Louca Létourneau et Maxime Roy

Choix technologique

**Les langages à utiliser :**

Pour programmer l’interface web du projet nous allons utiliser CSS/HTML puisque c’est le langage par défaut des navigateurs web.

Nous allons ajouter du PHP pour le fonctionnement web au niveau du serveur. Le PHP sera utilisé pour récupérer les informations de la base de données et pour interagir avec le AJAX. Nous avons choisi PHP au lieu d’un autre langage du même type puisque c’est celui que nous avons déjà travaillé avec et nous sommes donc déjà expérimenté avec celui-ci.

Pour mettre la page à jour et afficher les informations en temps réel, nous allons utiliser AJAX/JS puisque que nous avons déjà travaillé avec et nous avons déjà un peu expérimenté avec celui-ci.

MySQL sera utilisé pour la gestion de la base de données puisque nous sommes familiers avec son fonctionnement et fonctionne bien avec Workbench.

Pour la programmation sur le Raspberry Pi nous utiliserons le langage Python pour simuler le procédé de la machine à trier les bouchons. Nous avons choisi Python puisqu’il y a de nombreux tutoriel disponible en ligne et puisque la programmation sur Raspberry Pi utilise beaucoup ce langage.

Nous utiliserons aussi le C# pour la programmation de l’envoie de données de l’automate (S7-1200) vers la base de données. Nous avons choisi C# parce que le S7.Net PLC driver qui permet de communiquer avec l’automate fonctionne dans ce langage.

**Les Frameworks à utiliser :**

Nous allons utiliser le Framework .Net pour utiliser l’outil de gestionnaire de paquet NuGet et installé S7.Net PLC driver.

**L’implémentation et les outils nécessaires au fonctionnement du système :**

Un kit Raspberry et différent câble de branchement.

Un serveur afin d’hébergé notre site web et notre base de données.

**Les périphériques d’entrées/sorties :**

Un Raspberry Pi 4 (Canakit starter kit) munie d’un capteur de température et d’humidité.

**Système de gestion de base de données :**

Pour le la gestion de la base de données nous allons utiliser MySQL Workbench puisque nous somme familier avec son interface.

**Outil nécessaire au développement :**

Pour le développement du projet nous utiliserons un Raspberry Pi 4 et son kit de branchement afin de simuler l’automate présent sur la machine du client. Cela nous permettra développer un moyen de récupérer les différentes informations de production. Cette technologie nous permettra d’effectuer des tests sécuritaires à partir du cégep sans risque d’endommagé le matériel du client.

Nous allons aussi utiliser PyCharm pour le développement en Python sur le Raspberry Pi et pour l’interface web nous utiliserons Visual Code pour éditer notre code PHP, AJAX, etc.

Pour test local, nous allons utiliser WampServer pour faire fonctionner notre code PHP et AJAX.

Pour notre gestionnaire de source nous utiliserons SourceTree puisque nous l’avons déjà utilisé et qu’il fonctionne généralement bien. L’interface graphique de SourceTree facile son utilisation comparativement à git bash par exemple.