

# Stratégie de tests

## VUE D'ENSEMBLE

L'ISTQB définit la stratégie de test de la manière suivante :

"La stratégie de test est le document de haut niveau définissant, pour un programme, les niveaux de tests à exécuter et les tests dans chacun de ces niveaux (pour un ou plusieurs projets)".

Il s'agit donc d'un document fixant les tests répartis par niveau dans la pyramide des tests, le RACI pour chacun des niveaux et les attendus en termes d'objectifs pour chacun des types de tests mis en œuvre.

---

<b>VUE D'ENSEMBLE</b>	<b>1</b>
<b>Contexte du projet</b>	<b>4</b>
<b>Environnement de test</b>	<b>5</b>
Liste des composants critiques	5
<b>Recensement et suivi macro des tests</b>	<b>6</b>
<b>Détail des tests</b>	<b>9</b>
CTRL-001	9
CTRL-002	9
CTRL-003	9
CTRL-004	10
CTRL-005	10
CTRL-006	10
CTRL-007	11
CTRL-008	11
DOLI-001	12
DOLI-002	12
DOLI-003	13
DOLI-004	13
ENGN-001	13
ENGN-002	14
ENGN-003	14
ENGN-004	15
RFID-001	15
SRVO-001	15
TITRE	16
<b>Hors périmètre</b>	<b>17</b>
Défaillances physiques	17
Autres	17
<b>Gestion des versions</b>	<b>18</b>
Follow-up 1	18
A présenter :	18
A valider :	18
Questions :	18
Remarques :	18

---

## Contexte du projet

Ce projet a pour vocation la mise en place d'un système d'automatisation au sein d'une infrastructure de transport de colis sur tapis roulant. Il est piloté par la société Teachers et sous-traité par la société Students.

Pour une remise en contexte historique, une autre société était chargée de livrer le programme faisant fonctionner ce système, elle s'est finalement retirée après accord des parties prenantes. Il n'y a donc à ce jour aucune trace de code, ni de documentation.

Aussi et pour finir, le système complet ne pourra être testé qu'à trois reprises, et en présence de notre client, durant des dates communes qui seront explicitées plus bas dans ce document.

---

# Environnement de test

## Liste des composants critiques

Cette partie sert à dresser l'ensemble des composants critiques qui devront être testés ainsi que la raison du pourquoi.

- **Contrôleur** : C'est la partie chargée de piloter l'ensemble des composants, il détient le circuit intégré et en fait donc un composant critique.
- **Moteur** : C'est le module servant à l'activation et au déplacement du tapis roulant. Son utilisation est largement prioritaire puisqu'il alimente le déroulé du système mis en place. En revanche, il n'a pas de communication avec les autres composants, le focus à mettre dessus se limitera au bon fonctionnement mécanique.
- **Lecteur RFID** : C'est le lecteur chargé de la lecture des informations des colis. Il est le premier élément à avoir une responsabilité sur le tapis roulant puisqu'il doit ensuite communiquer avec l'API Dolibarr et donner les directives au composant suivant.
- **Servomoteur** : C'est la partie chargée de définir l'entrepôt dans lequel le colis se trouvera en bout de course.
- **Dolibarr** : C'est l'application web qui sert d'ERP/CRM. Elle a à charge de gérer les stocks, de détenir la définition des produits qui sont acheminés et de prévenir le système lorsqu'un entrepôt arrive possiblement à son maximum de stock.
- **Capteur présence de colis** : C'est le dernier module chargé de capter le passage d'un colis après l'articulation du bras de direction afin de s'assurer que le paquet est bien en route vers un des trois entrepôts et non coincé.

## Recensement et suivi macro des tests

Le tableau suivant indique les échelles de criticité et les plans d'actions à prévoir en fonction.

