



GNG1503 - Génie de la conception
Projet de conception

Livrable G
Prototype II et rétroaction de clients

Soumis par:

Succar, Mavie (300059711)

Bolohan, Ian (300310639)

Gbotta, Josée Danielle(300391871)

Fahim, Aya (300326408)

Boudabbous, Mohamed (300376202)

Chargé du cours: Emmanuel Bouendeu

Automne 2023
Université d'Ottawa

Livrable G - FA13

Table des matières :

1-Introduction.....	3
2- Compte-rendu de la rétroaction client.....	3
3- Développement du deuxième prototype.....	3-4
4- Modèle analytique, numérique ou expérimental.....	4-5-6
5- Documentation, analyse et résultat du plan d'essai de prototypage.....	6-7
6- Rétroaction du prototype des clients/utilisateurs.....	7-8
7- Mise à jour des spécifications cibles, de la conception détaillée et de la NDM.....	8-9-10
8- Plan d'essai de prototypage du troisième prototype.....	10-11-12
9- Conclusion.....	12-13
10- Bibliographie	13

1-Introduction

Dans le cadre de notre projet de conception, nous sommes chargés de développer un système de suivi automatique d'inventaire permettant aux clients une gestion efficace des articles de l'inventaire. Dans le livrable précédent, nous avons créé un prototype pour les étagères intelligentes. Pour ce faire, nous avons établi un plan d'essais et de prototype documenté, mis à jour les spécifications cibles, la conception détaillée et la NDM précédemment déterminés et recueilli la rétroaction de potentiels utilisateurs. Enfin, nous avons élaboré un plan de prototype relatif au second prototype, présentement étudié. De ce fait, lors de ce livrable, nous allons concevoir notre deuxième prototype, traitant des capteurs, et itérer les mêmes étapes suivies au cours du livrable précédent, comme énuméré ci-dessus.

2- Compte-rendu de la rétroaction client

La rétroaction recueillie du client suite à notre dernière rencontre est majoritairement positive. En effet, ils étaient satisfait par le prototype présenté mais ont relevé que celui-ci est basé sur la supposition que le personnel de l'entrepôt replace correctement les articles à leur place. Toutefois, il est possible que, par inadvertance, une erreur soit commise. Par conséquent, suite à l'analyse de leur retour, nous avons décidé d'incorporer une méthode de vérification quant au rangement des articles, de sorte à minimiser les pertes dues au facteur humain. Celle-ci consiste donc à assigner à chaque étagère un poids moyen d'articles présents, si bien qu'en cas d'ajout supplémentaire engendrant une modification de ce poids, à une marge de $\pm 0,02\%$, une notification sera envoyée pour le signaler. Semblablement, toute fluctuation minime sera programmée de sorte à être ignorée par la base de données car insignifiante. De plus, la seconde solution mise en place sera de prendre en compte les informations de livraisons enregistrées afin d'obtenir le nombre d'articles exportés ou importés prévu, connaissant le poids de ces articles, il est possible de calculer le poids total que devrait supporter chaque étagère. De telle manière, en cas d'incohérence, une alerte sera envoyée aux gestionnaires, qui seront alors avertis et pourront, conséquemment, vérifier manuellement le rayon en question.

3- Développement du deuxième prototype

Notre second prototype est consacré au système de capteurs ultrasons. Connectés à un circuit NODE MCU programmé, les deux capteurs ultrasons détectent l'entrée et la sortie d'individus dans l'entrepôt. Ainsi, si le premier capteur reçoit une information avant le second, il s'agit d'une personne entrante. Inversement, pour 2 microsecondes, ensuite à High (haut) pour 10 microsecondes ce qui permettra d'envoyer une impulsion ultrasonique, elle est réinitialisée à Low ensuite. Afin de mesurer le temps de l'écho, on fait appel à la fonction `pulseIn()`, et ensuite on convertit (ou calcule) ceci en centimètres (distance).

Les distances calculées du premier et du deuxième capteur seront alors affichées par la fonction `Serial.print()` dans le moniteur, afin qu'on puisse les comparer:

- Si distance capteur 1 < distance capteur 2 : cela entraînera une détection d'une entrée dans l'entrepôt.
- Si distance capteur 2 < distance capteur 1 : cela entraînera une détection d'une sortie de l'entrepôt.

Avant de relancer le processus suivant, une pause de 50 ms est requise.

