Nicolas Soulard-Bouchard

Sébastien Jean

**TP2 : Pompe à Essence**

Travail présenté à monsieur Alain Parent & Pierre-François Léon

Applications mobiles et objets connectés.

420-W46-SF

Départmement de la formation continue

Programmation, bases de données et serveurs

Cégep de Sainte-Foy

13 Janvier 2021

**Description détaillée du projet**

Description du contexte

Dans le cadre du cours « Applications mobiles et objets connectés », nous avons été mandantés afin de réaliser un système d’affichage pour l’achat de carburant.

Afin de réaliser cette tâche nous auron besoin d’au moins 3 composants physiques, ceux-ci étant un afficheur de quatre fois sept segments ainsi que deux boutons, l’un simulant la poignée de la pompe et l’autre simulant le paiment/la fin de la transaction/la réinitialisation de la pompe par le commis. Sommairement le second bouton devra remettre les information de l’afficheur à zéro.

L’affichage devra prendre la forme suivante sur notre afficheur : 00.00 (toujours deux chiffres avant le point et deux chiffres après le point) et devra commencer à zéro au début du plein.

Dans un premier temps nous devront faire en sorte que lorsque le bouton simulant la poignée de la pompe est tenu enfoncé, les informations à l’écran devront augmenter. Le volume du plein devra augmenter à raison de dix litres par minutes tandis que le prix augmentera proportionnellement au volume, au coût de 1.07$ le litre. Le plein devra s’interrompre automatiquement quand la limite d’affichage (99.99) du volume ou du prix sera atteinte.

Nous devrons alors afficher le prix du plein d’essence en temps réel lorsque le bouton est activé. Lorsque le bouton cesse d’être activé, le prix doit continuer s’afficher pendant 5 secondes. Suite à ce délais, l’afficheur devra afficher en alternance le prix du plein et le volume de celui-ci. Nous avons décidé d’utiliser un délais d’alternance de deux secondes, afin que l’affichage soit lisible pour le client sans toutefois être trop long. Cela permettra au client de voir rapidement et clairement le prix et le volume. Nous devrons également pouvoir poursuivre l’incrémentation du prix si le client réactive la pompe, tout en repassant à l’affichage constant du prix (sans alternance avec le volume).

Finalement, nous devrons programmer le bouton de réinitialisation de l’affichage. Lorsqu’il sera activé (appuyer simplement une fois, il ne sera pas nécessaire de le tenir enfoncé) nous devrons remettre le compteur de volume et de prix à zéro, teminant le plein précédent. L’afficheur sera alors prêt pour le prochain client.

|  |  |
| --- | --- |
| **Tâche** | **Nom du membre de l’équipe** |
| **Documentation** | |
| Rédaction de la liste de tâches | Sébastien Jean |
| Rédaction du document | Sébastien Jean |
| Rédaction du contexte | Sébastien Jean |
| Inventaire des pièces | Sébastien Jean |
| Réalisation du diagramme de classes | Sébastien Jean |
| **Simulation (Thinkercad)** | |
| Récupérer le schéma | Nicolas Soulard-Bouchard |
| **Programmation** | |
|  |  |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Inventaire des pièces** | |
| **Pièce** | **Quantitée** |
| Microcontroleur Arduino | 1 |
| Afficheur 4 x 7 segments | 1 |
| Bouton poussoir | 2 |
| Résistances 1000Ω | 2 |
| Résistances 510Ω | 8 |
| Mini platine d’essaie | 2 |
| Platine d’essaie | 1 |
| Cables cavalier ( longueurs & couleurs variées) | 60\* |
| \*Veuillez noter que nous avons mis un nombre de cables supérieur au nombre réel utilisé pour le montage afin de s’assurer d’avoir des cables de rechange au besoin. | |

|  |  |  |
| --- | --- | --- |
| **Carnet de bord** | | |
| **Tâche** | **Nom du membre d’équipe** | **Durée de réalisation** |
| **Documentation** | | |
| Rédaction de la liste de tâches | Sébastien Jean |  |
| Rédaction du document | Sébastien Jean |  |
| Rédaction du contexte | Sébastien Jean | 60 minutes |
| Inventaire des pièces | Sébastien Jean | 20 minutes |
| Réalisation du diagramme de classes | Sébastien Jean |  |
| **Simulation (Thinkercad)** | | |
| Récupérer le schéma & le code. | Nicolas Soulard-Bouchard | 5 minutes |
| **Programmation** | | |
|  |  |  |
|  |  |  |
| Debugage | Sébastien Jean |  |
| Debugage | Nicolas Soulard-Bouchard |  |