Criticité	Plan d'action
Très faible (1 à 3)	Pas de correction immédiate nécessaire.
Faible (4 à 6)	Documenter et planifier une correction à long terme. Surveiller si le problème devient plus fréquent.
Modéré (7 à 9)	Planifier une correction rapide. Tester régulièrement pour éviter des aggravations.
Élevé (10 à 12)	Corrigez-les dès que possible. A tester fréquemment.
Critique (13 à 16)	Correction immédiate prioritaire. Mettre en pause d'autres activités si nécessaire pour résoudre ce problème.

Dans le tableau majeur de cette partie nous trouverons trois colonnes nommées "Gravité", "Fréquence" et "Criticité".

Les deux premières colonnes ont un indice allant de 1 à 4 en fonction de l'impact. La dernière qui est Criticité est calculée par multiplication entre les deux premières. Si nous avons une fonction avec une gravité de 2 et une fréquence de 3 alors sa criticité sera de 6.

Composant(s)	Fonctions	ID	Gravité	Fréquence	Criticité	Statut
API Dolibarr	Requête d'envoi de tag produit	<a href="#">DOLI-002</a>	4	4	16	C2
API Dolibarr	Requête de mouvement de stock	<a href="#">DOLI-004</a>	4	4	16	C3
Contrôleur	En mode production : se connecter à un point d'accès WI-FI externe	<a href="#">CTRL-001</a>	4	3	12	C2
Moteur	Contrôler l'alimentation	<a href="#">ENGN-002</a>	4	3	12	Décommissionné

Composant(s)	Fonctions	ID	Gravité	Fréquence	Criticité	Statut
	du module					
<b>Moteur</b>	Mettre le moteur en marche avant.	<a href="#">ENGN-004</a>	4	3	12	C1
<b>Capteur présence colis</b>	Vérifier le passage d'un colis	SNSR-001	3	4	12	Non décrit
<b>API Dolibarr</b>	Détection lors d'un entrepôt plein	<a href="#">DOLI-001</a>	3	3	9	Décommissionné
<b>Moteur</b>	Contrôler la puissance que reçoit le module	<a href="#">ENGN-001</a>	3	3	9	Décommissionné
<b>Contrôleur</b>	Persistance des logs error sur des fichiers internes à l'ESP.	<a href="#">CTRL-004</a>	2	4	8	Décommissionné
<b>Contrôleur</b>	Détection du mode d'alimentation (secteur/batterie).	<a href="#">CTRL-006</a>	4	2	8	Décommissionné
<b>API Dolibarr</b>	Authentification	<a href="#">DOLI-003</a>	4	2	8	C2
<b>Moteur</b>	Mettre le moteur à l'arrêt.	<a href="#">ENGN-003</a>	4	2	8	C2
<b>Lecteur RFID</b>	Scan de tags en mouvement.	<a href="#">RFID-001</a>	4	2	8	C2
<b>Servomotor</b>	Bouger le moteur selon un angle	<a href="#">SRVO-001</a>	4	2	8	C2
<b>Contrôleur</b>	Navigation entre les	<a href="#">CTRL-008</a>	3	2	6	

Composant(s)	Fonctions	ID	Gravité	Fréquence	Criticité	Statut
	différents écrans.					
<b>Contrôleur</b>	Récupération des fichiers.	<a href="#">CTRL-005</a>	2	2	4	Décommissionné
<b>Contrôleur</b>	En mode configuration : exposer un point d'accès WI-FI	<a href="#">CTRL-002</a>	2	1	3	C2
<b>Contrôleur</b>	Validation d'une alerte sonore.	<a href="#">CTRL-007</a>	3	1	3	Décommissionné
<b>Contrôleur</b>	Switch du mode production à configuration	<a href="#">CTRL-003</a>	2	1	2	Décommissionné
<b>Capteur présence colis</b>	La capteur est en capacité de détecter un colis dans la demi-seconde.	SNSR-002				Non décrit

---

# Détail des tests

## CTRL-001

**Composant testé** : Contrôleur

**Objectif** : Le contrôleur doit-être capable de se connecter à un point d'accès WI-FI externe quand sa configuration est en mode production.

