.	Un message d'erreur "Connexion série non disponible" est affiché.	 Validé

5. Module : Vibreur

Objectif : Valider le fonctionnement du vibreur et sa durée d'activation.

ID	Test	Procédure	Résultat attendu	Statut
VIB_001	Activation	<ul style="list-style-type: none">- Connecter le vibreur au GPIO 13.- Exécuter le fichier vibration.py.- Observer le comportement du vibreur pendant 0,5 seconde.	Le vibreur s'active pendant 0,5 seconde, puis s'arrête.	 Validé
VIB_002	Durée variable	<ul style="list-style-type: none">- Exécuter le fichier vibration.py.- Modifier la durée d'activation dans le code (par exemple 1 seconde).	Le vibreur s'active pendant la durée spécifiée.	 Validé
VIB_003	Nettoyage des GPIO	<ul style="list-style-type: none">- Arrêter le script avec Ctrl + C.	Les GPIO sont libérés proprement.	 Validé

4. Conclusion

Tous les tests unitaires ont été exécutés avec succès. Les modules sont prêts pour les tests d'intégration. Aucun problème bloquant n'a été identifié. Les prochaines étapes consistent à tester les interactions entre les modules pour valider l'ensemble du système.