Ainsi, avec ce code on pourra détecter les entrées et les sorties tout en affichant les distances.

4- Modèle analytique, numérique ou expérimental

Dans l'optique de tester la faisabilité de notre prototype, nous avons basé celui-ci sur un modèle analytique, puis avec la marge de temps restante, nous nous sommes orientés sur le modèle physique. Notre première approche fut de concevoir le programme via ARDUINO IDE afin de respecter notre concept de conception. Pour cela, nous avons inclus certaines bibliothèques nécessaires au fonctionnement des capteurs ultrasons.

Trig Pin1 est la première broche qui permettra de déclencher le premier capteur, ensuite on trouve echoPin1 qui permettra d'envoyer un signal d'écho du premier capteur. Finalement trig Pin2 et echoPin 2 effectuent les mêmes fonctions, mais ils sont utilisés pour le second capteur.

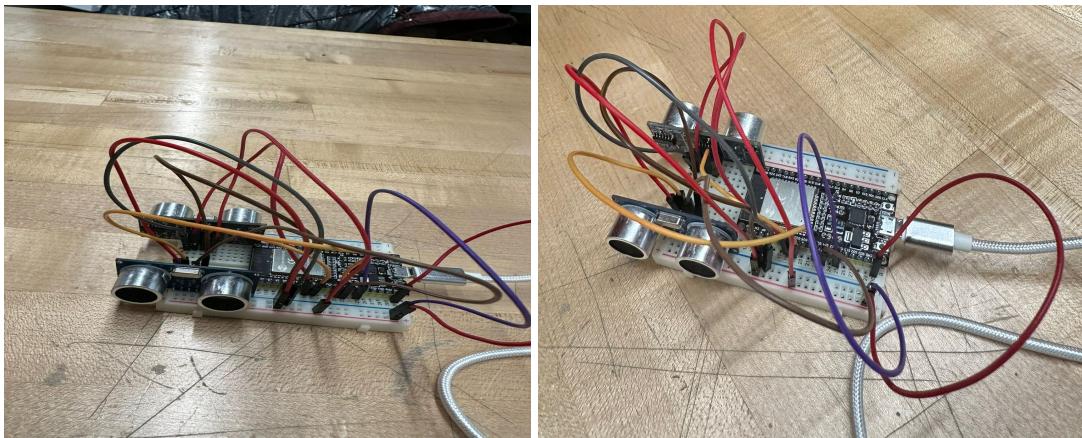
Voici le programme que nous avons conçus pour y parvenir:

```
1  | //#include <Servo.h> //orienté les capteurs ultrasons
2  | //#include <NewPing.h> //utilisateur des capteurs ultrasons
3  | //#include <LiquidCrystal.h> //afficheur LCD
4  | //#include <SD.h> //Enrégistrer les donners sur un carte SD
5
6  | // Selon connection internet:
7  | //#include <Ethernet.h>
8  | //#include <WiFi.h>
9
10 | // Pins pour les capteurs ultrasons
11 | const int trigPin1 = 27;
12 | const int echoPin1 = 14;
13 |
14 | const int trigPin2 = 33;
15 | const int echoPin2 = 32;
16 |
17 | const int MAX_DISTANCE = 10;
18
19 | //NewPing sonar(trigPin1, echoPin1, MAX_DISTANCE); // NewPing setup of pins and maximum distance.
20
21 | // Temps maximum en secondes pour rester dans la pièce
22 | const int maxTime = 30;
23
24 | // Heure d'entrée
25 | unsigned long entryTime;
26
27 | void setup() {
28 |
29 |     Serial.begin(115200);
30 |
31 |     // Initialiser les pins
32 |     pinMode(trigPin1, OUTPUT);
33 |     pinMode(echoPin1, INPUT);
34 |     pinMode(trigPin2, OUTPUT);
35 |     pinMode(echoPin2, INPUT);
36 |
37 | }
```