**Préconditions** : Mettre le contrôleur en mode production et avoir un point d'accès WI-FI externe.

**Étapes** :

1. TODO

**Résultat attendu** : Le contrôleur s'est correctement connecté au point d'accès WI-FI mis à disposition.

**Ticket** : <https://github.com/MelvinCou/The-translation/issues/14>

**Statut** : COFIL 2, OK.

## CTRL-002

**Composant testé** : Contrôleur

**Objectif** : Le contrôleur doit-être capable d'exposer un point d'accès WI-FI quand sa configuration est en mode configuration.

**Préconditions** : Que le contrôleur soit en mode configuration

**Étapes** :

1. Configurer le contrôleur en mode configuration,
2. Exposer l'accès WI-FI depuis le contrôleur,
3. Se connecter à cet accès WI-FI depuis l'API Dolibarr.

**Résultat attendu** : L'API Dolibarr est en capacité d'atteindre le point d'accès WI-FI exposé par le contrôleur.

**Ticket** : <https://github.com/MelvinCou/The-translation/issues/12>

**Statut** : COFIL 2, OK.

## CTRL-003

**Composant testé** : Contrôleur.

**Objectif** : Le contrôleur doit switcher du mode production au mode configuration et au mode maintenance, et vice-versa.

**Préconditions** : Avoir à disposition les trois modes de disponible et l'interface du contrôleur permet ce switch grâce à ses boutons de navigation.



---

**Étapes :**

1. TODO

**Résultat attendu :** Le contrôleur switch du mode production au mode configuration, et vice-versa et sans encombre ni problème de configuration.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/11>

**Statut :** Décomissionné, semblable au [CTRL-008](#).

## **CTRL-004**

**Composant testé :** Contrôleur.

**Objectif :** Persister les logs d'erreurs potentiels sur des fichiers internes au contrôleur.

**Préconditions :** Avoir de la place disponible dans le contrôleur et savoir simuler une erreur.

**Étapes :**

1. Déclencher une erreur,
2. Lire l'erreur depuis l'interface du contrôleur.

**Résultat attendu :** L'erreur est bien visible et lue depuis l'interface du contrôleur prévue à cet effet.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/16>

**Statut :** Décomissionné, non prioritaire.

## **CTRL-005**

**Composant testé :** Contrôleur

**Objectif :** Récupérer les fichiers de logs et pouvoir les manipuler en lecture.

**Préconditions :** Avoir des fichiers de logs existants au sein du contrôleur et programmer la navigation des boutons du contrôleur avec pour accès l'interface de lecture des logs.

**Étapes :**

1. Accéder depuis les boutons de navigation à l'interface des logs,
2. Lire les logs,
3. Sortir de l'interface de lecture des logs.

**Résultat attendu :**

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/16>

**Statut :** Décomissionné, non prioritaire.

## **CTRL-006**

**Composant testé :** Contrôleur

---

**Objectif** : Détecter le mode d'alimentation du contrôleur en vue d'alerter un état critique de l'alimentation (coupure de courant).

**Préconditions** : Avoir branché le contrôleur sur secteur.

**Étapes** :

1. Détecter le mode d'alimentation du contrôleur sur secteur,
2. Débrancher la source d'alimentation,
3. Détecter le mode d'alimentation du contrôleur sur batterie,
4. Rebrancher la source d'alimentation,
5. Détecter le mode d'alimentation du contrôleur sur secteur.

**Résultat attendu** : Le système est capable de savoir dans quel mode d'alimentation le contrôleur se trouve.

**Ticket** : <https://github.com/MelvinCou/The-translation/issues/17>

**Statut** : Décomissionné, pas de possibilité d'alimenter via batterie et donc qu'une seule alimentation est possible (secteur).

