

Journal de bord : Timbreuse

27/02/17 : Documentation sur les différents outils qui vont être utilisés comme NodeJS, Electron et les fonctionnalités de sqllite.

28/02/17 : Début de la mise en place de la base de données. Création de la fonction permettant de générer des bases de données de base.

01/03/17 : Suite de la mise en place de la base de données et début des fonctions de timbrage, c'est-à-dire les fonctions permettant de suivre les heures effectuées par les élèves.

02/03/17 : Usinage de la boîte du serveur.

06/03/17 : Mise en place de la pile de requête des timbreuses permettant aux élèves de continuer à timbrer en cas de panne de réseaux. Création d'une documentation en HTML.

06/03/17 : Suite de la mise en place de la pile de requête. Mise en place d'un environnement de test créé avec des machines virtuelles basées sur le système d'exploitation Debian, proche de Raspbian, pour avoir des tests fiables

07/03/2017 : Début de la création de la fonction permettant d'exporter la base de données tout entière sur un clé USB externe.

09/03/2017 : Fin de la fonction d'export de la base de données. Amélioration du système permettant que les timbreuses restent constamment connectées au serveur et qu'elles restent constamment connectées entre elles.

13/03/2017 : Mise en place de la fonction permettant aux timbreuses de copier leurs CSV de façon miroir lors du timbrage des élèves et réparation d'une erreur de conception sur la fonction permettant aux timbreuses d'être constamment connectées.

14/03/2017 : Fonction de fin de journée et amélioration de la configuration

16/03/2017 : Fin de la fonction de fin de journée. Début de réflexion sur comment appliquer le règlement de la timbreuse. Début de l'implémentation des détections de certaines erreurs utilisateurs.

17/03/2017 : Mise en place de la fonction de pause minimum. L'élève doit faire une pause toutes les quatre heures.

20/03/2017 : Début de la mise en place du système permettant de faire des demandes de congé. On peut créer des demandes de congé, les modifier ainsi que les effacer. Reste à automatiser leur mise en place. M. LOCATELLI demande un document décrivant les différents cas de figure **d'erreurs de timbrage** ainsi que les conséquences que cela implique. Un tel document a donc été écrit dans la journée.

21/03/2017 : Mise en place d'une base pour l'interface utilisateur, afin qu'il soit facile de créer des nouvelles pages (+modularité). C'est-à-dire, création d'une page (WEB) de base contenant un

élément **iframe** (page web dans une page web) permettant d'afficher d'autre page web à l'intérieur de la page principale. Ça permet de garder une page de menu très modulable avec des pages annexes qui viennent s'y greffer.

23/03/2017 : À la suite d'une discussion avec M.LOCATELLI, le design suivant sera adopté pour la page principale de l'interface des professeurs : une liste sous forme de tableau, avec possibilité d'afficher toutes les informations de l'élève dans une **nouvelle page**. Début de la création de cette la page d'accueil de l'interface des enseignants. Des problèmes sont rencontrés quant au niveau de la communication en utilisant le protocole **IPC** entre le processus de rendu visuel et le processus de méthode pour le logiciel client.

24/03/2017 : Problème dans le processus de rendu visuel : plusieurs librairies reportent que la librairie **jQuery** n'est pas définie. Il se trouve qu'il faut employer **la méthode de NodeJS** pour l'inclusion (*require*) plutôt que la méthode HTML (`<script src : "....."></script>`). De plus, la communication entre le processus de rendu visuel et le processus de méthode a pu être rétablie en remplaçant l'objet **iframe** par un objet **webview** (Protocole supporté par l'API du framework Electron pour des raisons de sécurité). Après la revue du projet avec M.LOCATELLI et M.CORNU, certains documents (cahier des charges, erreurs de timbrage, diagramme de Gantt) ont dû être corrigés. Finalement le document sur les **erreurs de timbrage** a été revu et validé par les deux professeurs. Un document sur le projet de déploiement a été écrit.

27/03/2017 : En début de journée, le **document sur le déploiement** de la timbreuse a été rendu à M.LOCATELLI. Une discussion a eu lieu en début d'après-midi concernant le risque sécuritaire qu'impliquent **les mots de passes navigant en clair** dans le système de la timbreuse. La fonction permettant de **se connecter** a été créée. Une **erreur** est toujours présente cependant, la fonction est inaccessible depuis la page web (Impossible d'appeler le script JS depuis la page HTML). Une **solution** sera trouvée dans la journée de demain.

28/03/2017 : Création de la fonction permettant de **créer** de nouveaux élèves. On peut remplir les informations directement sur le logiciel et l'élève sera créé correctement dans la base de données. Il sera possible, par la suite, de **modifier les informations** relatives à l'élève. Lors de la **création d'un nouvel élève**, on doit créer deux entités dans la base de données. La première est un « User » : dans cette table, vont être stockées les informations personnelles de l'élève (Nom, prénom, email). La deuxième entité est un « Student », elle contient les informations relatives à la quantité de travail de l'élève.