Livrable G - FA13

```
37    }
38
39 void loop() {
40
41     // Lire les distances
42     long duration1, distance1;
43     digitalWrite(trigPin1, LOW);
44     delayMicroseconds(2);
45     digitalWrite(trigPin1, HIGH);
46     delayMicroseconds(10);
47     digitalWrite(trigPin1, LOW);
48     duration1 = pulseIn(echoPin1, HIGH);
49     distance1 = (duration1/2) / 29.1;
50     Serial.print("1: ");
51     Serial.println(distance1);
52     delay (5);
53     long duration2, distance2;
54     digitalWrite(trigPin2, LOW);
55     delayMicroseconds(2);
56     digitalWrite(trigPin2, HIGH);
57     delayMicroseconds(10);
58     digitalWrite(trigPin2, LOW);
59     duration2 = pulseIn(echoPin2, HIGH);
60     distance2 = (duration2/2) / 29.1;
61     Serial.print("2: ");
62     Serial.println(distance2);
63
64     // Vérifier l'ordre de détection
65     if(distance1 < distance2) {
66
67         Serial.println("Entree detectee");
68         //entryTime = millis(); // Enregistrer l'heure d'entrée
69
70     }
71     else if(distance2 < distance1) {
72
73         Serial.println("Sortie detectee");
74         //entryTime = 0; // Réinitialiser l'heure d'entrée
75
76     }
77
78     /*
79     // Vérifier si délai dépassé
80     if(entryTime > 0 && millis() - entryTime > maxTime*1000) {
81
82         Serial.println("Alerte ! Personne restee trop longtemps");
83         entryTime = 0;
84
85     }
86     */
87     delay(50);
88 }
```

Désireux de tester l'efficacité de ce programme, nous avons construit un circuit basé sur une carte NODE MCU qui allie microcontrôleur et la connexion wifi.



Si le deuxième capteur reçoit une information avant le premier, il s'agit d'une personne sortante. Subséquemment, ces données sont renvoyées au serial monitor et sont affichées à l'écran.

5- Documentation, analyse et résultat du plan d'essai de prototypage

Nous nous sommes principalement concentrés sur le circuit et la programmation nécessaires au fonctionnement des capteurs ultrasons. Cette composante permet de renforcer la sécurité au sein des entrepôts, de sorte à s'assurer que les individus entrant soient ressortis au bout d'un délai raisonnable, après avoir modifié l'inventaire. Dans le cas où la personne reste longtemps, sans agir, une notification sera envoyée pour le signaler.

La carte Arduino, la planche à pain et les capteurs ultrasons seront placés dans une boîte de petites dimensions en MDF comportant 4 orifices au niveau des capteurs, permettant la détection du mouvement. Ce boîtier sera fixé au mur face à l'entrée, tel que les deux capteurs soient pointés en direction du mur opposé de l'entrée.

Nous sommes donc rendus à tester son efficacité technique face à notre problème de conception.

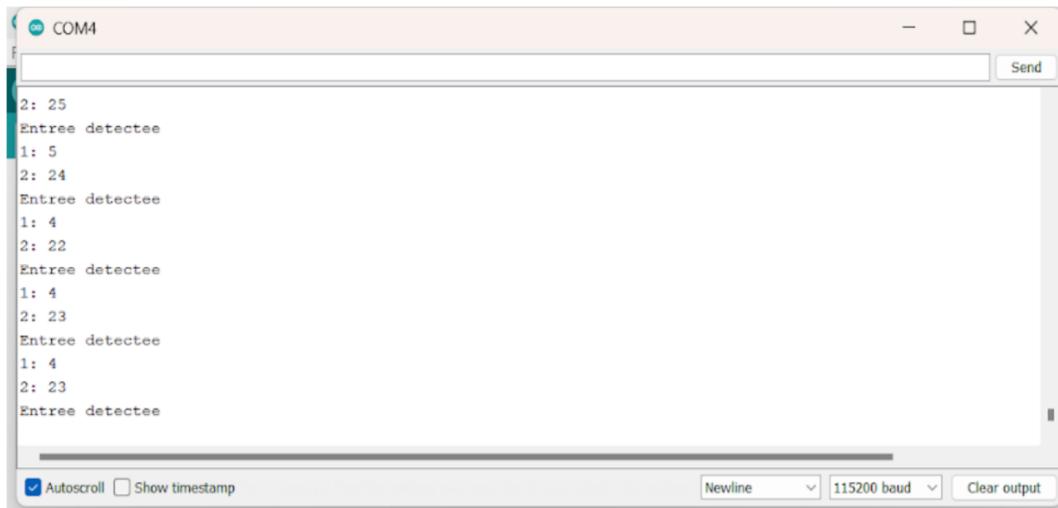
Lors de nos tests nous avons observé une fidélité moyennement forte face à nos spécifications cibles établies concernant les capteurs ultrasoniques. L'un de nos test est documenté ci-dessous:

```
2: 5
Sortie detectee
1: 37
2: 5
Sortie detectee
1: 37
2: 5
Sortie detectee
1: 37
2: 5
Sortie detectee
1: 40
2: 5
Sortie detectee
```

Moniteur série 1 - Test pour la sortie

Ce premier test fut de simuler une sortie en approchant un objet près du capteur 2 avant le capteur 1 et selon la sortie perçue par le biais du moniteur série, nous voyons que cela fonctionne.