## CTRL-007

**Composant testé** : Contrôleur

**Objectif** : Alerter les utilisateurs à l'aide d'une alarme sonore émise par le contrôleur.

**Préconditions** : Avoir conçu un état critique à lancer à la demande pour simuler le déclenchement de l'alerte.

**Étapes** :

1. Déclencher l'état critique,
2. L'alarme se met à sonner.

**Résultat attendu** : L'alarme sonore se déclenche et est assez audible pour prévenir les utilisateurs d'un état critique.

**Ticket** : <https://github.com/MelvinCou/The-translation/issues/17>

**Statut** : Décomissionné, non prioritaire.

## CTRL-008

**Composant testé** : Contrôleur

**Objectif** : Pouvoir naviguer entre les différentes interfaces du contrôleur, à savoir, Configurations, Production et Maintenance.

**Préconditions** : Avoir conçu les interfaces du contrôleur ainsi que le menu et programmer les boutons de navigations.

**Étapes** :

1. Démarrer le test en étant sur le menu,
2. Entrer dans la première interface (Configurations),
3. Retourner sur le menu et entrer dans la deuxième interface (Production),

- 
4. Retourner sur le menu et entrer dans la troisième interface (Maintenance),
  5. Retourner sur le menu.

**Résultat attendu :** L'utilisateur est en capacité de pouvoir naviguer entre les différentes interfaces à travers le menu général. Cela implique qu'aucun bug ne doit se présenter sur la notion d'entrer dans une interface et d'en ressortir.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/15>

**Statut :** COPIL 2, OK.

## DOLI-001

**Composant testé :** API Dolibarr

**Objectif :** Détecter lorsqu'un entrepôt arrive à son maximum de capacité produit afin de prévenir le contrôleur de cette information.

**Préconditions :** Avoir mis en place la communication entre l'API Dolibarr et le contrôleur et avoir préalablement ajusté le stock produit à une unité en dessous de la capacité maximum de l'entrepôt.

**Étapes :**

1. Scanner un produit,
2. L'instance Dolibarr détecte que son entrepôt arrive à son maximum de capacité,
3. L'API Dolibarr communique au contrôleur l'état de son entrepôt,
4. Le contrôleur reçoit l'information.

**Résultat attendu :** L'instance Dolibarr est en capacité de détecter lorsqu'un des entrepôts arrive à son maximum en capacité produit et alerte le contrôleur de cette surcharge.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/18>

**Statut :** Décommissionné, pas d'API permettant l'envoi de cette information.

## DOLI-002

**Composant testé :** API Dolibarr

**Objectif :** Transmettre au contrôleur les informations relatives au produit scanné par le lecteur RFID

**Préconditions :** Avoir mis en place la transmission des informations depuis le lecteur RFID jusqu'au contrôleur et une communication entre le contrôleur et l'API Dolibarr. Avoir configuré le produit dans l'instance Dolibarr et une communication entre ce dernier et le contrôleur.

**Étapes :**

1. Scanner un produit avec le lecteur RFID,
2. Le contrôleur reçoit les informations du produit et les envoie à l'instance Dolibarr,
3. Dolibarr reconnaît le produit et transmet au contrôleur le numéro de l'entrepôt dans lequel le système doit rediriger le colis.

---

**Résultat attendu** : Dolibarr envoie au contrôleur le bon numéro de l'entrepôt dans lequel le système doit rediriger le colis scanné.

**Ticket** : <https://github.com/MelvinCou/The-translation/issues/18>

**Statut** : COPIL 2, OK.

### DOLI-003

**Composant testé** : API Dolibarr

**Objectif** : Authentification

**Préconditions** : L'API à l'authentification est accessible, le mode configuration du contrôleur est conçu et celui-ci permet d'exposer un serveur web. Une interface est préalablement conçue afin de pouvoir rentrer la clé API.