30/03/2017 : Création de la fonction permettant **d'effacer des élèves**. Elle a la même problématique que la création, étant donné que les informations sont organisées dans deux tables différentes de la base de données. **Rassemblement du matériel** en préparation du premier déploiement qui approche. Mise en route d'un des deux Raspberry pi, installation des divers logiciels, création d'une image réinscriptible sur d'autres Raspberry pi. Après avoir lancé l'application depuis le Pi, on sait que celui-ci est compatible.

31/03/2017 : Création de la fonction permettant de **modifier des élèves** déjà existants. Un problème de fuite de mémoire a été réglé. Il s'agissait d'une fonction qui ajoutait des listeners (fonction appelée si un événement spécifique intervient) sans jamais effacer ceux qui n'ont pas été appelés.

Jalon du vendredi avec M.LOCATELLI. Apport de correctif sur le diagramme de Gantt, il manquait les dates de déploiement.

03/04/2017 : Réparation d'une **erreur sur la fonction pour éditer les élèves**. Il est en fait impossible d'exécuter plus de 1 niveau de sous-requête au sein d'une requête SQLite. Dernière préparation pour le déploiement du lendemain. Vérification des fonctions de base (tagger). Amélioration de la lecture du tag. L'UID ne sera plus lu dans une boîte de texte, mais **directement par commande clavier** envoyées à la fenêtre. Préparation d'un tag par élève pour pouvoir effectuer des tests. Discussion avec M.LOCATELLI sur la différence entre les circuits HC (transistors à effets de champs et LS).

04/04/2017 : Préparation d'une **interface** pour les utilisateurs de la timbreuse (élèves) avec l'heure affichée de façon dynamique. Vérification du fonctionnement dans un **environnement virtuel**. Déploiement en début d'après-midi. Les premiers tests montrent que les espaces ne sont pas supportés lors du timbrage. Les tags des élèves ne sont pas reconnus. Un formatage des tags, lors de la création et de l'édition des élèves, doit être ajouté.

06/04/2017 : Réparation d'un problème concernant le timbrage. Lorsque les tags des utilisateurs étaient enregistrés avec des espaces, le système ne les reconnaissait pas. Une amélioration de l'interface pour rendre celle-ci plus visible a eu lieu. Une subtilité du langage m'a obligé à commencer à **refaire toutes les fonctions utilisant le réseau**. En effet, le système lit les requêtes venant du réseau, non pas dans leurs **totalité**, mais pendant un temps donné. Il est donc nécessaire de créer des fonctions pour compiler ces requêtes.

24/04/2017 : Réparer les erreurs décrites par le logiciel serveur de la Timbreuse pendant les vacances. Chercher pourquoi le Pi qui servait de client est devenue **inutilisable** ! La théorie la plus probable est le soleil et un défaut de fabrication. Remplacement du pi par un nouveau. **Amélioration de la fonction de timbrage** pour que celle-ci utilise une pile de type FIFO (First In, First Out). Amélioration de l'**interface** des professeurs.

25/04/2017 : Réparation d'une **erreur d'affichage** qui faussait les données du tableau. Faire en sorte que tous **les messages réseau soient finis par un caractère d'échappement (0x00)** permettant de ne pas perdre de l'information durant les transferts. Création de 3 nouvelles fonctions. Une permettant de **remettre à zéro le compte d'un élève**. Une autre permettant de **mettre un élève à un nombre d'heures de travail** voulu (On met un élève à +3h de travail, par exemple). Une dernière fonction permettant **d'ajouter ou de soustraire du temps à l'élève**. Selon les indications de M.LOCATELLI, les paramètres concernant les heures des élèves doivent être modifiables avec une précision de l'ordre de la minute.

26/04/2017 : Finalisation des fonctions permettant de contrôler le temps des élèves. Des tests ont été effectués dans un environnement virtuel. Chacune des fonctions a été testée individuellement sur un élève test. La bibliothèque **fullcalendar** a été installée. Lecture de la documentation de la librairie.

27/04/2017 : Création des fonctions permettant aux professeurs de **voir les logs** des élèves avec précision dans un calendrier généré par la librairie **fullcalendar**. Remplacement de toutes les valeurs en secondes sur les interfaces graphiques par **des valeurs formatées en HH:MM:SS**. **Sécurisation des sockets clients** : Si un client effectuait deux actions, avant que la première ait fini de charger, des

comportements imprévus pouvaient survenir. Ce problème a été réglé. Création d'un schéma montrant le fonctionnement des clients dans leurs interfaces avec le serveur.