

Cahier des charges : Timbreuse

Le but de ce projet est de suivre précisément les activités (Arrivées et Départs) des élèves. Ce projet doit être repris pour la troisième fois. Il a été initialement conçu en Java en utilisant le *framework* *Tomcat*. Pour différentes raisons décrite dans le document « Choix et Motivations » le *framework* *NodeJS* sera utilisé à présent.

I. Premier niveau

- ☐ Écrire les données brutes de timbrage (tag et heure de timbrage) dans un fichier CSV qui sera changé chaque jour/semaine (N_Timbreuse_Date.csv).
- ☐ Les fichiers CSV seront synchronisés en permanence entre les timbreuses (voir figure ci-dessous)

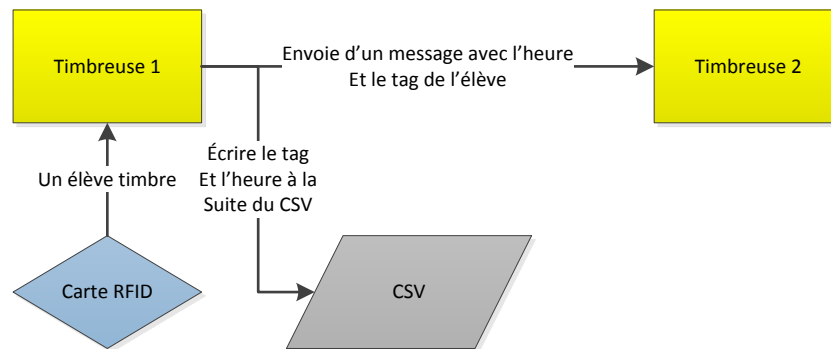


Figure 1 Schéma sur le fonctionnement de la distribution des CSV

- ☐ Pouvoir sauvegarder les fichiers CSV à l'aide d'une clé maître.
 - Premier jalon à la 4ème semaine

II. Deuxième niveau

a. Système

- ☐ Pouvoir suivre les arrivées et les départs des élèves
- ☐ Calculer le temps effectué par jour et le soustraire au temps obligatoire correspondant à la journée
- ☐ Utilisé un Raspberry Pi par classe et un autre qui servira de serveur (voir figure ci-dessous).

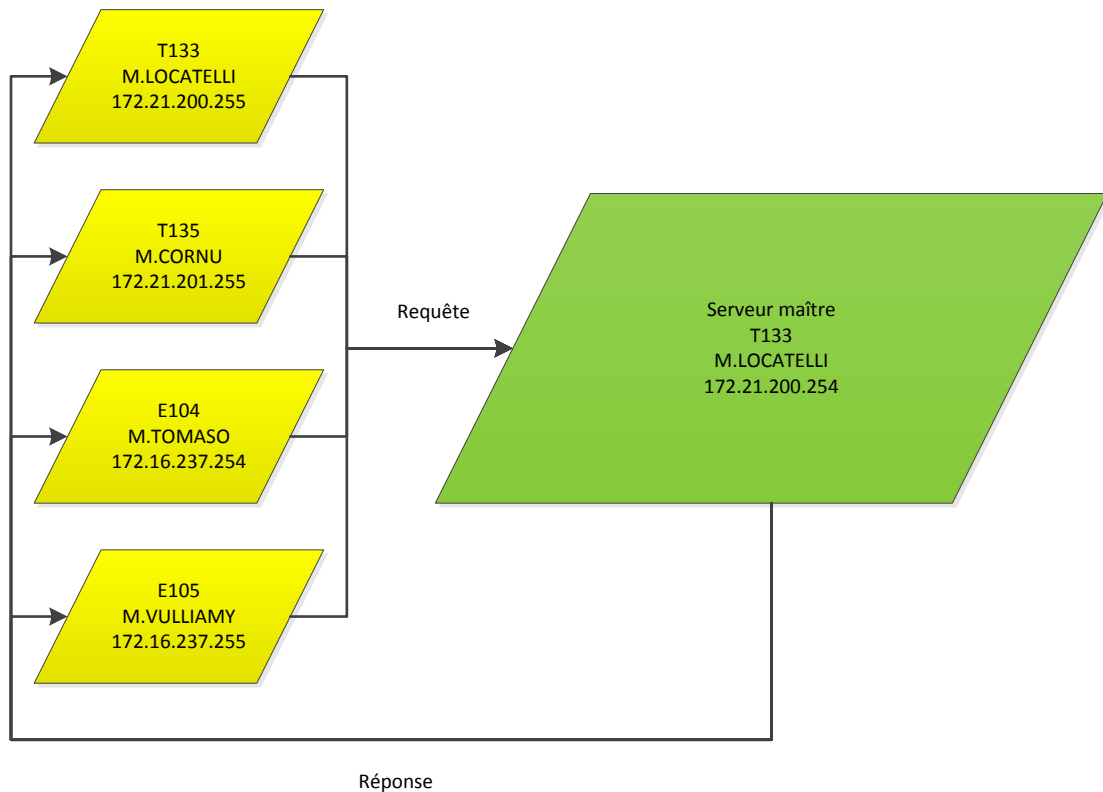


Figure 2 Schéma de l'organisation des machines au sein du réseau du CPNV

- ☐ Avoir un seul exécutable pour tous les logiciels et pouvoir choisir lequel faire fonctionner en fonction d'un argument au démarrage de l'application. (Voir figure ci-dessous)