Deuxièmement, nous avons simulé le cas inverse de la sortie pour aussi vérifier si cela fonctionnait.



Moniteur série 2 - Test pour l'entrée

Faute de temps, nous n'avons pas pu tester le paramètre permettant de lancer une alerte si une personne dépasse un certain délai lors de notre série d'essais.

6- Rétroaction du prototype des clients/utilisateurs

Une fois notre prototype conçu, nous nous sommes tournés vers des individus pouvant être nos potentiels clients afin de recueillir la rétroaction des utilisateurs. Ces retours nous permettront d'améliorer notre conception en développant de possibles solutions.

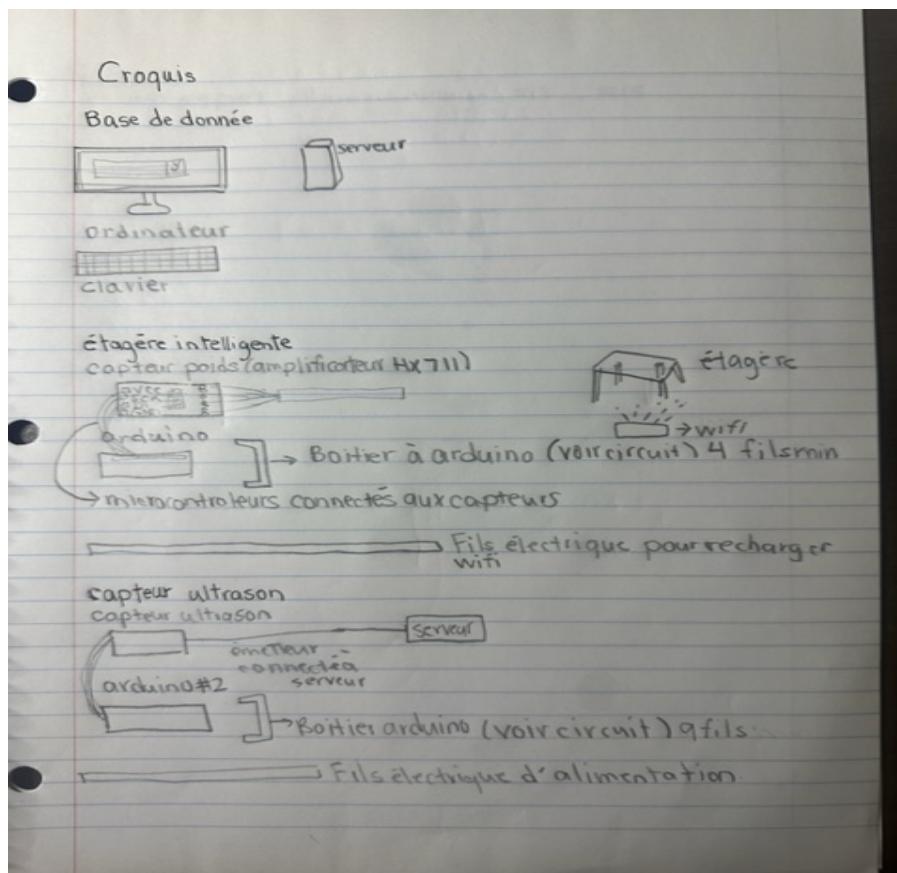
- **Rétroaction 1 :** Le prototype semble techniquement fonctionnel et esthétiquement appréciable avec l'ajout d'un boîtier permettant à la fois de masquer les différentes et de les protéger de possibles dommages extérieurs. La réduction à une seule carte Arduino au lieu de deux en alignant les capteurs sur un même côté a également été positivement relevée.
Toutefois, comment pourrait-il être alimenté en cas de panne de courant?
 - ❖ **Solution possible:** En cas de panne de courant au niveau de l'entrepôt, nous pourrions intégrer une batterie externe en tant que système d'alimentation de secours, de sorte à ce que le fonctionnement optimal des capteurs soit maintenu.
- **Rétroaction 2 :** Le prototype devrait pouvoir détecter combien de temps une personne passe à l'intérieur de l'entrepôt dans le cas où une situation d'urgence se passe ou que la personne dans l'entrepôt requiert une quelconque aide.
 - ❖ **Solution possible:** Après un certains temps, l'entrepôt devrait être en mesure grâce à un ajout dans le codage des capteurs, d'envoyer une alerte après un certains temps qu'une personne est rentrer dans l'entrepôt et qu'elle n'a pas passé une seconde fois à travers celui-ci signifiant que la personne est encore à l'intérieur. Cet ajout au système servira primordialement à la sécurité de l'entrepôt.
- **Rétroaction 3 :** Le prototype comporte des détails techniques en matière de programmation. Les deux capteurs semblent être une solution ingénieuse pour renforcer la sécurité en incluant l'envoi de notifications. Cependant, le mode d'alimentation mural pose un problème,