**Étapes** :

1. Le contrôleur est en mode configuration,
2. Le contrôleur expose un serveur web,
3. Une machine se connecte à ce serveur web,
4. L'interface disponible permet de rentrer la clé API de Dolibarr,
5. La réponse de l'authentification est reçue.

**Résultat attendu** : L'authentification à Dolibarr est réussie et permet au contrôleur de communiquer avec ce dernier.

**Ticket** : <https://github.com/MelvinCou/The-translation/issues/18>

**Statut** : COPIL 2, OK.

### DOLI-004

**Composant testé** : API Dolibarr

**Objectif** : Transmettre à l'instance Dolibarr qu'un colis a été scanné pour que celui-ci renvoie vers l'entrepôt souhaité et effectue un mouvement de stock en son sein.

**Préconditions** : Lecture de tag RFID,

**Étapes** :

1. Le lecteur RFID scan un colis,
2. Le contrôleur envoie ce tag à l'instance Dolibarr,
3. Dolibarr cible l'entrepôt pour le colis et envoie cette information au contrôleur,
4. Le contrôleur redirige l'axe du servomoteur vers l'entrepôt souhaité,
5. Le contrôleur communique à l'instance Dolibarr l'entrepôt vers lequel le colis a été redirigé,
6. L'instance Dolibarr incrémente son nombre de produits en son sein dans l'entrepôt ciblé.

**Résultat attendu** : L'instance Dolibarr incrémente son nombre de produits en son sein dans l'entrepôt ciblé.

**Ticket** : <https://github.com/MelvinCou/The-translation/issues/18>

**Statut** : COPIL 3, OK.

---

## ENGN-001

**Composant testé :** Moteur

**Objectif :** Contrôler la puissance que reçoit le module.

**Préconditions :** Avoir conçu un programme permettant au contrôleur de communiquer au moteur la puissance à laquelle il doit tourner.

**Étapes :**

1. Le contrôleur transmet au moteur une puissance inférieure à celle à laquelle il tourne en situation normal,
2. Le contrôleur transmet au moteur la puissance à laquelle il tourne en situation normal,
3. Le contrôleur transmet au moteur une puissance supérieure à celle à laquelle il tourne en situation normale.

**Résultat attendu :** Le contrôleur est en capacité de pouvoir modifier la puissance que reçoit le moteur.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/20>

**Statut :** Décomissionné, le module ne permet pas cette action.

## ENGN-002

**Composant testé :** Moteur

**Objectif :** Connaître l'état d'alimentation du module.

**Préconditions :** Avoir un listener permettant au contrôleur de connaître l'état de l'alimentation du moteur et sa puissance en temps réel.

**Étapes :**

1. Le moteur est branché et tourne,
2. Le moteur est débranché,
3. Le contrôleur détecte et constate l'état de l'alimentation défectueuse,
4. Le moteur est rebranché,
5. Le contrôleur détecte et constate l'état de l'alimentation en fonctionnement normal.

**Résultat attendu :** Le contrôleur est en capacité de reconnaître l'état de l'alimentation du moteur.

**Ticket :**

**Statut :** Décomissionné, le module ne permet pas cette action.

## ENGN-003

**Composant testé :** Moteur

**Objectif :** Mettre le moteur à l'arrêt.

**Préconditions :** Avoir conçu un programme permettant au contrôleur de mettre le moteur à l'arrêt de fonctionnement.

---

**Étapes :**

1. Le moteur tourne,
2. Le contrôleur indique au moteur de s'arrêter,
3. Le moteur s'arrête.

**Résultat attendu :** Le moteur s'arrête après que le contrôleur lui ait indiqué.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/20>

**Statut :** COPIL 2, OK.

## ENGN-004

**Composant testé :** Moteur

**Objectif :** Mettre le moteur en fonctionnement de marche avant.

**Préconditions :** Avoir conçu un programme permettant au contrôleur de mettre le moteur en fonctionnement de marche avant.