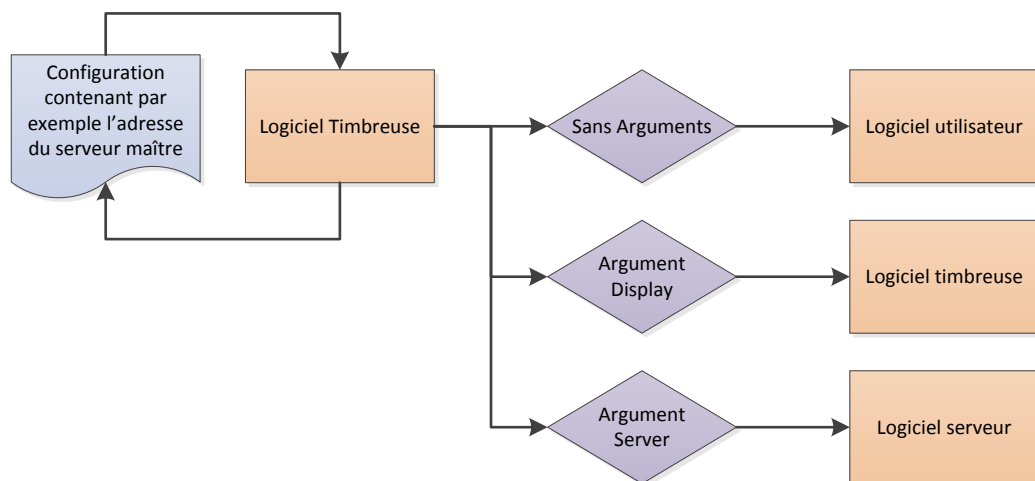


Figure 3 Schéma du lancement de tous les processus à partir d'un même exécutable

- ☐ Pouvoir faire aisément des sauvegardes de la base de données sous la forme d'un fichier CSV
- ☐ Placer tous les systèmes dans des boîtes.
- ☐ Ajouter des méthodes pour gérer les lois concernant la quantité de travail (Pause obligatoire, etc...). Cette fonction sera exécutée en fin de journée
- ☐ Avoir un serveur maître avec des timbreuses esclaves

- ☐ Si la timbreuse maîtresse vient à tomber en panne, toutes les requêtes sont enregistrées dans une pile et seront exécuté au retour en ligne du serveur. De plus la liste de toutes les requêtes de la journée seront enregistré en CSV tous les jours.
- ☐ Afficher l'heure actuelle, les heures faites aujourd'hui et la différence d'heure totale lorsqu'on timbre sur une timbreuse.

b. Administrateurs

- ☐ Permettre de crée ou d'effacer des professeurs
- ☐ Permettre de crée ou d'effacer des classes
- ☐ Assigne le professeur référent à chaque élève
- ☐ Mettre en place une interface (Voir un exemple sur la figure ci-dessous). Une ergonomie pourra être discuter à la fin de la 4^{ème} semaine.



Figure 4 Exemple d'interface pour administrateur

c. Professeurs

- ☐ Permettre aux professeurs d'ajouter ou de soustraire du temps à des élèves avec une description.
- ☐ Permettre aux professeurs de mettre un élève absent
- ☐ Permettre aux professeurs de changer les dates de vacances et de congé
- ☐ Permettre aux professeurs de crée ou d'effacer des élèves
- ☐ Permettre aux professeurs de suivre avec précision le temps de travail effectué par les élèves
- ☐ Pouvoir contrôler toutes ces fonctions depuis une application bureau
- ☐ Pouvoir choisir d'accepter ou de refuser une demande de congé et l'intégrer directement au système si oui
- ☐ Pouvoir mettre des élèves en horaire fixe
- ☐ Mettre en place une interface (Voir un exemple sur la figure ci-dessous)



Figure 5 Exemple d'interface pour professeur

d. Élèves

- ☐ Permettre aux élèves de timbrer depuis les timbreuses
- ☐ Formulaire de demande de congé
- ☐ Alerter les élèves en qui ont fait moins de temps que le temps demandé
- ☐ Mettre en place une interface (Voir un exemple sur la figure ci-dessous)

III. Troisième niveau

- ☐ Pouvoir modifier le règlement du la timbreuse facilement dans le code
- ☐ Ajouter des logs lorsque le système modifie des valeurs lors des mauvaises applications du règlement

Date :

Lieu :

Signature de M.LOCATELLI :

Signature de M.LE ROY :

Signature de M.VULLIAMY :

Signature de M.CORNU :

Signature de M.TOMMASO :