puisque'il pourrait handicaper le système en cas de coupure d'électricité, en arrêtant de fonctionner, ce qui nuira à la sécurité.

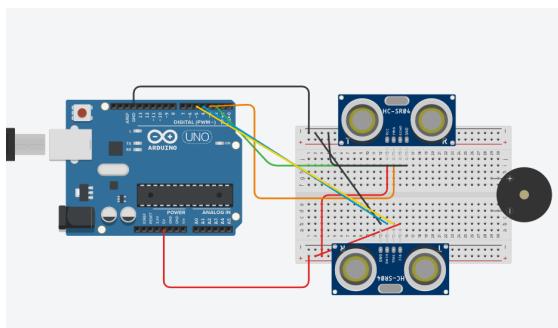
- ❖ **Solution possible:** On pourrait inclure une alimentation de secours telle qu'une batterie rechargeable, programmée pour fonctionner automatiquement en cas de coupure d'électricité. Un mode de repos pourrait également réduire les dépenses : mise en veille automatique en cas d'absence de mouvement.
- **Rétroaction 4:** Le prototype est bien configuré et constitue un concept intéressant. Cependant, comment comptez-vous le fixer sur un mur? Comment allez-vous créer une boîte pour facilement changer des composants défectueux?
 - ❖ **Solution possible:** Nous pourrions créer une petite étagère permettant de le poser sur le mur devant l'entrée. Pour la boîte, nous pourrions incorporer des retraits en haut de la boîte et un couvercle permettant de glisser au niveau de ces retraits, et ainsi, accéder facilement au composant concerné.
- **Rétroaction 5:** Le système de capteurs ultrasons aide à renforcer la sécurité au sein des entrepôts, mais il faudrait considérer leur limite en termes de résolution spatiale. Comment est-ce que les capteurs ultrasons pourront détecter la différence entre une personne entrante et sortante, et plusieurs individus qui entrent simultanément à l'intérieur de l'entrepôt. Il faudrait être en mesure de savoir si plusieurs personnes sont dans la salle.
 - ❖ **Solution possible:** Nous pourrions paramétrier les capteurs de telles sortes à détecter le nombre de personnes entrantes et sortantes à l'aide de leur signal. Seulement, notons qu'une telle définition dépend également du type de capteur utilisé.

7- Mise à jour des spécifications cibles, de la conception détaillée et de la NDM

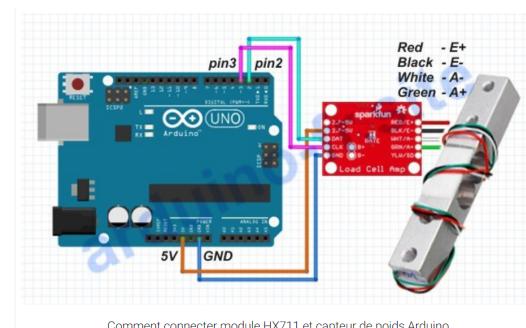
Les rétroactions obtenues nous permettent de réorganiser notre projet et donc de produire une nouvelle conception détaillée qui correspond plus aux modifications que nous avons apportées. Il est important de mettre à jour cette conception, de sorte à ce qu'elle englobe toutes les composantes de notre projet et nous permette de maintenir un fil conducteur et de plus ou moins visualiser leurs interfaces. Ci-dessous est présentée notre conception détaillée actuellement modifiée:



