13/04 :

-Tests de non-régression(onglet Performance -> defaut procédé

-Réunion avec le tuteur pour définir les besoins et objectifs de l’application

-Réalisation de la maquette de l’application

14/04

-Etude des préférences pour rentrer des paramètres dans l’application et les conserver.

<https://openclassrooms.com/courses/creez-des-applications-pour-android/le-stockage-de-donnees-2>

-pb : pour ajouter le xml : « preference.xml », on ne peut plus utiliser addPreferencesFrom Ressource qui est devenu obsolète (depricated). On doit donc passer par l’utilisation d’un fragment.

https://www.youtube.com/watch?v=1RDt0Lic3Bc

<https://www.youtube.com/watch?v=oXlQzs76QIQ>

18/04 :

Le fragment est de type PreferenceFragment (Il en hérite). Le but est de charger l’xml des paramètres dans ce fragment puis ensuite, dans une activité, de charger ce fragment dans un objet de type FragmentTransaction. Cela nous pemettra ensuite d’utiliser cette activité avec l’xml paramètre associé. L’un des avantages d’utiliser un fragment est entre autres de sauvegarder les éventuels modifications dans les paramètres.

Create CellObject class

18/04 Add an AlertDialog when the user quit the application and add the resume what i did

Adapt the text for both langages in URL parameters and set the default value of URL with the MelodieNet's adress

Rename classes and begin the ListView

19/04 : Make the CellAdapter, populate the ListView in MachineActivity.java

Add all the colour corresponding to color code in the cellAdapter.

Beaucoup galéré : pb -> je voulais parcourir la ListView dans l’activité pour mettre à chaque fois le fond TextView du code couleur avec la couleur correspondant au code. C’était impossible, la solution a été de le faire directement dans CellAdapter au moment du remplissage des vues.

* Doing the same with ProductionActivity

Rafraichir : <http://www.softwarepassion.com/android-series-custom-listview-items-and-adapters/>

20/04 :

Fini les tests de non-régression

Ajout du refresh dans la liste en éxecutant un thread en arrière-plan

21/04 : On ajoute un écran de chargement de l’application au lancement permettant de mettre en avant l’entreprise et le produit « MelodieNet »

On met le refresh sur l’activité production aussi

On fait toutes les traductions oubliées

Travail sur le style général de l’application : couleurs des boutons, couleurs de fond

Ajout de la classe Gson à build.gradle.

Création d’une variable statique pour récupérer l’URL dans paramètres et la rendre accessible au service Web

Ecriture de la classe WebService qui enverra des requetes au service Web