**Étapes :**

1. Le moteur est à l'arrêt,
2. Le contrôleur indique au moteur de tourner,
3. Le moteur tourne en marche avant.

**Résultat attendu :** Le moteur s'active et tourne en marche avant après que le contrôleur lui ait indiqué.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/20>

**Statut :** COPIL 1, OK.

## RFID-001

**Composant testé :** Lecteur RFID

**Objectif :** Vérifier la réponse du système lors d'une lecture RFID.

**Préconditions :** Colis avec un tag RFID et lecteur RFID allumé.

**Étapes :**

1. Poser un colis sur le tapis.
2. Attendre la lecture RFID.
3. Lecture et vérification des données transmises.

**Résultat attendu :** Le système reçoit les informations relatives à la lecture du tag RFID et sont conformes.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/19>

**Statut :** COPIL 2, OK.

## SRVO-001

---

**Composant testé :** Servomotor

**Objectif :** Bouger le moteur selon un angle souhaité.

**Préconditions :** Avoir un programme permettant la transmission de directives entre le contrôleur et le servomotor. Avoir défini les différentes positions possibles du servomotor.

**Étapes :**

1. Le contrôleur indique au servomotor de s'axer dans un angle souhaité,
2. Répéter l'étape 1 à partir de toutes les positions définies et pour tous les angles souhaités et définis.

**Résultat attendu :** Le servomotor est en capacité de pouvoir s'axer dans tous les angles possibles et définis et ce depuis toutes les positions.

**Ticket :** <https://github.com/MelvinCou/The-translation/issues/21>

**Statut :** COPIL 2, OK.

## TITRE

**Composant testé :**

**Objectif :**

**Préconditions :**

**Étapes :**

- 1.

**Résultat attendu :**

**Ticket :**

**Statut :**

---

## Hors périmètre

Cette partie dresse l'ensemble des tests qui ne seront pas pris en charge dans ce document en expliquant la raison.

Il reste toutefois essentiel de s'assurer que le périmètre est bien défini avec le client afin que chaque parties prenantes prennent les responsabilités adéquates.

## Défaillances physiques

**Le tapis roulant** : son état ne pourra être vérifié puisqu'aucun capteur n'a été pensé pour. Si le tapis présente un défaut et entraîne une défaillance du système (déchirement si accroche avec d'autres modules, entraîne un colis hors chemin prévu pour), le programme ne pourra pas en être tenu pour rigueur.

## Autres

**Taille des colis** : la vérification de la taille des colis n'est pas prise en charge. Surtout dans le cas où l'un d'eux serait plus grand que ce que le système peut accepter. En revanche, les passages au niveau du lecteur RFID et de l'axe de positionnement des bras présentent une taille minimale, si un colis venait à être coincé, une alerte sera générée pour indiquer qu'aucune activité n'a été recensée depuis un temps fixé.

**Colis collés** : si deux colis sont collés lorsqu'ils sont scannés par le lecteur RFID, celui-ci n'est pas capable de les différencier et les colis seront considérés comme un seul. Il faut donc que les colis soient au préalable espacés d'au moins 5 centimètres.



---

# Gestion des versions

Cette partie sert à dresser l'état d'avancement du produit et de sa qualité à travers la couverture des tests et de leur validation auprès du client, lors des réunions prévues à cet effet.

Elles seront au nombre de trois.

## Follow-up 1

A date du 10/10/2024

A présenter :

- Le [Wiki](#),
- Le document,
- La gestion des issues
- Batterie de tests

A valider :

- priorité des issues, et la ci,
- La priorité des tests,
- La partie hors périmètre afin de s'assurer de ce que nous devons prendre en compte,
- 

Questions :

- Faut-il ajouter l'indice de non détection dans le facteur de criticité d'une fonction/test ?
- La présence d'un autre sensor, présentation du [ticket #22](#)

Remarques :

- ISO 9001