Croquis - Conception détaillée du système



Composante 1- capteur ultrasons



Comment connecter module HX711 et capteur de poids Arduino

Composante 2- Capteur de poids

En ce qui concerne les spécifications cibles, nous les avons également actualisés en ajoutant de nouvelles exigences, en modifiant l'ordre d'importance de certaines spécifications et en supprimant d'autres que nous n'avons finalement pas jugés utile d'inclure, comme affiché dans le tableau suivant le modèle de spécifications de conception technique (SCT) ci-dessous:

Legende:

5: Critique - 4: Très désirable - 3: Bien mais n'est pas nécessaire - 2: Pas important - 1: Indésirable.

N°	Spécifications	Priorisation	Estimations
1	Poids	3	< 0.3 kg
2	Dimensions du scanner à code QR	4	Rectangulaire: 9cm x 7cm x 5cm

Livrable G - FA13

3	Coût	3	≤ 50 \$
4	Nombres d'items que la plateforme peut accepter	4	100.000 items
5	Usabilité	5	Très facile d'usage Adaptée à toutes les tranches d'âges
6	Sécurité	5	Pas piratable Détection d'individus entrant et sortant de l'entrepôt
7	Esthétique	2	Design futuriste
8	Portabilité avec les tags avec les camions	4	Partout dans le Canada
9	Fréquence des mises à jour automatiques	5	Instantanément
10	Nombres d'utilisateur que la plateforme peut accepter simultanément	4	50 utilisateurs
11	Fréquence des rapports de l'inventaire	5	Hebdomadaire
12	Fiabilité	5	Élevée (réduction de l'erreur humaine)

Tableau - Spécifications cibles de Conception Technique du produit

Relativement à la nomenclature des matériaux, nous y avons apporté quelques modifications afin de l'adapter aux nouvelles normes de notre projet. Ci-dessous est présentée la nouvelle nomenclature:

NUMÉRO	PRODUIT	DESCRIPTION	QUANTITÉ	PRIX UNITAIRE	PRIX ETENDU	LIEN
1	Lot de 4 Capteurs de poids de 50 kg	Module AD pour Arduino	1	12.98\$	12.98\$	https://www.amazon.ca/-/fr/DIYmalls-Lot-capteurs-poids-amplificateur/dp/B086ZHZNJH/ref=sr_1_4?crid=3870ODUKMMGP0&keywords=capteur+poids+arduino&qid=1699840850&s=industrial&sprefix=%2Cindustrial%2C145&r=1-4
2	Planches de contreplaqué mince (MDF)	18 inch x 24 inch	3	3.00\$	9.00\$	https://makerstore.ca/shop/ols/products/mdf/v/M003-1-8-18-NCH
3	HX711 Master idem	Bibliothèque pour la pesée	1	0\$	0\$	https://github.com/RobTillaart/HX711
4	Câbles ruban Dupont	Connexion entre les composantes	20	0\$	0\$	https://makerstore.ca/shop/ols/products/jumper-cables-per-10/v/JMP-CBL-20C-MLFML
5	Capteurs de distance	Titri HC-SR04	2	0\$	0\$	https://ca.robotshop.com/products/sunfounder-pir-sensor-module-hc-sr04?gad_source=4&gc

Livrable G - FA13

	ultrasonique					https://www.arduino.cc/reference/en/libraries/n_eeping/
6	Breadboard	Support des branchements pour les fils	1	0\$	0\$	https://makerstore.ca/shop/ols/products/breadboard/v/B15-BRD-FLL
7	NEW PING	Bibliothèque de capteurs ULTRASONS	1	0\$	0\$	https://www.arduino.cc/reference/en/libraries/n_eeping/
8	Hébergeur de la base de données	INFOMANIAK	1	10\$	10\$	https://shop.infomaniak.com/order2/select/domain
TOTAL (sans taxes + livraison)					31.98\$	
TOTAL (avec taxes + livraison)					36.14\$	

8- Plan d'essai de prototypage du troisième prototype

En vue de nous préparer à la fabrication du prototype au prochain livrable, il est nécessaire d'établir un plan d'essai de prototypage, incluant les objectifs à atteindre. Ainsi, dans un premier temps, nous définissons un critère d'arrêt, à savoir le fait qu'à partir de moins de trois articles entrant et sortant soient correctement répertoriés, autrement dit, détectés et ajoutés effectivement à la base de données, sans erreurs. Dans un second temps, nous suivons les étapes ci-dessous, nécessaires au développement de notre plan d'essai de prototypage.

- Étape 1 : Choix des spécifications cibles et des métriques que nous souhaitons évaluer.
- Étape 2 : Détermination d'un objectif pour chaque prototype, le plan de test étant basé sur ces spécifications. Celui-ci constituera éventuellement une justification pour valider la haute priorité de la métrique considérée.
- Étape 3 : Mention des moyens d'évaluer les métriques et la manière dont les résultats seront collectés.
- Étape 4 : Développement et construction d'un prototype capable d'effectuer les tests mentionnés en apportant les résultats souhaités.

Dans notre cas, nos prototypes sont axés sur le système de base de données express. Par conséquent, nous établissons le plan de prototypage suivant:

Prototypes					Tests		
N°	Type	Objectif	Fidélité	Rétroaction	Objectif	Description	Durée

Livrable G - FA13

1	Complet analytique	Analyser des sous-systèmes critiques	Élevée	Aucune du client ou d'utilisateur	1) Intégration entre le système d'inventaire et la base de données 2) Évaluation de la performance de la base de données	1) Connecter le système d'inventaire à la base de données en configurant les paramètres, créer des données et vérifier qu'elles sont correctement affichées. 2) Mesurer le temps de réponses à chaque opération effectuée	2 heures (13-11-2023)
2	Complet analytique	Analyser des sous-systèmes critiques	Élevée	Aucune du client ou d'utilisateur	1) Capacité de sauvegarde de la base de données 2) Accès sécurisé à la base de données	1) Planifier la sauvegarde en définissant la fréquence, le type et le lieu de stockage, effectuer une sauvegarde automatique et vérifier sa validité. 2) Vérifier qu'une authentification à l'aide d'un identifiant unique et un mot de passe personnalisé est requise à chaque demande d'accès.	2 heures (13-11-2023)
3	Complet analytique	Vérifier la faisabilité	Élevée	Aucune du client ou d'utilisateur	1) Mise à jour rapide et automatique de la base de données 2) Vérifier la capacité à rechercher et à filtrer les données. 3) Vérification de la fonctionnalité du prototype final	1) Apporter une modification, mesurer le temps nécessaire pour l'effectuer et vérifier sa justesse. 2) Effectuer des requêtes de recherche relatives à un article ou encore un fournisseur, appliquer des filtres et vérifier les résultats obtenus. 3) Réaliser les opérations de bases (créer des données, lire leur affichage, les supprimer et consulter la mise à jour) et vérifier leur fonctionnement correct.	2 heures (13-11-2023)

9- Conclusion

En somme, notre deuxième prototype permet de tester la faisabilité du système de capteurs ultrasons. Ainsi, nous avons créé le plan d'essai de prototype, obtenu des rétroactions et planifié le plan d'essai de prototype du prochain prototype. Lors du livrable I, nous allons donc concevoir notre troisième

prototype, basé sur la base de données EXPRESS, et itérer de nouveau les étapes présentement effectuées.

10- Bibliographie

- “Le fonctionnement et la technologie des détecteurs à ultrasons.” n.d. Baumer. Accessed November 12, 2023.
https://www.baumer.com/fr/fr/service-assistance/fonctionnement/le-fonctionnement-et-la-technologie-des-detecteurs-a-ultrasons/a/Know-how_Function_Ultrasonic-sensors.
- “Comment utiliser HC-SR04 capteur ultrason avec Arduino.” 2023. Moussasoft.
<https://www.moussasoft.com/hc-sr04-capteur-ultrason-avec-arduino>.
- “Mesurer une distance avec un capteur à ultrason HC-SR04 et une carte Arduino / Genuino.” 2016. Carnet du maker.
<https://www.carnetdumaker.net/articles/mesurer-une-distance-avec-un-capteur-ultrason-hc-sr04-et-une-carte-arduino-genuino/>.

Lien pour Wrike:

<https://www.wrike.com/workspace.htm?acc=4975842&wr=20#/folder/1222362642/timeline3?viewId=204157457>